

Leszek Rutkowski Rafał Scherer
Ryszard Tadeusiewicz Lotfi A. Zadeh
Jacek M. Zurada (Eds.)

LNAI 6113

Artificial Intelligence and Soft Computing

10th International Conference, ICAISC 2010
Zakopane, Poland, June 2010, Part I

1
Part I

 Springer

Lecture Notes in Artificial Intelligence 6113

Edited by R. Goebel, J. Siekmann, and W. Wahlster

Subseries of Lecture Notes in Computer Science

Leszek Rutkowski Rafał Scherer
Ryszard Tadeusiewicz Lotfi A. Zadeh
Jacek M. Zurada (Eds.)

Artificial Intelligence and Soft Computing

10th International Conference, ICAISC 2010
Zakopane, Poland, June 13-17, 2010, Part I

Volume Editors

Leszek Rutkowski
Częstochowa University of Technology, Poland
E-mail: lrutko@kik.pcz.czest.pl

Rafał Scherer
Częstochowa University of Technology, Poland
E-mail: rafal.scherer@kik.pcz.pl

Ryszard Tadeusiewicz
AGH University of Science and Technology, Kraków, Poland
E-mail: rtad@agh.edu.pl

Lotfi A. Zadeh
University of California, Berkeley, CA, USA
E-mail: zadeh@cs.berkeley.edu

Jacek M. Zurada
University of Louisville, KY, USA
E-mail: jacek.zurada@louisville.edu

Library of Congress Control Number: 2010927691

CR Subject Classification (1998): I.2, H.3, F.1, I.4, H.4, H.5

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-642-13207-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-13207-0 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

springer.com

© Springer-Verlag Berlin Heidelberg 2010
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper 06/3180

Preface

This volume constitutes the proceedings of the 10th International Conference on Artificial Intelligence and Soft Computing, ICAISC 2010, held in Zakopane, Poland during June 13-17, 2010. The conference was organized by the Polish Neural Network Society in cooperation with the Academy of Management in Łódź (SWSPiZ), the Department of Computer Engineering at the Czestochowa University of Technology, and the IEEE Computational Intelligence Society, Poland Chapter. The previous conferences took place in Kule (1994), Szczyrk (1996), Kule (1997) and Zakopane (1999, 2000, 2002, 2004, 2006, 2008) and attracted a large number of papers and internationally recognized speakers: Lotfi A. Zadeh, Shun-ichi Amari, Daniel Amit, Piero P. Bonissone, Zdzislaw Bubnicki, Andrzej Cichocki, Wlodzislaw Duch, Pablo A. Estévez, Jerzy Grzymala-Busse, Kaoru Hirota, Janusz Kacprzyk, Laszlo T. Koczy, Soo-Young Lee, Robert Marks, Evangelia Micheli-Tzanakou, Erkki Oja, Witold Pedrycz, Sarunas Raudys, Enrique Ruspini, Jorg Siekman, Roman Slowinski, Ryszard Tadeusiewicz, Shiro Usui, Ronald Y. Yager, Syozo Yasui and Jacek Zurada. The aim of this conference is to build a bridge between traditional artificial intelligence techniques and recently developed soft computing techniques. It was pointed out by Lotfi A. Zadeh that “Soft Computing (SC) is a coalition of methodologies which are oriented toward the conception and design of information/intelligent systems. The principal members of the coalition are: fuzzy logic (FL), neurocomputing (NC), evolutionary computing (EC), probabilistic computing (PC), chaotic computing (CC), and machine learning (ML). The constituent methodologies of SC are, for the most part, complementary and synergistic rather than competitive.” This volume presents both traditional artificial intelligence methods and soft computing techniques. Our goal is to bring together scientists representing both traditional artificial intelligence approaches and soft computing techniques. This volume is divided into four parts:

- Fuzzy Systems and Their Applications
- Data Mining, Classification and Forecasting
- Image and Speech Analysis
- Bioinformatics and Medical Applications

The conference attracted a total of 385 submissions from 44 countries and after the review process, 169 papers were accepted for publication. I would like to thank our participants, invited speakers and reviewers of the papers for their scientific and personal contribution to the conference. several reviewers were very helpful in reviewing the papers and are listed later.

Finally, I thank my co-workers Łukasz Bartczuk, Agnieszka Cpałka, Piotr Dziwiński, Marcin Gabryel, Marcin Korytkowski and the conference secretary Rafał Scherer for their enormous efforts to make the conference a very success-

ful event. Moreover, I would appreciate the work of Marcin Korytkowski, who designed the Internet submission system.

June 2010

Leszek Rutkowski

Organization

ICAISC 2010 was organized by the Polish Neural Network Society in cooperation with the Academy of Management in Łódź (SWSPiZ), the Department of Computer Engineering at Częstochowa University of Technology, and the IEEE Computational Intelligence Society, Poland Chapter.

Program Chairs

Honorary Chair	Lotfi Zadeh (USA)
	Jacek Żurada (USA)
General Chair	Leszek Rutkowski (Poland)
Co-Chairs	Włodzisław Duch (Poland)
	Janusz Kacprzyk (Poland)
	Józef Korbicz (Poland)
	Ryszard Tadeusiewicz (Poland)

Program Committee

Rafał Adamczak, Poland	Oscar Cordón, Spain
Cesare Alippi, Italy	Bernard De Baets, Belgium
Shun-ichi Amari, Japan	Nabil Derbel, Tunisia
Rafał A. Angryk, USA	Ewa Dudek-Dyduch, Poland
Jarosław Arabas, Poland	Ludmiła Dymowa, Poland
Robert Babuska, The Netherlands	Andrzej Dzieliński, Poland
Ildar Z. Batyrshin, Russia	David Elizondo, UK
James C. Bezdek, USA	Meng Joo Er, Singapore
Leon Bobrowski, Poland	Pablo Estevez, Chile
Leonard Bolc, Poland	János Fodor, Hungary
Piero P. Bonissone, USA	David B. Fogel, USA
Bernadette Bouchon-Meunier, France	Roman Galar, Poland
James Buckley, Poland	Alexander I. Galushkin, Russia
Tadeusz Burczynski, Poland	Adam Gaweda, USA
Andrzej Cader, Poland	Joydeep Ghosh, USA
Juan Luis Castro, Spain	Juan Jose Gonzalez de la Rosa, Spain
Yen-Wei CHEN, Japan	Marian Bolesław Gorzalczany, Poland
Wojciech Cholewa, Poland	Krzysztof Grąbczewski, Poland
Fahmida N. Chowdhury, USA	Garrison Greenwood, USA
Andrzej Cichocki, Japan	Jerzy W. Grzymala-Busse, USA
Paweł Cichosz, Poland	Hani Hagrass, UK
Krzysztof Cios, USA	Saman Halgamuge, Australia
Ian Cloete, Germany	Rainer Hampel, Germany

- Zygmunt Hasiewicz, Poland
 Yoichi Hayashi, Japan
 Tim Hendtlass, Australia
 Francisco Herrera, Spain
 Kaoru Hirota, Japan
 Adrian Horzyk, Poland
 Tingwen Huang, USA
 Hisao Ishibuchi, Japan
 Mo Jamshidi, USA
 Andrzej Janczak, Poland
 Norbert Jankowski, Poland
 Robert John, UK
 Jerzy Józefczyk, Poland
 Tadeusz Kaczorek, Poland
 Władysław Kamiński, Poland
 Nikola Kasabov, New Zealand
 Okyay Kaynak, Turkey
 Vojislav Kecman, New Zealand
 James M. Keller, USA
 Etienne Kerre, Belgium
 Frank Klawonn, Germany
 Jacek Kluska, Poland
 Leonid Kompanets, Poland
 Przemysław Korohoda, Poland
 Jacek Koronacki, Poland
 Witold Kosiński, Poland
 Jan M. Kościelny, Poland
 Zdzisław Kowalczyk, Poland
 Robert Kozma, USA
 László Kóczy, Hungary
 Rudolf Kruse, Germany
 Boris V. Kryzhanovsky, Russia
 Adam Krzyzak, Canada
 Juliusz Kulikowski, Poland
 Roman Kulikowski, Poland
 Věra Kůrková, Czech Republic
 Marek Kurzyński, Poland
 Halina Kwaśnicka, Poland
 Soo-Young Lee, Korea
 George Lendaris, USA
 Antoni Ligeza, Poland
 Zhi-Qiang LIU, Hong Kong
 Simon M. Lucas, UK
 Jacek Łeski, Poland
 Bohdan Macukow, Poland
 Kurosh Madani, France
 Luis Magdalena, Spain
 Witold Malina, Poland
 Krzysztof Malinowski, Poland
 Jacek Mańdziuk, Poland
 Antonino Marvuglia, Ireland
 Andrzej Materka, Poland
 Jarosław Meller, Poland
 Jerry M. Mendel, USA
 Radko Mesiar, Slovakia
 Zbigniew Michalewicz, Australia
 Zbigniew Mikrut, Poland
 Sudip Misra, USA
 Wojciech Moczulski, Poland
 Javier Montero, Spain
 Eduard Montseny, Spain
 Detlef D. Nauck, Germany
 Antoine Naud, Poland
 Edward Nawarecki, Poland
 Antoni Niederliński, Poland
 Robert Nowicki, Poland
 Andrzej Obuchowicz, Poland
 Marek Ogiela, Poland
 Erkki Oja, Finland
 Stanisław Osowski, Poland
 Nikhil R. Pal, India
 Maciej Patan, Poland
 Witold Pedrycz, Canada
 Leonid Perlovsky, USA
 Andrzej Pieczyński, Poland
 Andrzej Piegat, Poland
 Vincenzo Piuri, Italy
 Lech Polkowski, Poland
 Marios M. Polycarpou, Cyprus
 Danil Prokhorov, USA
 Anna Radzikowska, Poland
 Ewaryst Rafajłowicz, Poland
 Sarunas Raudys, Lithuania
 Olga Rebrova, Russia
 Vladimir Red'ko, Russia
 Raúl Rojas, Germany
 Imre J. Rudas, Hungary
 Enrique H. Ruspini, USA
 Khalid Saeed, Poland
 Dominik Sankowski, Poland

Norihide Sano, Japan	Burhan Turksen, Canada
Robert Schaefer, Poland	Shiro Usui, Japan
Rudy Setiono, Singapore	Michael Wagenknecht, Germany
Paweł Sewastianow, Poland	Tomasz Walkowiak, Poland
Jennie Si, USA	Deliang Wang, USA
Peter Sincak, Slovakia	Jun Wang, Hong Kong
Andrzej Skowron, Poland	Lipo Wang, Singapore
Ewa Skubalska-Rafajłowicz, Poland	Zenon Waszczyszyn, Poland
Roman Słowiński, Poland	Paul Werbos, USA
Tomasz G. Smolinski, USA	Sławo Wesolkowski, Canada
Czesław Smutnicki, Poland	Sławomir Wiak, Poland
Pilar Sobrevilla, Spain	Bernard Widrow, USA
Jerzy Stefanowski, Poland	Kay C. Wiese, Canada
Paweł Strumillo, Poland	Bogdan M. Wilamowski, USA
Ron Sun, USA	Donald C. Wunsch, USA
Johan Suykens Suykens, Belgium	Maciej Wygralak, Poland
Piotr Szczepaniak, Poland	Roman Wyrzykowski, Poland
Eulalia J. Szmidt, Poland	Ronald R. Yager, USA
Przemysław Śliwiński, Poland	Gary Yen, USA
Adam Słowik, Poland	John Yen, USA
Jerzy Świątek, Poland	Sławomir Zadrożny, Poland
Hideyuki Takagi, Japan	Ali M. S. Zalzala, United Arab Emirates
Yury Tiumentsev, Russia	
Vicenc Torra, Spain	

Organizing Committee

Rafał Scherer, Secretary
 Lukasz Bartczuk, Organizing Committee Member
 Piotr Dziwiński, Organizing Committee Member
 Marcin Gabryel, Organizing Committee Member
 Marcin Korytkowski, Databases and Internet Submissions

External Reviewers

R. Adamczak	M. Borawski	W. Cholewa
J. Arabas	A. Borkowski	R. Choraś
T. Babczyński	W. Bozejko	A. Cichocki
L. Bartczuk	T. Burczyński	P. Cichosz
A. Bielecki	R. Burduk	R. Cierniak
A. Bielskis	B. Butkiewicz	S. Concetto
J. Biesiada	C. Castro	B. Cyganek
M. Blachnik	K. Cetnarowicz	R. Czabański
L. Bobrowski	M. Chang	I. Czarnowski
P. Boguś	M. Chis	B. De Baets

K. Delac	J. Kościelny	R. Rojas
V. Denisov	L. Kotulski	L. Rolka
G. Dobrowolski	Z. Kowalczyk	I. Rudas
A. Dzieliński	J. Kozlak	M. Rudnicki
P. Dziwiński	M. Kretowski	L. Rutkowski
S. Ehtheram	R. Kruse	R. Schaefer
D. Elizondo	B. Kryzhanovsky	R. Scherer
M. Flasiński	A. Krzyzak	R. Setiono
C. Frowd	J. Kulikowski	A. Sędziwy
M. Gabryel	V. Kurkova	W. Skarbek
A. Gawęda	M. Kurzyński	A. Skowron
M. Giergiel	H. Kwaśnicka	E. Skubalska-
F. Gomide	A. Ligęza	Rafałowicz
M. Gorzałczany	J. Lęski	K. Slot
K. Grajbczewski	K. Madani	A. Słowik
K. Grudziński	W. Malina	R. Słowiński
J. Grzymala-Busse	J. Mańdziuk	T. Smolinski
P. Hajek	U. Markowska-Kaczmar	C. Smutnicki
Z. Hasiewicz	A. Marvuglia	J. Starczewski
Y. Hayashi	A. Materka	P. Strumiłło
O. Henniger	J. Mendel	J. Swacha
F. Herrera	R. Mesiar	E. Szmidt
Z. Hippe	Z. Michalewicz	P. Śliwiński
A. Horzyk	J. Michalkiewicz	J. Świątek
M. Hrebień	Z. Mikrut	R. Tadeusiewicz
A. Janczak	W. Mokrzycki	H. Takagi
N. Jankowski	E. Nawarecki	Y. Tiumentsev
J. Jelonkiewicz	M. Nieniewski	V. Torra
J. Kacprzyk	A. Nwiadomski	J. Verstraete
W. Kamiński	R. Nowicki	M. Wagenknecht
A. Kasperski	A. Obuchowicz	T. Walkowiak
V. Kecman	S. Osowski	J. Wang
E. Kerre	A. Owczarek	L. Wang
F. Klawonn	F. Pappalardo	S. Wiak
L. Koczy	K. Patan	B. Wilamowski
J. Konopacki	W. Pedrycz	P. Wojewnik
J. Korbicz	A. Pieczyński	M. Wygrałak
P. Korohoda	Z. Pietrzykowski	W. Xu
J. Koronacki	V. Piuri	F. Zacarias
M. Korytkowski	T. Przybyła	S. Zadrożny
M. Korzeń	E. Rafałowicz	J. Zieliński

Table of Contents – Part I

Part I: Fuzzy Systems and Their Applications

On the Distributivity of Fuzzy Implications over Continuous Archimedean Triangular Norms	3
<i>Michał Baczyński</i>	
Fuzzy Decision Support System for Post-Mining Regions Restoration Designing	11
<i>Marzena Bielecka and Jadwiga Król-Korczak</i>	
Fuzzy Digital Filters with Triangular Norms	19
<i>Bohdan S. Butkiewicz</i>	
A Novel Fuzzy Color Median Filter Based on an Adaptive Cascade of Fuzzy Inference Systems	27
<i>Mihaela Cislariu, Mihaela Gordan, and Aurel Vlaicu</i>	
Automatic Modeling of Fuzzy Systems Using Particle Swarm Optimization	35
<i>Sergio Oliveira Costa Jr., Nadia Nedjah, and Luiza de Macedo Mourelle</i>	
On Automatic Design of Neuro-fuzzy Systems	43
<i>Krzysztof Cpałka, Leszek Rutkowski, and Meng Joo Er</i>	
An Efficient Adaptive Fuzzy Neural Network (EAFNN) Approach for Short Term Load Forecasting	49
<i>Juan Du, Meng Joo Er, and Leszek Rutkowski</i>	
Fault Diagnosis of an Air-Handling Unit System Using a Dynamic Fuzzy-Neural Approach	58
<i>Juan Du, Meng Joo Er, and Leszek Rutkowski</i>	
An Interpretation of Intuitionistic Fuzzy Sets in the Framework of the Dempster-Shafer Theory	66
<i>Ludmila Dymova and Pavel Sevastjanov</i>	
Evolutionary Learning for Neuro-fuzzy Ensembles with Generalized Parametric Triangular Norms	74
<i>Marcin Gabryel, Marcin Korytkowski, Agata Pokropinska, Rafał Scherer, and Stanisław Drozda</i>	
Fuzzy Spatial Analysis Techniques for Mathematical Expression Recognition	80
<i>Ray Genoe and Tahar Kechadi</i>	

A Modified Pittsburg Approach to Design a Genetic Fuzzy Rule-Based Classifier from Data	88
<i>Marian B. Gorzalczany and Filip Rudziński</i>	
Automatic and Incremental Generation of Membership Functions	97
<i>Narjes Hachani, Imen Derbel, and Habib Ounelli</i>	
A Multi-criteria Evaluation of Linguistic Summaries of Time Series via a Measure of Informativeness	105
<i>Anna Wilbik and Janusz Kacprzyk</i>	
Negative Correlation Learning of Neuro-fuzzy System Ensembles	114
<i>Marcin Korytkowski and Rafał Scherer</i>	
A New Fuzzy Approach to Ordinary Differential Equations	120
<i>Witold Kosiński, Kurt Frischmuth, and Dorota Wilczyńska-Sztyma</i>	
K2F - A Novel Framework for Converting Fuzzy Cognitive Maps into Rule-Based Fuzzy Inference Systems	128
<i>Lars Krüger</i>	
On Prediction Generation in Efficient MPC Algorithms Based on Fuzzy Hammerstein Models	136
<i>Piotr M. Marusak</i>	
Fuzzy Number as Input for Approximate Reasoning and Applied to Optimal Control Problem	144
<i>Takashi Mitsuishi and Yasunari Shidama</i>	
Fuzzy Functional Dependencies in Multiargument Relationships	152
<i>Krzysztof Myszkorowski</i>	
Methods of Evaluating Degrees of Truth for Linguistic Summaries of Data: A Comparative Analysis	160
<i>Adam Niewiadomski and Oskar Korczak</i>	
On Non-singleton Fuzzification with DCOG Defuzzification	168
<i>Robert K. Nowicki and Janusz T. Starczewski</i>	
Does an Optimal Form of an Expert Fuzzy Model Exist?	175
<i>Andrzej Piegat and Marcin Olchowy</i>	
Fuzzy Logic in the Navigational Decision Support Process Onboard a Sea-Going Vessel	185
<i>Zbigniew Pietrzykowski, Janusz Magaj, Piotr Wolejsza, and Jarosław Chomski</i>	
A Hybrid Approach for Fault Tree Analysis Combining Probabilistic Method with Fuzzy Numbers	194
<i>Julwan H. Purba, Jie Lu, Da Ruan, and Guangquan Zhang</i>	

Imputing Missing Values in Nuclear Safeguards Evaluation by a 2-Tuple Computational Model	202
<i>Rosa M. Rodríguez, Da Ruan, Jun Liu, Alberto Calzada, and Luis Martínez</i>	
Neuro-fuzzy Systems with Relation Matrix	210
<i>Rafał Scherer</i>	
Fuzzy Multiple Support Associative Classification Approach for Prediction	216
<i>Bilal Sowan, Keshav Dahal, and Alamgir Hussain</i>	
Learning Methods for Type-2 FLS Based on FCM	224
<i>Janusz T. Starczewski, Lukasz Bartczuk, Piotr Dziwiński, and Antonino Marvuglia</i>	
On an Enhanced Method for a More Meaningful Ranking of Intuitionistic Fuzzy Alternatives	232
<i>Eulalia Szmídt and Janusz Kacprzyk</i>	
I-Fuzzy Partitions for Representing Clustering Uncertainties	240
<i>Vicenç Torra and Ji-Hee Min</i>	
A Quantitative Approach to Topology for Fuzzy Regions	248
<i>Jörg Verstraete</i>	
Fuzzy $Q(\lambda)$ -Learning Algorithm	256
<i>Roman Zajdel</i>	

Part II: Data Mining, Classification and Forecasting

Mining Closed Gradual Patterns	267
<i>Sarra Ayouni, Anne Laurent, Sadok Ben Yahia, and P. Poncelet</i>	
New Method for Generation Type-2 Fuzzy Partition for FDT	275
<i>Lukasz Bartczuk, Piotr Dziwiński, and Janusz T. Starczewski</i>	
Performance of Ontology-Based Semantic Similarities in Clustering	281
<i>Montserrat Batet, Aida Valls, and Karina Gibert</i>	
Information Theory vs. Correlation Based Feature Ranking Methods in Application to Metallurgical Problem Solving	289
<i>Marcin Blachnik, Adam Bukowiec, Mirosław Kordos, and Jacek Biesiada</i>	
Generic Model for Experimenting and Using a Family of Classifiers Systems: Description and Basic Applications	299
<i>Cédric Buche and Pierre De Loor</i>	

Neural Pattern Recognition with Self-organizing Maps for Efficient Processing of Forex Market Data Streams	307
<i>Piotr Ciskowski and Marek Zaton</i>	
Measures for Comparing Association Rule Sets	315
<i>Damian Dudek</i>	
Distributed Data Mining Methodology for Clustering and Classification Model	323
<i>Marcin Gorawski and Ewa Pluciennik-Psota</i>	
Task Management in Advanced Computational Intelligence System	331
<i>Krzysztof Grąbczewski and Norbert Jankowski</i>	
Combining the Results in Pairwise Classification Using Dempster-Shafer Theory: A Comparison of Two Approaches	339
<i>Marcin Gromisz and Sławomir Zadrozny</i>	
Pruning Classification Rules with Reference Vector Selection Methods	347
<i>Karol Grudziński, Marek Grochowski, and Włodzisław Duch</i>	
Sensitivity and Specificity for Mining Data with Increased Incompleteness	355
<i>Jerzy W. Grzymala-Busse and Shantanu R. Marepally</i>	
A New Implementation of the co-VAT Algorithm for Visual Assessment of Clusters in Rectangular Relational Data	363
<i>Timothy C. Havens, James C. Bezdek, and James M. Keller</i>	
User Behavior Prediction in Energy Consumption in Housing Using Bayesian Networks	372
<i>Lamis Hawarah, Stéphane Ploix, and Mireille Jacomino</i>	
Increasing Efficiency of Data Mining Systems by Machine Unification and Double Machine Cache	380
<i>Norbert Jankowski and Krzysztof Grąbczewski</i>	
Infosel++: I nformation Based Feature S election C++ Library	388
<i>Adam Kachel, Jacek Biesiada, Marcin Blachnik, and Włodzisław Duch</i>	
Stacking Class Probabilities Obtained from View-Based Cluster Ensembles	397
<i>Heysem Kaya, Olcay Kurşun, and Hüseyin Şeker</i>	
Market Trajectory Recognition and Trajectory Prediction Using Markov Models	405
<i>Przemysław Klęsk and Antoni Wiliński</i>	

Do We Need Whatever More Than k-NN?	414
<i>Miroslaw Kordos, Marcin Blachnik, and Dawid Strzempa</i>	
Pattern Recognition with Linearly Structured Labels Using Recursive Kernel Estimator	422
<i>Adam Krzyżak and Ewaryst Rafajłowicz</i>	
Canonical Correlation Analysis for Multiview Semisupervised Feature Extraction	430
<i>Olca Kursun and Ethem Alpaydin</i>	
Evaluation of Distance Measures for Multi-class Classification in Binary SVM Decision Tree.....	437
<i>Gjorgji Madzarov and Dejan Gjorgjevikj</i>	
Triangular Visualization	445
<i>Tomasz Maszczyk and Włodzisław Duch</i>	
Recognition of Finite Structures with Application to Moving Objects Identification	453
<i>Ewaryst Rafajłowicz and Jerzy Wietrzyk</i>	
Clustering of Data and Nearest Neighbors Search for Pattern Recognition with Dimensionality Reduction Using Random Projections.....	462
<i>Ewa Skubalska-Rafajłowicz</i>	
Noise Detection for Ensemble Methods	471
<i>Ryszard Szupiluk, Piotr Wojewnik, and Tomasz Zabkowski</i>	
Divergence Based Online Learning in Vector Quantization.....	479
<i>Thomas Villmann, Sven Haase, Frank-Michael Schleif, and Barbara Hammer</i>	
Using Feature Selection Approaches to Find the Dependent Features ...	487
<i>Qin Yang, Elham Salehi, and Robin Gras</i>	
Performance Assessment of Data Mining Methods for Loan Granting Decisions: A Preliminary Study	495
<i>Jozef Zurada and Niki Kunene</i>	

Part III: Image and Speech Analysis

A Three-Dimensional Neural Network Based Approach to the Image Reconstruction from Projections Problem	505
<i>Robert Cierniak</i>	
Spatial Emerging Patterns for Scene Classification	515
<i>Lukasz Kobyliński and Krzysztof Walczak</i>	

Automatic Methods for Determining the Characteristic Points in Face Image	523
<i>Mariusz Kubanek</i>	
Effectiveness Comparison of Three Types of Signatures on the Example of the Initial Selection of Aerial Images	531
<i>Zbigniew Mikrut</i>	
Combined Full-Reference Image Quality Metric Linearly Correlated with Subjective Assessment	539
<i>Krzysztof Okarma</i>	
Evaluation of Pose Hypotheses by Image Feature Extraction for Vehicle Localization	547
<i>Kristin Schönherr, Björn Giesler, and Alois Knoll</i>	
Beyond Keypoints: Novel Techniques for Content-Based Image Matching and Retrieval	555
<i>Andrzej Śluzek, Duanduan Yang, and Mariusz Paradowski</i>	
Sequential Coordinate-Wise DNMF for Face Recognition	563
<i>Rafal Zdunek and Andrzej Cichocki</i>	
A New Image Mixed Noise Removal Algorithm Based on Measuring of Medium Truth Scale	571
<i>Ning-Ning Zhou and Long Hong</i>	

Part IV: Bioinformatics and Medical Applications

Clinical Examples as Non-uniform Learning and Testing Sets	581
<i>Piotr Augustyniak</i>	
Identifying the Borders of the Upper and Lower Metacarpophalangeal Joint Surfaces on Hand Radiographs	589
<i>Andrzej Bielecki, Mariusz Korkosz, Wadim Wojciechowski, and Bartosz Zieliński</i>	
Decision Tree Approach to Rules Extraction for Human Gait Analysis	597
<i>Marcin Derlatka and Mikhail Ihnatouski</i>	
Data Mining Approaches for Intelligent E-Social Care Decision Support System	605
<i>Dariusz Drungilas, Antanas Andrius Bielskis, Vitalij Denisov, and Dalé Dzemydienė</i>	
Erythematous-Squamous Diseases Diagnosis by Support Vector Machines and RBF NN	613
<i>Vojislav Kecman and Mirna Kikec</i>	

Neural Network-Based Assessment of Femur Stress after Hip Joint Alloplasty	621
<i>Marcin Korytkowski, Leszek Rutkowski, Rafał Scherer, and Arkadiusz Szarek</i>	
Automated Detection of Dementia Symptoms in MR Brain Images	627
<i>Karol Kuczyński, Maciej Siczek, Rafał Stegierski, and Waldemar Suszyński</i>	
Classification of Stabilometric Time-Series Using an Adaptive Fuzzy Inference Neural Network System	635
<i>Juan A. Lara, Pari Jahankhani, Aurora Pérez, Juan P. Valente, and Vassilis Kodigoianis</i>	
An Approach to Brain Thinker Type Recognition Based on Facial Asymmetry	643
<i>Piotr Milczarski, Leonid Kompanets, and Damian Kurach</i>	
Application of C&RT, CHAID, C4.5 and WizWhy Algorithms for Stroke Type Diagnosis	651
<i>Igor S. Naftulin and Olga Yu. Rebrova</i>	
Discovering Potential Precursors of Mammography Abnormalities Based on Textual Features, Frequencies, and Sequences	657
<i>Robert M. Patton and Thomas E. Potok</i>	
An Expert System for Human Personality Characteristics Recognition	665
<i>Danuta Rutkowska</i>	
Author Index	673

Table of Contents – Part II

Part I: Neural Networks and Their Applications

Complex-Valued Neurons with Phase-Dependent Activation Functions	3
<i>Igor Aizenberg</i>	
ART-Type Artificial Neural Networks Applications for Classification of Operational States in Wind Turbines	11
<i>Tomasz Barszcz, Andrzej Bielecki, and Mateusz Wójcik</i>	
Parallel Realisation of the Recurrent Elman Neural Network Learning	19
<i>Jarostaw Bilski and Jacek Smoląg</i>	
The Investigating of Influence of Quality Criteria Coefficients on Global Complex Models	26
<i>Grzegorz Dralus</i>	
Quasi-parametric Recovery of Hammerstein System Nonlinearity by Smart Model Selection	34
<i>Zygmunt Hasiewicz, Grzegorz Mzyk, and Przemysław Śliwiński</i>	
Recent Progress in Applications of Complex-Valued Neural Networks . . .	42
<i>Akira Hirose</i>	
Hybrid-Maximum Neural Network for Depth Analysis from Stereo-Image	47
<i>Lukasz Laskowski</i>	
Towards Application of Soft Computing in Structural Health Monitoring	56
<i>Piotr Nazarko and Leonard Ziemiański</i>	
Persistent Activation Blobs in Spiking Neural Networks with Mexican Hat Connectivity	64
<i>Filip Piekniowski</i>	
Neurogenetic Approach for Solving Dynamic Programming Problems . . .	72
<i>Matheus Giovanni Pires and Ivan Nunes da Silva</i>	
Optimization of Parameters of Feed-Back Pulse Coupled Neural Network Applied to the Segmentation of Material Microstructure Images	80
<i>Lukasz Rauch, Lukasz Sztangret, and Jan Kusiak</i>	

Hybrid Neural Networks as Prediction Models	88
<i>Izabela Rojek</i>	
Fast Robust Learning Algorithm Dedicated to LMLS Criterion	96
<i>Andrzej Rusiecki</i>	
Using Neural Networks for Simplified Discovery of Some Psychological Phenomena	104
<i>Ryszard Tadeusiewicz</i>	
Hybrid Learning of Regularization Neural Networks	124
<i>Petra Vidnerová and Roman Neruda</i>	
Computer Assisted Peptide Design and Optimization with Topology Preserving Neural Networks	132
<i>Jörg D. Wichard, Sebastian Bandholtz, Carsten Grötzinger, and Ronald Kühne</i>	
Part II: Evolutionary Algorithms and Their Applications	
Evolutionary Designing of Logic-Type Fuzzy Systems	143
<i>Marcin Gabryel and Leszek Rutkowski</i>	
Combining Evolutionary and Sequential Search Strategies for Unsupervised Feature Selection	149
<i>Artur Klepaczko and Andrzej Materka</i>	
An Evolutionary Algorithm for Global Induction of Regression Trees ...	157
<i>Marek Krętowski and Marcin Czajkowski</i>	
Using Genetic Algorithm for Selection of Initial Cluster Centers for the K -Means Method	165
<i>Wojciech Kwedlo and Piotr Iwanowicz</i>	
Classified-Chime Sound Generation Support System Using an Interactive Genetic Algorithm	173
<i>Noriko Okada, Mitsunori Miki, Tomoyuki Hiroyasu, and Masato Yoshimi</i>	
Evolutionary Algorithms with Stable Mutations Based on a Discrete Spectral Measure	181
<i>Andrzej Obuchowicz and Przemysław Prętki</i>	
Determining Subunits for Sign Language Recognition by Evolutionary Cluster-Based Segmentation of Time Series	189
<i>Mariusz Oszust and Marian Wysocki</i>	

Analysis of the Distribution of Individuals in Modified Genetic Algorithms	197
<i>Krzysztof Pytel and Tadeusz Nawarycz</i>	
Performance Analysis for Genetic Quantum Circuit Synthesis	205
<i>Cristian Ruican, Mihai Udrescu, Lucian Prodan, and Mircea Vladutiu</i>	
Steering of Balance between Exploration and Exploitation Properties of Evolutionary Algorithms - Mix Selection	213
<i>Adam Słowik</i>	
Extending Genetic Programming to Evolve Perceptron-Like Learning Programs	221
<i>Marcin Suchorzewski</i>	
An Informed Genetic Algorithm for University Course and Student Timetabling Problems	229
<i>Suyanto</i>	
Part III: Agent Systems, Robotics and Control	
Evaluation of a Communication Platform for Safety Critical Robotics . . .	239
<i>Frederico M. Cunha, Rodrigo A.M. Braga, and Luis P. Reis</i>	
How to Gain Emotional Rewards during Human-Robot Interaction Using Music? Formulation and Propositions	247
<i>Thi-Hai-Ha Dang, Guillaume Hutzler, and Philippe Hoppenot</i>	
Discrete Dual-Heuristic Programming in 3DOF Manipulator Control . . .	256
<i>Piotr Gierlak, Marcin Szuster, and Wiesław Żylski</i>	
Discrete Model-Based Adaptive Critic Designs in Wheeled Mobile Robot Control	264
<i>Zenon Hendzel and Marcin Szuster</i>	
Using Hierarchical Temporal Memory for Vision-Based Hand Shape Recognition under Large Variations in Hand's Rotation	272
<i>Tomasz Kapuscinski</i>	
Parallel Graph Transformations with Double Pushout Grammars	280
<i>Leszek Kotulski and Adam Sędziwy</i>	
Ant Agents with Distributed Knowledge Applied to Adaptive Control of a Nonstationary Traffic in Ad-Hoc Networks	289
<i>Michał Kudelski and Andrzej Pacut</i>	
Dynamic Matrix Control Algorithm Based on Interpolated Step Response Neural Models	297
<i>Maciej Ławryńczuk</i>	

Approximate Neural Economic Set-Point Optimisation for Control Systems	305
<i>Maciej Lawryńczuk and Piotr Tatjewski</i>	
Injecting Service-Oriented into Multi-Agent Systems in Industrial Automation	313
<i>J. Marco Mendes, Francisco Restivo, Paulo Leitão, and Armando W. Colombo</i>	
Design of a Neural Network for an Identification of a Robot Model with a Positive Definite Inertia Matrix	321
<i>Jakub Możaryn and Jerzy E. Kurek</i>	
A Fast Image Analysis Technique for the Line Tracking Robots	329
<i>Krzysztof Okarma and Piotr Lech</i>	
Multi-agent Logic with Distances Based on Linear Temporal Frames	337
<i>Vladimir Rybakov and Sergey Babenyshev</i>	
On Data Representation in Reactive Systems Based on Activity Trace Concept	345
<i>Krzysztof Skrzypczyk</i>	

Part IV: Various Problems of Artificial Intelligence

Optimization of the Height of Height-Adjustable Luminaire for Intelligent Lighting System	355
<i>Masatoshi Akita, Mitsunori Miki, Tomoyuki Hiroyasu, and Masato Yoshimi</i>	
RSIE: A Tool Dedicated to Reflexive Systems	363
<i>Yann Barloy, Jean-Marc Nigro, Sophie Loriette, and Baptiste Cable</i>	
A Model for Temperature Prediction of Melted Steel in the Electric Arc Furnace (EAF)	371
<i>Marcin Blachnik, Krystian Mączka, and Tadeusz Wiecek</i>	
Parallel Hybrid Metaheuristics for the Scheduling with Fuzzy Processing Times	379
<i>Wojciech Bożejko, Michał Czapinowski, and Mieczysław Wodecki</i>	
A Neuro-tabu Search Algorithm for the Job Shop Problem	387
<i>Wojciech Bożejko and Mariusz Uchroński</i>	
Parallel Meta ² heuristics for the Flexible Job Shop Problem	395
<i>Wojciech Bożejko, Mariusz Uchroński, and Mieczysław Wodecki</i>	
Particle Swarm Optimization for Container Loading of Nonorthogonal Objects	403
<i>Isaac Cano and Vicenç Torra</i>	

Distributed Control of Illuminance and Color Temperature in Intelligent Lighting System	411
<i>Chitose Tomishima, Mitsunori Miki, Maiko Ashibe, Tomoyuki Hiroyasu, and Masato Yoshimi</i>	
Adaptive Spring Systems for Shape Programming	420
<i>Maja Czoków and Tomasz Schreiber</i>	
Iterated Local Search for de Novo Genomic Sequencing	428
<i>Bernabé Dorronsoro, Pascal Bouvry, and Enrique Alba</i>	
Tournament Searching Method to Feature Selection Problem	437
<i>Grzegorz Dudek</i>	
New Linguistic Hedges in Construction of Interval Type-2 FLS	445
<i>Piotr Dziwiński, Janusz T. Starczewski, and Lukasz Bartczuk</i>	
Construction of Intelligent Lighting System Providing Desired Illuminance Distributions in Actual Office Environment	451
<i>Fumiya Kaku, Mitsunori Miki, Tomoyuki Hiroyasu, Masato Yoshimi, Shingo Tanaka, Takeshi Nishida, Naoto Kida, Masatoshi Akita, Junichi Tanisawa, and Tatsuo Nishimoto</i>	
The Theory of Affinities Applied to the Suppliers' Sustainable Management	461
<i>Anna María Gil Lafuente and Luciano Barcellos de Paula</i>	
Protrace: Effective Recursion Tracing and Debugging Library for Functional Programming Style in Common Lisp	468
<i>Konrad Grzaneek and Andrzej Cader</i>	
Automatic Data Understanding: A Necessity of Intelligent Communication	476
<i>Wladyslaw Homenda</i>	
Memory Usage Reduction in Hough Transform Based Music Tunes Recognition Systems	484
<i>Maciej Hrebień and Józef Korbicz</i>	
CogBox - Combined Artificial Intelligence Methodologies to Achieve a Semi-realistic Agent in Serious Games	492
<i>David Irvine and Mario A. Gongora</i>	
Coupling of Immune Algorithms and Game Theory in Multiobjective Optimization	500
<i>Pawel Jarosz and Tadeusz Burczynski</i>	
Intelligent E-Learning Systems for Evaluation of User's Knowledge and Skills with Efficient Information Processing	508
<i>Wojciech Kacalak, Maciej Majewski, and Jacek M. Zurada</i>	

Interactive Cognitive-Behavioral Decision Making System	516
<i>Zdzisław Kowalczyk and Michał Czubenko</i>	
The Influence of Censoring for the Performance of Survival Tree Ensemble	524
<i>Małgorzata Krętowska</i>	
Clustering Polish Texts with Latent Semantic Analysis	532
<i>Marcin Kuta and Jacek Kitowski</i>	
Hybrid Immune Algorithm for Many Optima	540
<i>Małgorzata Lucińska</i>	
Combining ESOMs Trained on a Hierarchy of Feature Subsets for Single-Trial Decoding of LFP Responses in Monkey Area V4	548
<i>Nikolay V. Manyakov, Jonas Poelmans, Rufin Vogels, and Marc M. Van Hulle</i>	
XML Schema and Data Summarization	556
<i>Jakub Marciniak</i>	
Sample-Based Collection and Adjustment Algorithm for Metadata Extraction Parameter of Flexible Format Document	566
<i>Toshiko Matsumoto, Mitsuharu Oba, and Takashi Onoyama</i>	
A New Stochastic Algorithm for Strategy Optimisation in Bayesian Influence Diagrams	574
<i>Michał Matuszak and Tomasz Schreiber</i>	
Forecasting in a Multi-skill Call Centre	582
<i>David Millán-Ruiz, Jorge Pacheco, J. Ignacio Hidalgo, and José L. Vélez</i>	
Identification of Load Parameters for an Elastic-Plastic Beam Basing on Dynamic Characteristics Changes	590
<i>Bartosz Miller, Zenon Waszczyszyn, and Leonard Ziemiański</i>	
Architecture of the HeaRT Hybrid Rule Engine	598
<i>Grzegorz J. Nalepa</i>	
Using Extended Cardinal Direction Calculus in Natural Language Based Systems	606
<i>Jedrzej Osinski</i>	
Metamodelling Approach towards a Disaster Management Decision Support System	614
<i>Siti Hajar Othman and Ghassan Beydoun</i>	
Comparison Judgments in Incomplete Saaty Matrices	622
<i>Henryk Piech and Urszula Bednarska</i>	

Application of an Expert System for Some Logistic Problems	630
<i>Andrzej Pieczyński and Silva Robak</i>	
AI Methods for a Prediction of the Pedagogical Efficiency Factors for Classical and e-Learning System	638
<i>Krzysztof Przybyszewski</i>	
Online Speed Profile Generation for Industrial Machine Tool Based on Neuro-fuzzy Approach	645
<i>Leszek Rutkowski, Andrzej Przybył, Krzysztof Cpałka, and Meng Joo Er</i>	
The Design of an Active Seismic Control System for a Building Using the Particle Swarm Optimization	651
<i>Adam Schmidt and Roman Lewandowski</i>	
The Normalization of the Dempster’s Rule of Combination	659
<i>Pavel Sevastjanov, Pavel Bartosiewicz, and Kamil Tkacz</i>	
CI in General Game Playing - To Date Achievements and Perspectives	667
<i>Karol Walędzik and Jacek Mańdziuk</i>	
Soft Computing Approach to Discrete Transport System Management	675
<i>Tomasz Walkowiak and Jacek Mazurkiewicz</i>	
Crowd Dynamics Modeling in the Light of Proxemic Theories	683
<i>Jarostaw Wąs</i>	
The Use of Psycholinguistics Rules in Case of Creating an Intelligent Chatterbot	689
<i>Stawomir Wiak and Przemysław Kosiorowski</i>	
UMTS Base Station Location Planning with Invasive Weed Optimization	698
<i>Rafał Zdunek and Tomasz Ignor</i>	
Author Index	707

Part I

Fuzzy Systems and Their Applications

On the Distributivity of Fuzzy Implications over Continuous Archimedean Triangular Norms

Michał Baczyński

Institute of Mathematics, University of Silesia,
40-007 Katowice, ul. Bankowa 14, Poland
michal.baczynski@us.edu.pl

Abstract. Recently, we have examined solutions of the following distributive functional equation $I(x, S_1(y, z)) = S_2(I(x, y), I(x, z))$, when S_1, S_2 are continuous Archimedean t-conorms and I is an unknown function [5,3]. Earlier, in [1,2], we have also discussed solutions of the following distributive equation $I(x, T_1(y, z)) = T_2(I(x, y), I(x, z))$, when T_1, T_2 are strict t-norms. In particular, in both cases, we have presented solutions which are fuzzy implications in the sense of Fodor and Roubens. In this paper we continue these investigations for the situation when T_1, T_2 are continuous Archimedean t-norms, thus we give a partial answer for one open problem postulated in [2]. Obtained results are not only theoretical – they can be also useful for the practical problems, since such distributive equations have an important role to play in efficient inferencing in approximate reasoning, especially in fuzzy control systems.

Keywords: Fuzzy connectives; Fuzzy implication; Distributivity Equations; T-norm; Combs Methods.

1 Introduction

Distributivity of fuzzy implications over different fuzzy logic connectives, like t-norms, t-conorms and uninorms, has been studied in the recent past by many authors (see [1,2,5,8,17,18,20]). The importance of such equations in fuzzy logic has been introduced by Combs and Andrews [10], wherein they exploit the following classical tautology

$$(p \wedge q) \rightarrow r \equiv (p \rightarrow r) \vee (q \rightarrow r)$$

in their inference mechanism towards reduction in the complexity of fuzzy “IF-THEN” rules. Subsequently, there were many discussions (see [11,13,16]), most of them pointing out the need for a theoretical investigation required for employing such equations, as concluded by Dick and Kandel [13]: “Future work on this issue will require an examination of the properties of various combinations of fuzzy unions, intersections and implications”.

Trillas and Alsina [20] investigated the generalized version of the above law:

$$I(T(x, y), z) = S(I(x, z), I(y, z)), \quad x, y, z \in [0, 1], \quad (1)$$

where I is a fuzzy implication and T, S are a t-norm and a t-conorm, generalizing the \wedge, \vee operators. From their investigations for three main families of fuzzy implications, it was shown that in the case of R-implications and S-implications, [\(1\)](#) holds if and only if $T = \min$ and $S = \max$. Also, along the above lines, Balasubramaniam and Rao [\[8\]](#) considered the following dual equations of [\(1\)](#):

$$I(S(x, y), z) = T(I(x, z), I(y, z)), \quad (2)$$

$$I(x, T_1(y, z)) = T_2(I(x, y), I(x, z)), \quad (3)$$

$$I(x, S_1(y, z)) = S_2(I(x, y), I(x, z)), \quad (4)$$

where again T, T_1, T_2 and S, S_1, S_2 are t-norms and t-conorms, respectively, and I is a fuzzy implication. Similarly, it was shown that when I is either an R-implication or an S-implication, in almost all the cases the distributivity holds only when $T_1 = T_2 = T = \min$ and $S_1 = S_2 = S = \max$.

Recently, in [\[5,3\]](#), we have examined solutions of [\(4\)](#), when S_1, S_2 are continuous Archimedean t-conorms and I is an unknown function. Between all the solutions we have indicated fuzzy implications. In that paper we have also answered for one open problem from [\[8\]](#). Meanwhile, the author in [\[11,2\]](#) considered the equation [\(3\)](#) along with other equations, and characterized fuzzy implications I in the case when $T_1 = T_2$ is a strict t-norm. In this paper we would like to continue these investigations, i.e., we attempt to solve the problem in a more general setting, by characterizing functions (fuzzy implications) I which satisfy the equation [\(3\)](#) when T_1, T_2 are continuous Archimedean t-norms. This will give a partial answer for the Problem 1 postulated in [\[2\]](#).

Obtained results are not only theoretical – as we mentioned earlier, such distributive equations have an important role to play in inference invariant rule reduction in fuzzy inference systems (see e.g. [\[7,19\]](#)). Such developments connected with solutions of different systems can be also useful in other topics like fuzzy mathematical morphology (see [\[12\]](#)) or similarity measures (cf. [\[9\]](#)). That more recent works dealing with distributivity of fuzzy implications over uni-norms (see [\[17,18\]](#)) have been written is an indication of the sustained interest in the above equations.

2 Basic Notations and Facts

We suppose the reader to be familiar with the basic theory of t-norms and fuzzy implications (see e.g. [\[14,15\]](#)), hence we recall here only some basic facts.

Definition 2.1 ([\[15, Definitions 2.9 and 2.13\]](#)). *An associative, commutative and increasing operation $T: [0, 1]^2 \rightarrow [0, 1]$ is called a t-norm if it has the neutral element 1. A continuous t-norm T is said to be*

- (i) *Archimedean, if $T(x, x) < x$ for every $x \in (0, 1)$,*
- (ii) *strict, if it is strictly monotone, i.e., $T(x, y) < T(x, z)$ for $x > 0$ and $y < z$,*
- (iii) *nilpotent, if for each $x \in (0, 1)$ there exists $n \in \mathbb{N}$ such that $T(\underbrace{x, \dots, x}_{n \text{ times}}) = 0$.*

Theorem 2.2 (cf. [15, Theorem 5.1]). *For a function $T: [0, 1]^2 \rightarrow [0, 1]$ the following statements are equivalent:*

- (i) T is a continuous Archimedean t -norm.
- (ii) T has a continuous additive generator, i.e., there exists a continuous, strictly decreasing function $t: [0, 1] \rightarrow [0, \infty]$ with $t(1) = 0$, which is uniquely determined up to a positive multiplicative constant, such that

$$T(x, y) = t^{-1}(\min(t(x) + t(y), t(0))), \quad x, y \in [0, 1]. \quad (5)$$

Remark 2.3 (cf. [15, Section 3.2]). T is a strict t -norm if and only if each generator t of T satisfies $t(0) = \infty$. T is a nilpotent t -norm if and only if each generator t of T satisfies $t(0) < \infty$.

In this paper we are interested in finding solutions of (3) for continuous Archimedean t -norms, so it is enough to consider the following 4 cases:

- both t -norms T_1, T_2 are strict,
- both t -norms T_1, T_2 are nilpotent,
- t -norm T_1 is strict and t -norm T_2 is nilpotent,
- t -norm T_1 is nilpotent and t -norm T_2 is strict.

Finally, we will use the following definition of fuzzy implications, which is equivalent to the definition used by Fodor and Roubens [14, Definition 1.15].

Definition 2.4. *A function $I: [0, 1]^2 \rightarrow [0, 1]$ is called a fuzzy implication if it is decreasing in the first variable, it is increasing in the second variable and it satisfies the following conditions:*

$$I(0, 0) = 1, \quad I(1, 1) = 1, \quad I(1, 0) = 0. \quad (13)$$

3 Eq. (3) When T_1, T_2 Are Strict

The case when both t -norms T_1, T_2 in (3) are strict has been considered in details by the author in [12], so we only mention this situation. In these papers we have presented solutions in the terms of increasing bijections $\varphi: [0, 1] \rightarrow [0, 1]$. These solutions can be translated to additive generators, putting $t(x) = -\ln \varphi(x)$. The main results in this case can be also obtained directly by using solutions of the Cauchy additive functional equation for $f: [0, \infty] \rightarrow [0, \infty]$. One possible solution has the following form

$$I(x, y) = t_2^{-1}(c_x t_1(y)), \quad x, y \in [0, 1],$$

where t_1, t_2 are continuous, strictly decreasing functions from $[0, 1]$ onto $[0, \infty]$, with $t_1(1) = t_2(1) = 0$, $t_1(0) = t_2(0) = \infty$ and $c_x \in (0, \infty)$ is a certain constant. In the special case, when $t_1 = t_2 = t$ and $c_x = x$ we get the following family of fuzzy implications which are continuous except at one point $(0, 0)$:

$$I(x, y) = t^{-1}(xt(y)), \quad x, y \in [0, 1],$$

with the convention that $0 \cdot \infty = 0$. One can easily see that these are well developed f -generated fuzzy implications used by Yager (see [21]). Therefore this yet another positive point for using them in the practical applications.

4 Eq. (3) When T_1, T_2 Are Nilpotent

Our main goal in this section is to present the representations of some classes of fuzzy implications that satisfy equation (3) when both t-norms T_1, T_2 are nilpotent. Within this context, we firstly describe the general solutions of (3) with the above assumption. From this result we will see that there are no continuous fuzzy implications I that are solutions for (3) for nilpotent t-norms.

Theorem 4.1. *Let T_1, T_2 be nilpotent t-norms. For a function $I: [0, 1]^2 \rightarrow [0, 1]$ the following statements are equivalent:*

- (i) *A triple of functions T_1, T_2, I satisfies the equation (3) for all $x, y, z \in [0, 1]$.*
- (ii) *There exist continuous and strictly decreasing functions $t_1, t_2: [0, 1] \rightarrow [0, \infty]$ with $t_1(1) = t_2(1) = 0$, $t_1(0) < \infty$ and $t_2(0) < \infty$, which are uniquely determined up to positive multiplicative constants, such that T_1, T_2 admit the representation (5) with t_1, t_2 , respectively, and for every fixed $x \in [0, 1]$, the vertical section $I(x, \cdot)$ has one of the following forms:*

$$I(x, y) = 0, \quad y \in [0, 1], \quad (6)$$

$$I(x, y) = 1, \quad y \in [0, 1], \quad (7)$$

$$I(x, y) = \begin{cases} 0, & \text{if } y = 0, \\ 1, & \text{if } y > 0, \end{cases} \quad y \in [0, 1], \quad (8)$$

$$I(x, y) = t_2^{-1}(\min(c_x t_1(y), t_2(0))), \quad y \in [0, 1], \quad (9)$$

with a certain $c_x \in \left[\frac{t_2(0)}{t_1(0)}, \infty \right)$.

Proof. (ii) \implies (i) This proof can be checked directly.

(i) \implies (ii) Let us assume that functions T_1, T_2 and I are solutions of (3) satisfying the required properties. From Theorem 2.2 and Remark 2.3 the t-norms T_1, T_2 admit the representation (5) for some continuous additive generators $t_1, t_2: [0, 1] \rightarrow [0, \infty]$ such that $t_1(1) = t_2(1) = 0$, $t_1(0) < \infty$ and $t_2(0) < \infty$. Moreover, both generators are uniquely determined up to positive multiplicative constants. Now, (3) becomes, for all $x, y, z \in [0, 1]$,

$$I(x, t_1^{-1}(\min(t_1(y) + t_1(z)), t_1(0))) = t_2^{-1}(\min(t_2(I(x, y)) + t_2(I(x, z)), t_2(0))).$$

Let $x \in [0, 1]$ be arbitrarily fixed and define a function $I_x: [0, 1] \rightarrow [0, 1]$ by $I_x(y) = I(x, y)$ for $y \in [0, 1]$. By substitutions, $h_x = t_2 \circ I_x \circ t_1^{-1}$, $t_1(0) = a$, $t_2(0) = b$, $u = t_1(y)$, $v = t_1(z)$ for $y, z \in [0, 1]$, from the above we obtain

$$h_x(\min(u + v, a)) = \min(h_x(u) + h_x(v), b), \quad u, v \in [0, a],$$

where $h_x: [0, a] \rightarrow [0, b]$ is an unknown function. By [5, Proposition 3] we get all possible formulas for h_x , and, consequently, all possible formulas for $I(x, \cdot)$. \square

From the above theorem we can present an infinite number of solutions that are fuzzy implications. It should be noted that, with this assumption, the vertical section for $x = 0$ should be (7), while the vertical section (6) is not possible.

Example 4.2. (i) If T_1, T_2 are both nilpotent t-norms, then the least solution of (3) which is a fuzzy implication is the following

$$I(x, y) = \begin{cases} 1, & \text{if } x = 0, \\ t_2^{-1} \left(\min \left(\frac{t_2(0)}{t_1(0)} t_1(y), t_2(0) \right) \right), & \text{if } x > 0, \end{cases} \quad x, y \in [0, 1].$$

When $t_1 = t_2$ we get the least (S,N)-implication (see [4] Example 1.5):

$$I(x, y) = \begin{cases} 1, & \text{if } x = 0, \\ y, & \text{if } x > 0, \end{cases} \quad x, y \in [0, 1].$$

(ii) If T_1, T_2 are both nilpotent t-norms, then the greatest solution of (3) which is a fuzzy implication is the greatest fuzzy implication I_1 (see [6]):

$$I_1(x, y) = \begin{cases} 0, & \text{if } x = 1 \text{ and } y = 0, \\ 1, & \text{otherwise,} \end{cases} \quad x, y \in [0, 1].$$

One can easily check that if T_1, T_2 are nilpotent t-norms, then there are no continuous solutions I of (3) which satisfy (13). Also, there are no solutions which are fuzzy implications non-continuous only at $(0, 0)$ (cf. [5] Corollary 8]).

5 Eq. (3) When T_1 Is Strict and T_2 Is Nilpotent

In this section we repeat investigations presented in the previous one, but with the assumption that T_1 is a strict t-norm and T_2 is a nilpotent t-norm.

Theorem 5.1. *Let T_1 be a strict t-norm and T_2 be a nilpotent t-norm. For a function $I: [0, 1]^2 \rightarrow [0, 1]$ the following statements are equivalent:*

- (i) *A triple of functions T_1, T_2, I satisfies the equation (3) for all $x, y, z \in [0, 1]$.*
- (ii) *There exist continuous and strictly increasing functions $t_1, t_2: [0, 1] \rightarrow [0, \infty]$ with $t_1(1) = t_2(1) = 0$, $t_1(0) = \infty$ and $t_2(0) < \infty$, which are uniquely determined up to positive multiplicative constants, such that T_1, T_2 admit the representation (5) with t_1, t_2 , respectively, and for every fixed $x \in [0, 1]$, the vertical section $I(x, \cdot)$ has one of the following forms:*

$$I(x, y) = 0, \quad y \in [0, 1], \quad (10)$$

$$I(x, y) = 1, \quad y \in [0, 1], \quad (11)$$

$$I(x, y) = \begin{cases} 0, & \text{if } y = 0, \\ 1, & \text{if } y > 0, \end{cases} \quad y \in [0, 1], \quad (12)$$

$$I(x, y) = \begin{cases} 0, & \text{if } y < 1, \\ 1, & \text{if } y = 1, \end{cases} \quad y \in [0, 1], \quad (13)$$

$$I(x, y) = t_2^{-1} \left(\min(c_x t_1(y), t_2(0)) \right), \quad y \in [0, 1], \quad (14)$$

with a certain $c_x \in (0, \infty)$.

Proof. (ii) \implies (i) This proof can be checked directly.

(i) \implies (ii) Using similar reasoning like in the proof of Theorem 4.1 we obtain, for a fixed $x \in [0, 1]$, the following functional equation

$$h_x(u + v) = \min(h_x(u) + h_x(v), b), \quad u, v \in [0, \infty],$$

where $h_x: [0, \infty] \rightarrow [0, b]$. By [3] Proposition 3.4] we get all possible formulas for h_x , and, consequently, all possible formulas for $I(x, \cdot)$. \square

Since we are interested in finding solutions of (3) in the fuzzy logic context, we can easily obtain an infinite number of solutions which are fuzzy implications. It should be noted that with this assumption the vertical section (10) is not possible, while for $x = 0$ the vertical section should be (11).

Example 5.2

- (i) If T_1 is a strict t-norm and T_2 is a nilpotent t-norm, then the least implication which satisfies (3) is the least fuzzy implication I_0 (see [6]):

$$I_0(x, y) = \begin{cases} 1, & \text{if } x = 0 \text{ or } y = 1, \\ 0, & \text{if } x > 0 \text{ and } y < 1, \end{cases} \quad x, y \in [0, 1].$$

- (ii) If T_1 is a strict t-norm and T_2 is a nilpotent t-norm, then the greatest solution of (3) which is a fuzzy implication is the greatest fuzzy implication I_1 .

Now, we are in a position to describe the continuous solutions of (3). In fact we have three possibilities: either $I = 0$, or $I = 1$, or there exists a unique continuous function $c: [0, 1] \rightarrow (0, \infty)$, such that I has the form

$$I(x, y) = t_2^{-1}(\min(c(x)t_1(y), t_2(0))), \quad x, y \in [0, 1].$$

From this fact we get

Corollary 5.3. *If T_1 is a strict t-norm and T_2 is a nilpotent t-norm, then there are no continuous solutions I of (3) which satisfy (13).*

Therefore it is obvious that we need to look for solutions which are non-continuous at $(0, 0)$. In this case, opposite to the previous section, the answer is positive and using similar methods as earlier we can prove the following fact.

Theorem 5.4. *Let T_1 be a strict t-norm and T_2 be a nilpotent t-norm and $I: [0, 1]^2 \rightarrow [0, 1]$ be a fuzzy implication continuous except at the point $(0, 0)$. Then the following statements are equivalent:*

- (i) *A triple of functions T_1, T_2, I satisfies the equation (3) for all $x, y, z \in [0, 1]$.*
(ii) *There exist continuous and strictly increasing functions $t_1, t_2: [0, 1] \rightarrow [0, \infty]$ with $t_1(1) = t_2(1) = 0$, $t_1(0) = \infty$ and $t_2(0) < \infty$, which are uniquely determined up to positive multiplicative constants, such that T_1, T_2 admit the representation (5) with t_1, t_2 , respectively, and a unique continuous increasing function $c: [0, 1] \rightarrow [0, \infty)$ with $c(x) > 0$ for $x \in (0, 1]$, $c(0) = \infty$, such that I has the form*

$$I(x, y) = \begin{cases} 1, & \text{if } x = y = 0, \\ t_2^{-1}(\min(c(x)t_1(y), t_2(0))), & \text{otherwise,} \end{cases} \quad x, y \in [0, 1].$$

Example 5.5. Let us assume that $t_1(x) = -\ln x$, $t_2(x) = 1 - x$ and $c(x) = x$. Then we get the following fuzzy implication, which satisfies distributive equation with the product t-norm as T_1 and Łukasiewicz t-norm as T_2 :

$$I(x, y) = \begin{cases} 1, & \text{if } x = y = 0, \\ \max(1 + x \ln y, 0), & \text{otherwise,} \end{cases} \quad x, y \in [0, 1].$$

6 Eq. (3) When T_1 Is Nilpotent and T_2 Is Strict

In this section we repeat investigations presented in the previous one, but with the assumption that T_1 is a nilpotent t-norm and T_2 is a strict t-norm.

Theorem 6.1. *Let T_1 be a nilpotent t-norm and T_2 be a strict t-norm. For a function $I: [0, 1]^2 \rightarrow [0, 1]$ the following statements are equivalent:*

- (i) *A triple of functions T_1, T_2, I satisfies the equation (3) for all $x, y, z \in [0, 1]$.*
- (ii) *There exist continuous and strictly increasing functions $t_1, t_2: [0, 1] \rightarrow [0, \infty]$ with $t_1(1) = t_2(1) = 0$, $t_1(0) < \infty$ and $t_2(0) = \infty$, which are uniquely determined up to positive multiplicative constants, such that T_1, T_2 admit the representation (5) with t_1, t_2 , respectively, and for every fixed $x \in [0, 1]$, the vertical section $I(x, \cdot)$ has one of the following forms:*

$$\begin{aligned} I(x, y) &= 0, & y \in [0, 1], \\ I(x, y) &= 1, & y \in [0, 1], \\ I(x, y) &= \begin{cases} 0, & \text{if } y = 0, \\ 1, & \text{if } y > 0, \end{cases} & y \in [0, 1]. \end{aligned}$$

Proof. (ii) \implies (i) This proof can be checked directly.

(i) \implies (ii) Using similar reasoning like in the proof of Theorem 4.1 we obtain, for a fixed $x \in [0, 1]$, the following functional equation

$$h_x(\min(u + v, a)) = h_x(u) + h_x(v), \quad u, v \in [0, a],$$

where $h_x: [0, a] \rightarrow [0, \infty]$. By [3] Proposition 3.6] we get all possible formulas for the function h_x , and, consequently, all possible formulas for $I(x, \cdot)$. \square

In this case we have only trivial solutions, i.e., solutions which values are in the set $\{0, 1\}$. In particular only $I = 0$ or $I = 1$ are continuous solutions.

Example 6.2

- (i) If T_1 is a nilpotent t-norm and T_2 is a strict t-norm, then the least solution of (3) which is a fuzzy implication is the following:

$$I(x, y) = \begin{cases} 0, & \text{if } x > 0 \text{ and } y = 0, \\ 1, & \text{otherwise,} \end{cases} \quad x, y \in [0, 1].$$

- (ii) If T_1 is a nilpotent t-norm and T_2 is a strict t-norm, then the greatest solution of (3) which is a fuzzy implication is the greatest fuzzy implication I_1 .

References

1. Baczyński, M.: On a class of distributive fuzzy implications, *Internat. J. Uncertain. Fuzziness Knowledge-Based Systems* 9, 229–238 (2001)
2. Baczyński, M.: Contrapositive symmetry of distributive fuzzy implications, *Internat. J. Uncertain. Fuzziness Knowledge-Based Systems* 10, 135–147 (2002)
3. Baczyński, M.: On the distributivity of fuzzy implications over continuous and Archimedean triangular conorms. *Fuzzy Sets and Systems* 161, 1406–1419 (2010)
4. Baczyński, M., Jayaram, B.: On the characterizations of (S,N) -implications. *Fuzzy Sets and Systems* 158, 1713–1727 (2007)
5. Baczyński, M., Jayaram, B.: On the distributivity of fuzzy implications over nilpotent or strict triangular conorms. *IEEE Trans. Fuzzy Syst.* 17, 590–603 (2009)
6. Baczyński, M., Drewniak, J.: Monotonic fuzzy implication. In: Szczepaniak, P.S., Lisboa, P.J.G., Kacprzyk, J. (eds.) *Fuzzy Systems in Medicine. Studies in Fuzziness and Soft Computing*, vol. 41, pp. 90–111. Physica-Verlag, Heidelberg (2000)
7. Balasubramaniam, J., Rao, C.J.M.: R-implication operators and rule reduction in Mamdani-type fuzzy systems. In: *Proc. 6th Joint Conf. Information Sciences, Fuzzy Theory, Technology*, Durham, USA, pp. 82–84 (2002)
8. Balasubramaniam, J., Rao, C.J.M.: On the distributivity of implication operators over T- and S-norms. *IEEE Trans. Fuzzy Syst.* 12, 194–198 (2004)
9. Bustince, H., Pagola, M., Barrenechea, E.: Construction of fuzzy indices from fuzzy DI-subsethood measures: Application to the global comparison of images. *Inform. Sci.* 177, 906–929 (2007)
10. Combs, W.E., Andrews, J.E.: Combinatorial rule explosion eliminated by a fuzzy rule configuration. *IEEE Trans. Fuzzy Syst.* 6, 1–11 (1998)
11. Combs, W.E.: Author's reply. *IEEE Trans. Fuzzy Syst.* 7, 371, 478–479 (1999)
12. De Baets, B., Kerre, E., Gupta, M.: The fundamentals of fuzzy mathematical morphology, Part 1: Basic concepts. *Int. J. Gen. Syst.* 23, 155–171 (1994)
13. Dick, S., Kandel, A.: Comments on Combinatorial rule explosion eliminated by a fuzzy rule configuration. *IEEE Trans. Fuzzy Syst.* 7, 475–477 (1999)
14. Fodor, J., Roubens, M.: *Fuzzy Preference Modelling and Multicriteria Decision Support*. Kluwer, Dordrecht (1994)
15. Klement, E.P., Mesiar, R., Pap, E.: *Triangular Norms*. Kluwer, Dordrecht (2000)
16. Mendel, J.M., Liang, Q.: Comments on Combinatorial rule explosion eliminated by a fuzzy rule configuration. *IEEE Trans. Fuzzy Syst.* 7, 369–371 (1999)
17. Ruiz-Aguilera, D., Torrens, J.: Distributivity of strong implications over conjunctive and disjunctive uninorms. *Kybernetika* 42, 319–336 (2005)
18. Ruiz-Aguilera, D., Torrens, J.: Distributivity of residual implications over conjunctive and disjunctive uninorms. *Fuzzy Sets and Systems* 158, 23–37 (2007)
19. Sokhansanj, B.A., Rodrigue, G.H., Fitch, J.P.: Applying URC fuzzy logic to model complex biological systems in the language of biologists. In: *2nd Int. Conf. Systems Biology (ICSB 2001)*, Pasadena, USA, p. 102 (2001)
20. Trillas E., Alsina C.: On the law $[p \wedge q \rightarrow r] = [(p \rightarrow r) \vee (q \rightarrow r)]$ in fuzzy logic. *IEEE Trans. Fuzzy Syst.* 10, 84–88 (2002)
21. Yager, R.R.: On some new classes of implication operators and their role in approximate reasoning. *Inform. Sci.* 167, 193–216 (2004)

Fuzzy Decision Support System for Post-Mining Regions Restoration Designing

Marzena Bielecka¹ and Jadwiga Król-Korczak²

¹ Chair of Geoinformatics and Applied Computer Science,
Faculty of Geology, Geophysics and Environmental Protection,
University of Science and Technology,
Al. Mickiewicza 30, 30-059 Cracow, Poland

² State Mining Authority,
District Mining Office in Cracow,
Lubicz 25, 31-503 Cracow, Poland
bielecka@agh.edu.pl, krolkorczak@gmail.com

Abstract. Reclamation as one of the stages in the cycle of life of mine is realized using different techniques and technology, adapted to the unique characteristics for any given mining institution. Restoration of terrain from opencast mining is influenced by many factors and processes and the results are open to interpretation and are not predictable. Most of the mentioned factors has qualitative character. The number and complex connections among these factors cause the fact that the analysis of post-mining terrain restoration is expensive and time-consuming. Therefore the automatization of the decision making is very desirable. In this paper a fuzzy decision support system for post-mining regions restoration designing is proposed. The system was applied to testing decision making concerning revitalization direction in opencast mining institution in Zator community, southern Poland.

1 Introduction

The mining of mineral materials affects the environment in special ways degrading or destroying the scenery, altering and changing the terrain, removing considerable areas of agricultural soil and forest. The opencast mines, also known as open-pit mining, open-cut mining and strip mining, excavates the soil to exploit the natural resources giving birth to the problem of redevelopment or restoration of the excavations when the exploitation is over.

In Poland the duty of reclamation and redevelopment of the terrain, which lost usable value, are defined in two executive laws; the act of 4th February 1994 The Geological and Mining Law (J.L. No 27, item 96 from changes later), and the act of 3rd February 1995 on The Protection of Agricultural and Forest Land (J.L. No 16, item 78). In the case of reclamation, the person or institution responsible for the destruction of the environment is obliged to obtain agreement on the restoration required and perform the repairs. There is no detailed legal description of the details for redevelopment except for the definition. Currently

the definition for executing basic reclamation is to return the terrain to the same use as the local surrounding terrain.

One must pay attention to the complexity and the many alternative revitalization processes that characterize post-mining terrain restoration. The number and complex connections among these factors cause the fact that the analysis of post-mining terrain restoration is expensive and time-consuming. Therefore the automatization of the decision making is very desirable. Since some factors characterizing excavation sites has qualitative character and can not be expressed numerically, the fuzzy rule system is suitable as a decision support system (6). Therefore this sort of system is proposed for post-mining aiding regions restoration designing - see section 3. The system was applied to testing decision making concerning revitalization direction in opencast mining institution in Zator community, southern Poland - section 4.

2 Arrangement of Factors Characterizing the Terrain after Exploitation of the Natural Resources

The complex studies for the range of classification of factors characterizing the terrains of exploitation of natural resources, were prepared based on accessible data, as well as literature data. Often there is lack of legal information about the terrain, its past ownership and historical data connected to possible exploitation on the terrain. However shortage of some data is not an obstacle to preparing a characteristic of such terrains in individual post-mining areas. Introduced systematic incline to context of the project subject to choice of optimum direction of development, reflecting first of all the economic factor, which will determine the most rational prognosis of expenses for a mining businessman. In this case, the spatial factors, environmental, hydro geological, social, legal, geological-engineering will be perceived as extremely essential, but in general picture will be treated as economical.

Economic factors

Corrected estimation liquidation costs and then reclamation and development post-mining regions, is extremely complicated, time consuming, and extensive. Based on that assumption, we accept that the economic calculation was finished, but it won't be included in the project. As it was mentioned before, the economic factor will be the main criteria in decision making process regarding the choice of the optimum form of revitalization. This means, that including general tendencies of the remaining factors for the most optimum (the cheapest) liquidation process will be useful in projecting expenses and verifying them from economic point of view. A businessman will have specified and systematized, both the factors characterizing a given post-exploitation area, and the variant conceptions.

Social factors

The post exploitation area of natural resources is often not suitable for commercial use. However, it can be utilized for leisure by local residents. Particularly,

submerged excavations can be adapted for use as water sports recreation areas. Therefore, the proper utilization and accessibility of developed terrains can attract more users. The number of potential users depends on proposed activities and different leisure preferences. For example, a visitor to an area of post-mining exploitation might use the terrain in a number of ways: as a swimming pool, for walks, picnics, and cycling if there are bicycle paths. The interests of the local community are influencing factors for the development, for example, an expressed interest in recreation or fishing.

Based on specification of certain post-exploitation terrains of mineral materials, some local communities might show interest in entrepreneurship and handing down from generation to generation farm commercial process traditions, for example fish farming. That kind of enterprise with specific industry and local connections is based on local traditions, and knowledge about what to produce and how to produce, as well as craftsman's skills. Assuming that the local community is the main potential user of developed terrain, it is important to analyze demographic data, unemployment levels and local interests.

Legal factors

Legal factors determine the form of exploitation and often concur with environmental factors. However, companies are under increasing pressure to produce ongoing legal documentation, concerning their economic utilization of mineral deposits and protection of land resources during the process of liquidation of a mining institution. Therefore, it will become necessary to reflect this in data collection.

Environmental factors

The environmental changes happened in places of excavation, which submerged on its own as a result of exploitation or removal of mineral deposits. The terrains not transformed by mining activity can be used as agricultural soils, meadows and pastures. Threats to the environment come from the possibility of geodynamic phenomena - the loss of stability of slopes in deposit and in baring because of loose character of the deposit, washout of slopes of exploitation pools as well as sinking slopes caused by rain waters. In the case of water terrains it is possible an inception of water fowl nesting as well as occurrence of amphibians.

As mentioned above the preferences of reclamation and the different forms of natural resource development for post exploitation should taken into consideration a protection of the valuable plants, animals and curiosities characteristic for the given countryside, for example geological ones.

Spatial factors - the location

The terrain location, bus and railway communication possibilities, distance from the big cities are crucial factors influenced the possibilities of the region development. The picturesque aspects can make the considered countryside suitable for agro-tourism development. Existence of forests and pure waters can be a good bases for recreational direction of restoration.

Hydrological factors

The hydrological factors are the essential ones influencing the choice of water, economic, or recreational development, resulting from demand on, for example, the reservoirs of drinking water, interest in angling or agricultural (fish farming) connected to the tradition of farm commercial process.

Geological-engineering factors (technical)

The analysis of geological-engineering factors make possible the diagnosis of these geological factors, which determine optimum utilization of engineering techniques and the minimization of the negative influence of investments on geological environment which is the cost of liquidation of the mining institution. In characterizing the geological-engineering (technical) factors, the parameters of excavations should be mainly taken under attention. The depth of excavation, to maximum 15 meters makes a possibility for fish farming. Correlations with interest in fishing and required inclination of the slopes of excavation determines creation of a water recreational direction.

Cultural factors

The existence of objects of technique and material culture, described above in Table 1, can be a basis for creating thematic routes, museums, in particular industrial ones, and heritage parks. Furthermore, locations such as places of martyrdom, extermination camps and tombs can be destinations for educational trips.

The described factors collectively constitute a vector of data described by quantitative, qualitative, linguistic variables. Sometimes, possibilities of some data gaining may be problematic.

3 Fuzzy Rules System

In the implemented fuzzy rules system the following variables were used as premises in inference rules.

- (a) Population is described by two fuzzy sets: small and large.
- (b) Interest in recreation is described by two fuzzy sets: small and large.
- (c) Level of unemployment is described by two fuzzy sets: small and large.
- (d) Traditional economic activity is described by two fuzzy sets: small and large.
- (e) Enterprise is described by two fuzzy sets: small and large.
- (f) Possibility of introduction of protected plants and animals is described by a singleton.
- (g) Water purity class is described by three fuzzy sets: I, II and III.
- (h) Slope gradient of excavation is described by two fuzzy sets: small and large.
- (i) Depth of excavation is described by three fuzzy sets: very small, small and large.
- (j) Interest in angling is described by two fuzzy sets: small and large.
- (k) The quality of public transport is described by two fuzzy sets: good and bad.

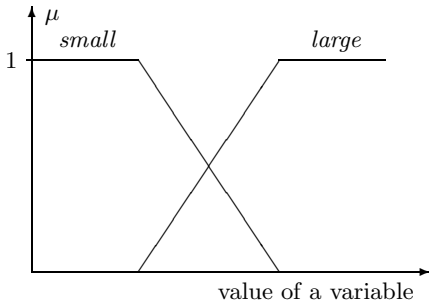
In fuzzy inference modules a few types of fuzzy sets has been used. Variables described in points (a)-(e), (h)-(j) and o are characterized by fuzzy sets small and large which membership function are shown in Fig.1a. In the case of the point (g) three classes of water purity (class I, II and III) has been used - see Fig.1b. In the case of the point (i) three fuzzy sets has been used for variable description: very small depth of open cast(VS), small (S) and large (L) see Fig.1c. In the case of points (f) singletons have been used as a membership function. However, in implementations done in MATLAB, very narrow triangular functions have been used because singletons are no allowed in this software environment.

Defuzzification was performed in the following way. The output possibilities were encoded as triangular fuzzy sets with maxima in natural numbers representing the output possibilities - see Fig.1d. After calculating the output fuzzy set, the value of x-coordinate of its membership function maximum, say x_0 , is checked. Then, there was chosen this output possibility which is encoded by the number nearest to x_0 . If the calculated membership function of the output fuzzy set has a few maxima in which its values are equal to one then the alternative concerning all pointed solutions is put as the system answer.

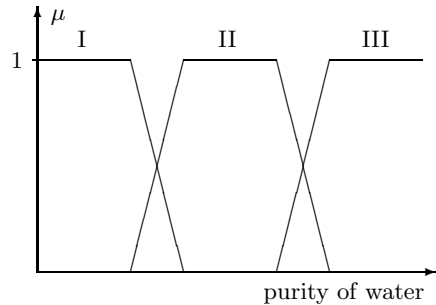
The fuzzy rule system consists of the following nineteen if-then rules.

1. If the level of population is large and the interest in recreation likewise, then mode of the revitalisation is recreational water (lido).
2. If the interest in fishing is large then mode of the revitalisation is recreational water (angling).
3. If the level of unemployment is large and the large traditional farm commercial process likewise then mode of the revitalisation is agricultural (fish farming).
4. If the level unemployment is large and the tradition farm commercial process is small then mode of the revitalisation is recreational (lido).
5. If the level of enterprise is large and there is large interest in recreation then mode of the revitalisation is recreational (lido).
6. If the level of enterprise is large and small interest in recreation then mode of the revitalisation is agricultural (fish farming).
7. If the level of enterprise is small and large interest in an angling then mode of the revitalisation is recreational (fishery).
8. If the level of enterprise is small and interest in recreation is small then mode of the revitalisation is water-forest recreational park.
9. If the level of enterprise is small and there is a possibility of introduction protected plants and animals then mode of the revitalisation is natural - nature reserve.
10. If water purity is first class and there is large traditional farm commercial process then mode of the revitalisation is agricultural (salmon fish farming).
11. If water purity is second class and there is large traditional farm commercial process then mode of the revitalisation is agricultural (fish farming).
12. If there is lack of traditions and large interest in an angling then mode of the revitalisation is recreational (fishery).
13. If there is a small depth of excavation then mode of the revitalisation is agricultural (fish farming).

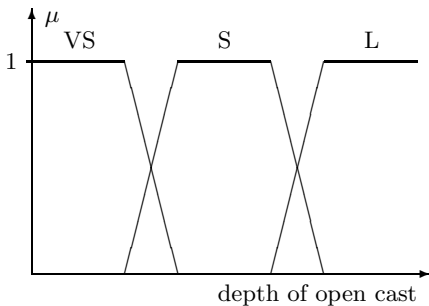
14. If there is a very small depth of excavation and small slope of open cast then mode of the revitalisation is recreational (lido).
15. If there is large depth of excavation and large interest in an angling then mode of the revitalisation is recreational (fishery).
16. If there is large depth of excavation and small interest in an angling then mode of the revitalisation is water/forest (recreational-park).
17. If there is small depth of excavation and the slope gradient of excavation is small then mode of the revitalisation is water-forest (recreational-park).
18. If there is bad public transport and lack of interest in recreation then mode of the revitalisation is natural (nature reserve) or water/forest (recreational-park).
19. If there is good public transport and large interest in recreation then mode of the revitalisation is water recreational (lido).



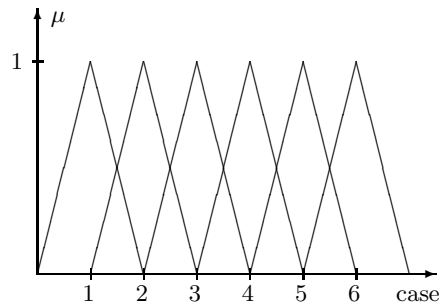
a) The membership functions of fuzzy sets *small* and *large*.



b) The membership functions of water purity.



c) The membership functions of depth of open cast.



d) The membership functions for output values.

Fig. 1. The membership functions

4 Applications and Results

The introduced inference system is applied to make a decision of revitalization direction in opencast mining institution in Zator located on the area of Zator community, southern Poland. The position of a commune 46 km west from Cracow and 48 km from Katowice provide good communication accessibility. Good public communication (railway and buses), made Zator a potential region for development. This is an attractive location, especially for a weekend getaway for occupants from Silesia and Cracow. Zator deposit is a partially waterlogged deposit. In turn Zakole deposit is almost entirely waterlogged. The state of water cleanness is between first and second class. The result is that in context with already existing attractiveness of the Zator area as well as the "Frog's country", and the existing fish ponds, this is the preferred way of development, for which essential characteristics are such as: recreation and protection of natural values. The surface of the Zator community is about 52 square kilometers. The Zator community consists of 20 per cent fishing ponds, and approximately 33 per cent of the remaining general surface is forest terrain. The area formation is represented by forest-water-ornithological reserves, as well as Nature 2000, and the White Park. The structure of the ponds is determined in large part by the fishing economy. This economic determination is reflected in the unusual biodiversity, as manifested by the occurrence of rare and endangered bird's species (the *Gorsachius leuconotus*). The adjoining gravel pits and post mining terrains increase the variety of nesting and feeding sites. The whole nesting system creates an attractive spot for ornithology lovers. The thick net of roads is suitable for the bicycle tourism. The above description implies that post mining area is waterlogged with the first class of the water purity.

The following values of input variables were assumed.

- The level of population is 175 where the interval of values is (90; 216).
- The level of unemployment is 10,6 per cent; the values interval is (9;14.3).
- The enterprise is 66 where the interval of values is (0;94).
- There is the first class of water purity.
- The depth of excavation is 8,8 where the interval of values is (1;20).
- The slope gradient is about 30 degree; the values interval is over 5 degree.
- There is a possibility of introduction protected plants and animals.
- The public transport is good where the interval of values is (0h;2h).

Values of the rest of variables can not be deduced from accessible information concerning Zator community. The following variants were assumed.

The first case: The interest in recreation is 3.17 where the interval of values is (1;10). The tradition farm commercial process is 6.61 where the interval of values is (1;10). The interest in an angling is 7.22 where the interval of values is (1;10). In this case a water body for angling or salmon fish farm has been obtained as a result of the fuzzy interference.

The second case: The interest in recreation is 3.17, the tradition farm commercial process is 6.61 and the interest in an angling is 2.78. In this case a

water-forest recreational park or salmon fish farm has been obtained as a result of the fuzzy interference.

The results obtained from the implemented hybrid inference system coincide with the direction of the applied restoration in Zator community. Generally, the water development was utilized.

5 Concluding Remarks

The exact analysis and profile protection of the environment factors should account various aspects including engineering and economic conditions, cultural values, traditions and expectations of the society. The practical basis for reclamation must consider both the economic side and be socially acceptable. The number and complex connections among the mentioned factors cause the fact that the analysis of post-mining terrain restoration is expensive and time-consuming. Therefore the automatization of the decision making is very desirable. Though computer methods were used as a decision support system in a post mining restoration possibilities analysis ([1][2][3][4]), it seems that artificial intelligence systems have not been used in such context so far. The described fuzzy system was implemented as Mamdani-like systems, in which defuzzification algorithm was a little different than those used in classical Mamdani systems ([5]). The introduced inference system was applied to testing decision making concerning revitalization direction in opencast mining institution in Zator community, southern Poland. It turned out that the system postulated the same solutions as those implemented earlier in the mentioned institution, positively verified in practice.

References

1. Evans, K.G., Willgoose, G.R.: Post-mining landform evolution modelling: Effects of vegetation and surface ripping. *Earth Surface Processes and Landforms* 25, 803–823 (2000)
2. Evans, K.G., Willgoose, G.R., Saynor, M.J., Riley, S.J.: Post-mining landform evolution modelling: Derivation of sediment transport model and rainfall-runoff model parameters. *Earth Surface Processes and Landforms* 25, 743–763 (2000)
3. Hancock, G.R.: The use of landscape evolution models in mining rehabilitation design. *Environmental Geology* 46, 561–573 (2004)
4. Hancock, G.R., Willgoose, G.R., Evans, K.G., Moliere, D.R., Saynor, M.J.: Medium term erosion simulation of a abandoned mine site using the SIBERIA landscape evolution model. *Australian Journal of Soil Research* 38, 249–263 (2000)
5. Jang, J.S.R., Sun, C.T., Mizutani, E.: Neuro-Fuzzy and Soft Computing. In: *A Computational Approach to Learning and Machine Intelligence*. Simon and Schuster, London (1997)
6. Rutkowska, D., Rutkowski, L.: Fuzzy and fuzzy-neural systems. In: Duch, W., Korbicz, J., Rutkowski, L., Tadeusiewicz, R. (eds.) *Academic Printing House EXIT, Warsaw*, pp. 135–178 (2000) (in Polish)

Fuzzy Digital Filters with Triangular Norms

Bohdan S. Butkiewicz

Warsaw University of Technology,
Nowowiejska 15/19, 00-665 Warsaw, Poland
b.butkiewicz@ii.pw.edu.pl
<http://www.ise.pw.edu.pl/~bb/index.html>

Abstract. The convolution operation is used to describe the relation between input and output in the linear filters theory. In this paper the product operation in discrete convolution is replaced by triangular norm operation. It is an extension of conventional idea of convolution. Such approach leads to nonlinear filtering. Some interesting properties as identical impulse, step, and frequency responses of digital filters with T -norms and conorms (S -norms) operations are shown. Moreover, filters with fuzzy parameters and crisp signals passing by such fuzzy nonlinear filters are investigated.

1 Introduction

The convolution is inseparably related to a concept of linear filters. Mathematically, the convolution operations are inherently a linear combination operation. In this paper, the product in linear combination is replaced by more general operation, triangular norm or conorm [6]. Such idea was firstly presented by Lee *et al.* [7]. They considered different types of discrete convolution where product was replaced by triangular norm or compensatory operation and addition was replaced by triangular conorm or compensatory operation. It leads to many different types of such convolutions. The idea proposed in [7] was not developed later and properties of such filters are not investigated. Only in [1], published unfortunately in Japanese, such median filters are discussed.

In the paper it was shown that digital filters with T -norms and conorms have interesting properties. It was proved that such filters have appropriately identical: impulse, step and frequency responses. It does not mean that responses to different input signals are identical because of the nonlinearity property. The properties are investigated also for such filters with fuzzy parameters.

2 Digital Filters with T -Norms

The response $y[n]$ of conventional discrete linear filter on input signal $x[n]$ equals

$$y[n] = \sum_{m=0}^{N-1} h[m]x[n-m] \quad (1)$$

where n , m denote discrete time instant and appropriately $h[n]$ and $x[m - n]$ values of impulse response and input signal in discrete instants. It was supposed that the filter is causal, i.e. $h[n] \equiv 0$ for $n < 0$, and we take in consideration only N samples of $h[n]$ and $x[n]$, $n = 0, 1, \dots, N - 1$.

Under the symbol Σ , algebraic product of $h[m]$ and $x[n - m]$ is put. The algebraic product is one of the T -norm operations. Generally T -norms are binary operations $[0, 1] \times [0, 1] \rightarrow [0, 1]$, which fulfill conditions of commutativity, associativity, monotonicity, and boundary conditions [6].

Now, consider a situation where algebraic product in (II) is replaced by any T -norm. Discrete impulse response is the response to Kronecker delta

$$\delta[n] = \begin{cases} 1 & \text{if } n = 0 \\ 0 & \text{if } n \neq 0. \end{cases} \quad (2)$$

Lemma 1. *Let $h[n]$ be conventional discrete impulse response of the filter. The impulse response of the filter with any T -norm is equal to the response $h[n]$.*

Proof. The impulse response of the filter with any T -norm will be equal

$$h'[n] = \sum_{m=0}^{N-1} h[m] T \delta[n - m] = h[n] T 1 = h[n]. \quad (3)$$

Thus, all filters with different T -norms have the same impulse response equal to conventional impulse response. From Lemma 1 does not arise that responses to an input signals will be identical for all T -norm filters. Such filters are nonlinear, thus responses can be different.

Discrete step response of a filter is the response to unit step signal

$$1[n] = \begin{cases} 1 & \text{if } n \geq 0 \\ 0 & \text{if } n < 0. \end{cases} \quad (4)$$

Lemma 2. *Let conventional discrete step response of the filter be denoted $k[n]$. The step response of the filter with any T -norm is equal to $k[n]$.*

Proof. The step response of the filter with any T -norm is equal to

$$k'[n] = \sum_{m=0}^{N-1} h[m] T 1[n - m] = \sum_{n=0}^n h[n] T 1 = \sum_{n=0}^n h[n] = k[n]. \quad (5)$$

Thus, all filters with different T -norms have the same step response equal to conventional step response.

Example 1. Let impulse response of the filter $h[n] = \exp(-a\mathcal{Y}n) 1[n]$ and input signal $x[n] = \exp(-b\mathcal{Y}n) 1[n]$. Sampling interval is equal $\mathcal{Y}=0.01$ s, number of samples $N = 100$. Conventional, discrete, output signal equals to

$$y[n] = [\exp(-a\mathcal{Y}(n + 1)) - \exp(-b\mathcal{Y}(n + 1))]/[\exp(-a\mathcal{Y}) - \exp(-b\mathcal{Y})].$$

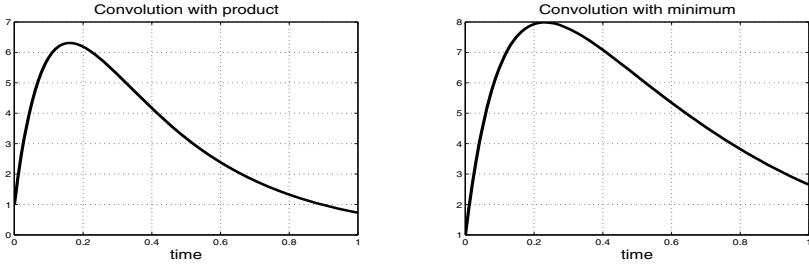


Fig. 1. The output signal of conventional filter (left) and logic product (right)

In Fig. 1 conventional discrete output signal $y[n]$ and output with logic product (minimum) are shown.

The main description of the filter in spectral domain is frequency response. It is Fourier transform of an impulse response. In discrete domain it has the form of sum

$$H[m] = \frac{1}{N} \sum_{n=0}^{N-1} h[n] e^{-j2\pi nm/N} \tag{6}$$

Lemma 3. *Frequency response of any filter with T -norm is identical and equal to conventional frequency response.*

Proof. Impulse responses of the filters with T -norms are identical, thus frequency responses must be identical.

As it was mentioned before, from identical frequency responses it does not arise that for any input signal the amplitude and phase of output signal will be identical for all T -norm filters.

Example 2. Consider the filter in Example 1. Conventional, discrete, amplitude spectrum and spectrum for filter with Hamacher product are shown in Fig. 2.

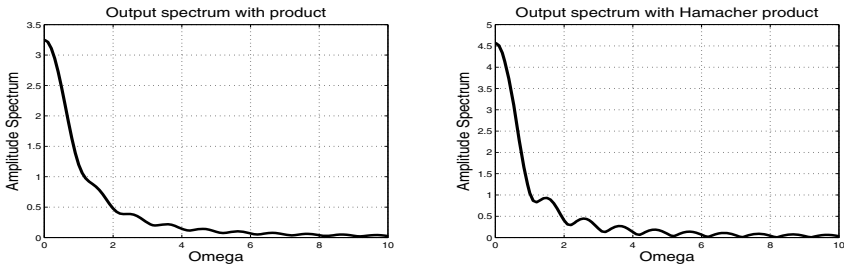


Fig. 2. Frequency responses for algebraic (left) and Hamacher product (right)

Lemma 1-3 can be joined in following theorem.

Theorem 1. *Filters with any T -norm operation have appropriately identical: impulse, step, and frequency responses.*

3 Digital Filters with Conorms (S -Norms)

Algebraic product in discrete convolution sum can be replaced also by other operations. More simple idea is replaced it by triangular conorm [6], frequently, but not quite correctly called S -norms. These S operations substantially differ from previous definition for T -norms by boundary conditions. It must be noted that such approach with S -norms is not generalization of conventional convolution.

Lemma 4. *Impulse response of the filter with any S -norm does not depend on the type of S -norm used in the filter equation.*

Proof. The impulse response of the filter with any S -norm equals

$$h'[n] = \sum_{m=0}^{N-1} h[m] S \delta[n - m] = h[n] S 1 + \sum_{m=0, m \neq n}^{N-1} h[m] S 0 = 1 + \sum_{m=0, m \neq n}^{N-1} h[m]. \tag{7}$$

Thus, all filters with different S -norms have the same impulse response.

Lemma 5. *Step response of the filter with any S -norm does not depend on the type of S -norm used in the filter equation.*

Proof. The step response of the filter with any S -norm equals

$$k'[n] = \sum_{m=0}^{N-1} h[m] S 1[n - m] = n + 1 + \sum_{m=n+1}^{N-1} h[m]. \tag{8}$$

Example 3. Let input signal $x[n] = b\mathcal{Y}n \exp(1 - b\mathcal{Y}n)$. Conventional impulse response of the filter is the same as in Example 1. Output signals for logic sum (maximum) and Yager sum are shown in Fig. 3.

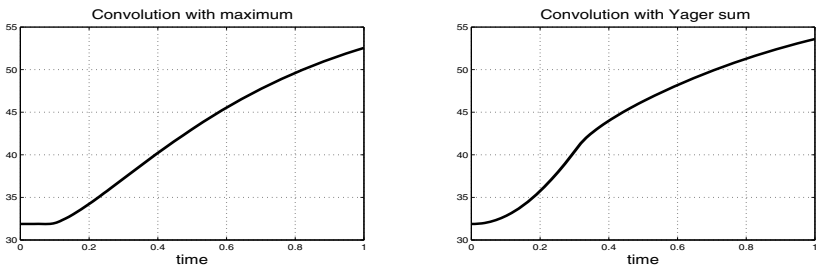


Fig. 3. Output responses of the filter with logic sum (left) and Yager sum (right)

Lemma 6. *Frequency response of the filter with any S -norm does not depend on the type of S -norm used in the filter equation.*

Proof. The frequency response is Fourier transform of impulse response, which are identical for all S -norm filters.

Lemma 4-6 can be gathered in one theorem.

Theorem 2. *Filters with any S -norm operation have appropriately identical: impulse, step, and frequency responses.*

It must be noted that, as for T -norms, output signals and its spectrum can be different for different S -norm filters because of nonlinearity.

4 Fuzzy Filters with Triangular Norms

It is interesting, can the result obtained in section 2 and 3 be generalized for filters with fuzzy parameters? Consider at the beginning filters with T -norms. Let only one filter parameter be fuzzy number α , described by membership function $\mu_\alpha(a)$. The concept of fuzzy convolution was discussed by the author in [2] [3]. It is based on the concept of α -level curves [5]. Symbol \mp and \pm are applied to emphasize possibility of intermingled curves. Similar concept of fuzzy convolution for discrete case, based on discrete α -level curves $h_\alpha^-[m, a]$ $h_\alpha^+[m, a]$, is used here

$$y^\mp[n, a] = \sum_{m=0}^{N-1} x(n-m)h_\alpha^-[m, a] \quad (9)$$

$$y^\pm[n, a] = \sum_{m=0}^{N-1} x(n-m)h_\alpha^+[m, a] \quad (10)$$

where assumption of finite set of samples $x[n]$, and fuzzy samples $\mathbf{h}[n, a]$, $n = 0, 1, \dots, N-1$, are applied. The notation

$$\mathbf{y}[n, \alpha] = \sum_{m=0}^{N-1} x[n-m]\mathbf{h}[m, \alpha] \quad (11)$$

is used for abbreviation.

Lemma 7. *The impulse response of fuzzy filter with any T -norm equals to conventional fuzzy filter with algebraic product.*

Proof. Now, we have fuzzy convolution. Using above concept, impulse response of fuzzy filter with T -norm is calculated as follows

$$h'^\mp[n, a] = \sum_{m=0}^{N-1} h^-[m, a] T \delta[n-m] = \sum_{m=0}^n h^-[m, a] T \delta[n-m] = h^-[n, a] \quad (12)$$

Similarly for $h'^\pm[n, a]$. Then

$$\mathbf{h}'[n, \alpha] = \sum_{m=0}^n \mathbf{h}[m, \alpha] T \delta[n-m] = \mathbf{h}[n, \alpha]. \quad (13)$$

Let values of crisp function $h[n, a]$ be denoted by χ and inversion of function $h[n, a]$ exists, i.e. $a = h^{-1}(\chi)$. Then membership function

$$\mu_h(\chi) = \mu_\alpha[h^{-1}(\chi)]. \tag{14}$$

If number of parameters is greater than one the extension principle, sup – min procedure, must be applied in order to find membership function of $\mathbf{h}[n, \alpha]$.

Lemma 8. *The step response of fuzzy filter with any T-norm equals to step response of fuzzy filter with algebraic product.*

Proof. Using fuzzy convolution concept, the step response of fuzzy filter with T-norm is equal

$$k'^{\mp}[n, a] = \sum_{m=0}^{N-1} h^{-}[m, a] T 1[n - m] = \sum_{m=0}^n h^{-}[m, a] T 1 = \sum_{m=0}^n h^{-}[m, a] = k^{-}[n, a]. \tag{15}$$

Similarly for $k'^{\pm}[n, a]$. Then

$$\mathbf{k}'[n, \alpha] = \sum_{m=0}^n \mathbf{h}[m, \alpha] T 1[n - m] = \mathbf{k}[n, \alpha]. \tag{16}$$

Example 4. Let impulse response of fuzzy filter equals $h[n] = \exp(-\alpha\gamma n)1[n]$, where membership function of set α is triangular $\mu_\alpha(a) = \text{triangle}(2, 3, 4)$. Let input signal $x[n] = \exp(-\beta\gamma n)1[n]$ and sampling interval be equal to $\gamma=0.01s$, number of samples $N = 100$. Fuzzy discrete impulse response and output signal of the filter with logic product (minimum) is shown in Fig. 4.

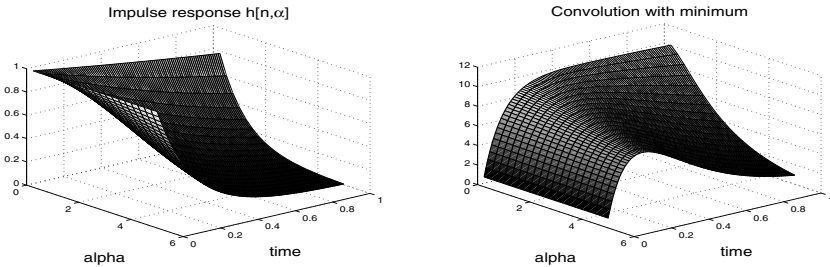


Fig. 4. Impulse response of fuzzy filter with T-norm (left) and output signal with logic product (right)

Lemma 9. *Frequency response of fuzzy filter with any T-norm equals to impulse response of fuzzy filter with algebraic product.*

Proof. The frequency response is fuzzy Fourier transform (see 4) of impulse response, which are identical for all T-norm filters.

It follows that such fuzzy filter with T -norm are direct generalization of conventional filter on fuzzy domain.

Theorem 3. *Fuzzy filters with any T -norm operation have appropriately identical: impulse, step, and frequency responses.*

Now, consider fuzzy filters with S -norms.

Lemma 10. *Impulse response of fuzzy filter with any S -norm does not depend on the type of S -norm used in the filter equation.*

Proof. The impulse response of the filter with any S -norm and $h^- [n, a]$ α -curve is equal to

$$h'^{\mp}[n, a] = \sum_{m=0}^{N-1} h^- [m, a] S \delta[n - m] = 1 + \sum_{m=0, m \neq n}^{N-1} h^- [m, a]. \quad (17)$$

Similarly for $h'^{\pm}[m, a]$. Thus, all filters with different S -norms have the same impulse response

$$\mathbf{h}'[n, \alpha] = 1 + \sum_{m=0, m \neq n}^{N-1} \mathbf{h}[m, \alpha]. \quad (18)$$

Lemma 11. *Step response of fuzzy filter with any S -norm does not depend on the type of S -norm used in the filter equation.*

Proof. The step response of fuzzy filter with any S -norm is equal to

$$k'^{\mp}[n, a] = \sum_{m=0}^{N-1} h^- [m, a] S 1[n - m] = n + 1 + \sum_{m=n+1}^{N-1} h^- [m, a]. \quad (19)$$

Similarly for $k'^{\pm}[n, a]$. Then

$$\mathbf{k}'[n, \alpha] = n + 1 + \sum_{m=n+1}^{N-1} \mathbf{h}[m, \alpha]. \quad (20)$$

Lemma 12. *Frequency response of fuzzy filter with any S -norm equals to impulse response of fuzzy filter with algebraic product.*

Proof. The frequency response is fuzzy Fourier transform (see [4]) of impulse response, which are identical for all S -norm filters.

Example 5. Consider example as before: $h[n] = \exp(-\alpha \Upsilon n) 1[n]$, membership function $\mu_{\alpha}(a) = \text{triangle}(2, 3, 4)$, $x[n] = \exp(-b \Upsilon n) 1[n]$, $\Upsilon = 0.01\text{s}$, $N = 100$. Fuzzy discrete impulse response and frequency response are shown in Fig. 5.

Theorem 4. *Fuzzy filters with any S -norm operation have appropriately identical: impulse, step, and frequency responses.*

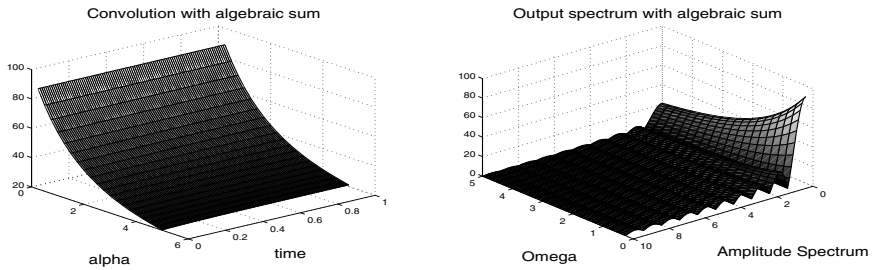


Fig. 5. Impulse responses (left) and frequency response (right) of fuzzy filter with S -norms

5 Conclusion

Some interesting properties of crisp filter with triangular norms and conorms are discovered. Impulse response, step response and frequency response do not depend on the T -norm used in the filter equations. The results are identical with ones obtained for conventional filter. It doesn't mean that all responses of such filters with different T -norms on an input signal are identical. Filters are nonlinear then responses can be different. Similar properties are shown for S -norms. Moreover, it was shown that if fuzzy parameter is introduced in the impulse response of the filter general conclusions are not changed. Impulse, step, and frequency responses do not depend on T -norm and appropriately S -norm applied in the fuzzy filter. When more parameters of the impulse response are fuzzy then general conclusions remain right. Only membership function for response is more difficult to calculate because of sup-min procedure must be applied.

References

1. Arakawa, K., Arakawa, Y.: Proposal of Median Type Fuzzy Filter and its Optimum Design. *EKE Trans.* J75-A(12), 1792–1799 (1992) (in Japanese)
2. Butkiewicz, B.S.: Towards Fuzzy Fourier Transform. In: 11th Int. Conf. IPMU, Paris, France, pp. 2560–2565 (2006)
3. Butkiewicz, B.S.: Fuzzy analog and discrete time invariant systems. In: Proc. SPIE, vol. 6937, pp. 1–8 (2007) (invited paper 693736)
4. Butkiewicz, B.S.: An Approach to Theory of Fuzzy Signals, Basic Definitions. *IEEE Trans. on Fuzzy Systems* 16(4), 982–993 (2008)
5. Dubois, D., Prade, H.: *Fuzzy Sets and Systems*. Academic Press, New York (1980)
6. Klement, E.P., Mesiar, R., Pap, E.: *Triangular norms*. Kluwer Acad. Publish., Dordrecht (2000)
7. Lee, K.-M., et al.: *Proceedings of NAFIPS*, pp. 577–580 (1996)

A Novel Fuzzy Color Median Filter Based on an Adaptive Cascade of Fuzzy Inference Systems

Mihaela Cislariu, Mihaela Gordan, and Aurel Vlaicu

Technical University of Cluj-Napoca, C. Daicoviciu 15,
400020 Cluj-Napoca, Romania

{Mihaela.Suteu, Mihaela.Gordan, Aurel.Vlaicu}@com.utcluj.ro

Abstract. Median filters are widely used for the reduction of impulse noise in digital images. Since particular problems appear for color images affected by color impulse noise (as color altering or imperfect noise elimination), many color median filtering methods are still developed. Among these, fuzzy median filters are reported to be highly efficient. In this paper, we propose a novel adaptive fuzzy logic-based color filtering algorithm, designed for the hue-saturation-value (HSV) space, in the form of an adaptive cascade of fuzzy logic systems. The performance of the proposed filter is superior to other fuzzy approaches, as shown by the experimental results.

Keywords: median filter, fuzzy logic, color noise, HSV color space.

1 Introduction

A fundamental problem of image processing is to effectively reduce noise from a digital image while keeping its features intact. A frequent and disturbing type of noise is the impulse noise, which, in color images, can alter the three color channels, appearing as color impulse noise. Whereas the most common filter type for the impulse noise removal in the grey scale images is the median filter, specific problems when filtering color images appear, that is, one cannot simply apply a median filter on each color component independently, because by this procedure, “false” colors may appear. Therefore, a number of methods have been proposed in the literature, such as: the adaptive scalar median filter; the median filter applied to the chromaticity in the hue-saturation-intensity (HSI) space; the median filter based on conditional ordering in the hue-saturation-value (HSV) space [3]; the arithmetic mean filter [1]. In the context of color image filtering for impulse noise removal, a significant number of algorithms use artificial intelligence in general (as e. g. [7, 6]) and fuzzy techniques in particular; the latter prove better performance than the crisp techniques - probably due to their specific flexibility. Thus, in [4] and [5], the authors propose a filtering algorithm based on a fuzzy ordering of colors, decided by a Mamdani fuzzy logic system, whose output is the most plausible “ordering plane value” of the input color. In [10], the authors propose a fuzzy two steps filtering algorithm in which a fuzzy noise detection phase

is followed by an iterative fuzzy filtering technique. The fuzzy noise detection method is based on the calculation of fuzzy gradient values and on fuzzy reasoning. Fuzzy methods can also be applied in deciding the appropriate color like in [2] where the authors propose to model the hues of the perceptually distinct colors as fuzzy sets and then select as output the most plausible color among those in the filtering window. Another fuzzy filtering approach is proposed in [8] where the authors define a new fuzzy metric that reduces the computational cost of the filtering procedure. In [12], an adaptive method of fuzzy filtering is proposed in respect to the filtering window size, which is adapted to the local level of noise contamination (level judged by a fuzzy set theoretic approach). If most of the pixels in the filtering window are not corrupted by noise, then the noisy pixel is replaced by some median value.

Despite the large variety of fuzzy color filtering methods, few of them refer explicitly to the particularities of filtering color impulse noise (e.g. [2], [8], [14]), as opposed to the simpler case of monochrome impulse noise - although some of the works analyze the filters' performance for the color noise also [9] [11]. Color impulse noise filtering in color images may rise particular problems especially in respect to fine details preservation both in color and achromatic regions (the behavior of color noise filters for "grey" regions may be different than in color regions of a color image). In this respect, room for the effective color noise reduction in color images still exist - as addressed by this work. We propose a scheme for the reduction of color impulse noise in color images using an adaptive cascade of three fuzzy logic systems. The adaptation refers to the configuration of the cascade: in the chromatic areas, a two fuzzy systems cascade is sufficient to efficiently remove the color noise, whereas in the mostly achromatic (gray) areas, three fuzzy filters, one on each color component, must be applied, otherwise noisy pixels are not efficiently removed. The details and experimental results of the proposed system are presented in the following.

2 Single Input - Single Output Fuzzy Inference Systems for Color Components Ordering

A suitable representation space for a color image is HSV, since its components are directly correlated to the human perception of color. Hue (h) represents the dominant wavelength, saturation (s) refers to the purity and the value (v) is a measure of the color position along the lightness axis.

In [3] the authors propose an ordering color scheme in the HSV color space, defined as follows. Let any color be represented as $c(h, s, v)$. Two ordering operators of colors $<_c$ and $=_c$ are then defined for two colors $c(h_i, s_i, v_i)$ and $c(h_k, s_k, v_k)$ as:

$$\begin{aligned} c(h_i, s_i, v_i) <_c c(h_k, s_k, v_k) &\Leftrightarrow \\ (v_i < v_k) \vee (v_i = v_k \wedge s_i > s_k) \vee (v_i = v_k \wedge s_i = s_k \wedge h_i < h_k) & \quad (1) \\ c(h_i, s_i, v_i) =_c c(h_k, s_k, v_k) &\Leftrightarrow (v_i = v_k) \wedge (s_i = s_k) \wedge (h_i = h_k). \end{aligned}$$

The above equations can be seen as crisp relations between the color components. A drawback of the crisp conditions is however a lack of flexibility in the ordering, which can even sometimes lead to an incorrect ordering, making an outlier be found as the best candidate even in rather uniform color patches; as a result, even in some “simple” cases for the human eye, the algorithm does not eliminate all the color noise, as it can be seen in Fig. 3f). An improvement of the flexibility of the ordering may be achieved by fuzzy relations, as proposed here. We propose a straightforward extension of the crisp ordering to fuzzy ordering relations, as follows. Each fuzzy ordering relation is implemented in the form of a single input - single output Takagi-Sugeno fuzzy system, as shown in Fig. 1. The input variable of each such color ordering fuzzy system for a certain color component is the difference between the values of that component in the two colors; the values of the difference are described in the fuzzy system by three linguistic values (represented by three fuzzy sets): NEGATIVE, ZERO and POSITIVE. In the current implementation, we have chosen the most simple, piecewise linear shape of the three membership functions (the membership functions of the fuzzy sets describing the linguistic values NEGATIVE and POSITIVE are trapezoidal, and the membership function of the fuzzy set ZERO is triangular symmetrical versus zero). The membership functions are set to form a fuzzy partition of the universe of discourse of the input variable. The values of their defining parameters are found in a tuning step, from several color images affected by different amounts of color noise, to optimize the noise removal performance, individually, on each color component. Due to the fuzzy partition property, it is enough to define the parameters of the membership function ZERO, whose optimal values have been found to be: $h_{NEGATIVE} = -0.1$, $h_{POSITIVE} = 0.1$, $s_{NEGATIVE} = -0.01$, $s_{POSITIVE} = 0.01$, $v_{NEGATIVE} = -0.02$, $v_{POSITIVE} = 0.02$.

The output of each fuzzy system is an indicator of the plausibility that the colors are ordered increasingly; for Takagi-Sugeno systems, the output sets are singletons (numerical constants), which have been set to -1, 0, +1. The value +1 indicates that the first color considered is “approximately greater than” the second color; the value -1 indicates that the first color considered is “approximately less than” the second color; 0 indicates they are “approximately equal”.

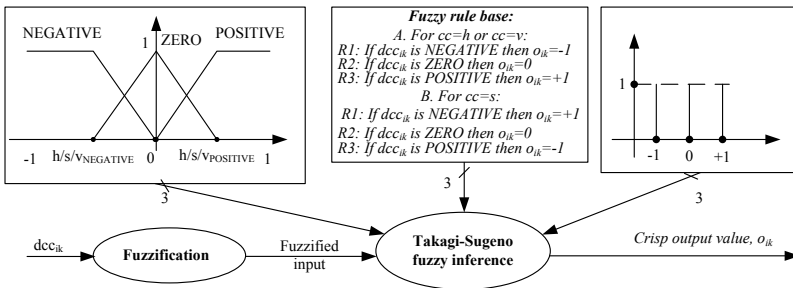


Fig. 1. Fuzzy logic system block diagram

Let us denote any of the three color components by cc , $cc \in \{h, s, v\}$. Let c_i and c_k be the two colors considered in Eqn.(1), and cc_i , cc_k - their color components. We denote the variation of a color component between the two colors c_i and c_k by $dcc_{ik} = cc_i - cc_k$. Let o_{ik} be the output of the corresponding fuzzy system, $o_{ik} \in [-1; +1]$. Each of the three fuzzy systems used for ordering a particular component (h, s, v) includes a fuzzy rule base consisting of three rules, described in Fig. 1. The different definition of the fuzzy rule bases for the three types of color components is needed to obey the crisp ordering conditions from Eqn.(1) - since we want the proposed fuzzy orderings to be straightforward generalizations of the crisp color orderings in [3].

The Takagi-Sugeno inference yields the fuzzy system's output according to:

$$o_{ik} = \frac{NEGATIVE(dcc_{ik}) \cdot (-1) + ZERO(dcc_{ik}) \cdot 0 + POSITIVE(dcc_{ik}) \cdot (+1)}{NEGATIVE(dcc_{ik}) + ZERO(dcc_{ik}) + POSITIVE(dcc_{ik})}. \quad (2)$$

where $NEGATIVE(dcc_{ik})$ is the membership degree of the value dcc_{ik} to the fuzzy set $NEGATIVE$, $ZERO(dcc_{ik})$ - its membership degree to the fuzzy set $ZERO$, $POSITIVE(dcc_{ik})$ - its membership degree to the fuzzy set $POSITIVE$.

3 The Proposed Architecture for Color Filtering in HSV Space

Unlike in similar works as [3] and [5], where the colors in a given pixels window are ordered in a single step in the HSV space to get their median value, the approach proposed here consists in a cascade of fuzzy median filters applied on the color components, adapted to the local "chromaticity" of the image. In a previous work [13] we proposed a technique in which the ordering relations could be applied in exactly the same fashion as in Eqn.(1), in a cascade of three filters guided by the "or" operator. While highly effective in noise removal in uniform areas, this technique may fail to preserve sharp edges, due to the large amount of filtering; better results are achieved by the adaptation of the number of filters in the cascade to the local chromaticity.

An additional modification of the crisp color ordering algorithm, apart from the fuzzification of the ordering relations, refers to the selection of the priorities of the color components in noise filtering. According to the human perception, one possible way of reducing color noise is by applying filters separately, first on the perceptually most important color component, then on the middle important and last, on the least important color component, thus producing a cascade of filters. This logic is also employed in the algorithm described by Eqn.(1), but probably since the filter presented there is devoted to black & white noise filtering, the value v component is considered to be the most significant in filtering. This is however not the case of the color noise, since the perceptually most important component for color visualization in the HSV space is the hue; the next important is the saturation and the least important is the value. Therefore, in the case of the algorithm proposed in this paper, we first consider the application of a

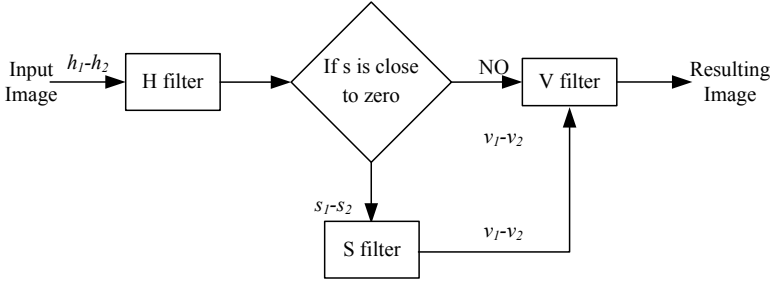


Fig. 2. The adaptive cascade scheme

fuzzy ordering relation on hue h , secondly on saturation s , and only last on the value v . Additional considerations regarding the color noise perception in highly chromatic and achromatic regions can be made, allowing for different filtering configurations in the two cases, as follows. In the achromatic local regions, the saturation component provides one of the most discriminative criteria in filtering, since all the pixels in the patch except for the noise pixel have their saturation s close to zero. Neither the hue h nor the value v are not enough in filtering, except for black/white noise pixel. In this case, the three filters must be applied in the local achromatic region. In the highly chromatic regions, hue and value components make usually a spurious point to appear as noise - we mostly do not take into account the saturation. Therefore, in this case, it is sufficient to filter such highly chromatic patches only based on the hue and value components, excluding the saturation filter from the cascade. The cascade adaptation is based on a patch chromaticity classification performed within a certain neighborhood of the current examined pixel. The block diagram of the cascaded filtering is outlined in Fig. 2. In brief, the operation of the cascade of filters can be described by the following steps, applied on each pixel in the image:

1. In the 3×3 pixels neighborhood around, sort decreasingly the colors from those with smallest h to those with greatest h , using the output o_{ik} from the fuzzy system having dh_{ik} at its input. Two colors c_i and c_k are interchanged in the string if $o_{ik} < 0.48$. Replace the current color by the middle of the ordered vector string.
2. *Pixel classification based on its local chromaticity:* Let a window of size $n \times n$ pixels, $n \gg 3$, be some local neighborhood of our current processed pixel. If the 90% percentile of the saturation s of the sample of n^2 pixels is less than a threshold $t_s = 0.2$, then the pixel is classified as being in a “gray” (achromatic) patch. Otherwise, the pixel is said to be in a “colored” patch.
3. If the pixel is in a “gray” patch, apply the filtering procedure from step (1) on the saturation component, s . Otherwise, go to step (4).
4. Apply the same procedure as in step (1) on the resulting image obtained from step (3) or step (1), using the value component v and the corresponding fuzzy system.

4 Implementation and Results

The performance of the proposed algorithm has been evaluated and compared with several other existing filters for impulse noise reduction. The algorithm of the proposed cascade of filters has been implemented in Matlab, using the Image Processing Toolbox and Fuzzy Logic Toolbox. In the experiment we use a filtering window of 3×3 pixels. For pixel classification based on its local chromaticity - required in the adaptation of the fuzzy filters cascade - we use a neighborhood size of 17×17 pixels, which is large enough to be relevant to our local classification goal. To assess the performance of the proposed algorithm as compared to some familiar state-of-the-art methods, we have also implemented the related filtering approaches presented in [3] and [5] and also the previous work [13]. A set of standard color images have been chosen for testing the performance of the filter; they were corrupted by a controlled amount of color impulse noise, at different noise levels. The objective quantitative measure used for the evaluation of the filtering performance is the peak signal to noise ratio (PSNR), defined, for a color image, as: $PSNR(F, O) = 10 \log_{10}(S^2/MSE(F, O))$, where O is the original color image, F is the filtered image, and S - the maximum intensity of each color component. Some experimental results are illustrated in Fig. 3 below for the Peppers image. One can see that both visually (in respect to the elimination of the noisy pixels) and numerically (in respect to the PSNR, summarized in

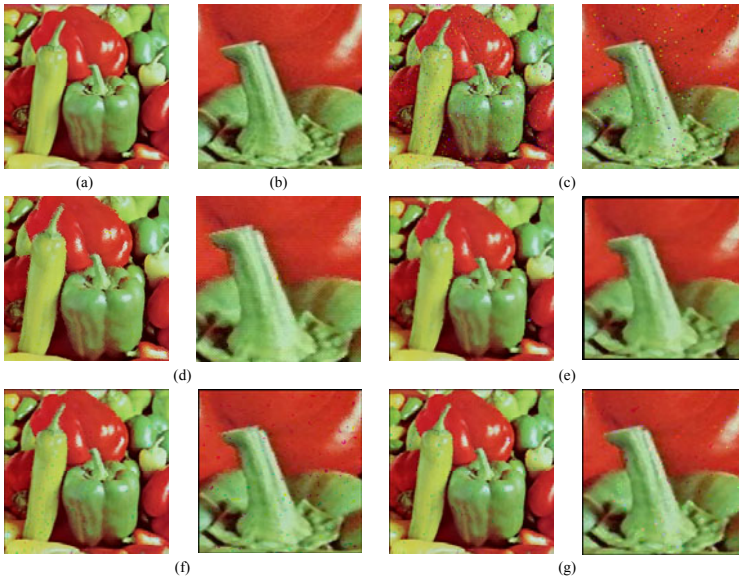


Fig. 3. (a) The original image “Peppers”; (b) a part of the original image; (c) the image affected by color noise; (d) the result for the proposed algorithm; (e) the result of the previous work [13]; (f) the result for the crisp algorithm in [3]; (g) the result for the fuzzy algorithm in [5]

Table 1. Comparative results in PSNR for various percentages of Impulse Noise; HSVmf - HSV median filter [3]; Fcmf - fuzzy color median filter [5]; HSVc - HSV cascade from previous work [13]; HSVac - HSV adaptive cascade algorithm proposed

Image	Basalt					Peppers				
Noise level	Noisy	HSVmf	Fcmf	HSVc	HSVac	Noisy	HSVmf	Fcmf	HSVc	HSVac
3%	18.34	28.86	31.31	31.95	32.64	21.01	26.80	26.48	28.95	29.69
5%	17.38	27.10	30.37	32.01	31.64	18.64	25.11	25.64	28.57	28.57
10%	14.34	23.46	23.53	29.06	29.93	15.81	22.23	22.76	27.26	27.31
15%	12.19	20.23	22.03	28.91	29.49	13.73	19.46	19.68	23.03	24.46
20%	11.03	18.56	19.13	30.04	28.98	12.56	16.92	16.96	17.84	20.49
Image	Baboon					Boats				
Noise level	Noisy	HSVmf	Fcmf	HSVc	HSVac	Noisy	HSVmf	Fcmf	HSVc	HSVac
3%	20.62	24.86	25.88	28.85	34.42	25.76	33.95	31.88	34.43	34.69
5%	18.43	22.94	23.71	27.96	30.40	23.55	30.96	28.49	30.70	33.57
10%	15.33	19.99	20.58	26.27	26.66	20.30	24.85	24.72	25.74	26.69
15%	13.59	17.93	18.25	22.69	28.79	18.36	22.60	21.92	21.65	22.90
20%	12.38	15.96	15.93	19.05	19.26	17.32	19.57	20.25	20.42	20.22
Image	Lena									
Noise level	Noisy	HSVmf	Fcmf	HSVc	HSVac					
3%	25.40	28.56	29.36	29.34	30.94					
5%	23.16	27.16	27.61	28.52	29.10					
10%	19.81	24.01	24.66	24.91	24.89					
15%	18.43	22.18	23.09	23.09	23.10					
20%	17.04	19.78	20.49	20.38	21.12					

Table 1, for three color test images: Basalt, Peppers and Baboon), the new solution outperforms the other filters used for comparison. As the numerical results show, the PSNR of the proposed algorithm is in most of the cases superior to the other solutions, except for some individual regions very rich in details (with many sharp and narrow edges) - where the local PSNR provided by our algorithm is decreased because of the unwanted blurring effect mentioned in the previous section. On the other hand, it is well known that the visual perception is not always in concordance to the PSNR, and in this respect our method performs very well.

5 Conclusions

In this paper a new fuzzy color median filter suitable for restoration of color images affected by color impulse noise is proposed. Using fuzzy systems theory, we propose a fuzzy ordering procedure for color vectors in the HSV color space. The main feature of this filter is that it can adapt itself to the local chromaticity of the image and it is adapted to the human perception of color, defining and evaluating some approximate perceptual color differences. Numerical measurements (namely, the PSNR) and visual observation show that the proposed filter has good performance, as compared to other similar color median filters

for color noise removal presented in the literature. The filter reduces the color impulse noise extremely effectively, but due to the cascade application of median filters, the ability to preserve sharp edges is not extremely good - although it is much better than the previous work [13]. The improvement of the filter in this aspect will make the object of our future work.

References

1. Plataniotis, K.N., Venetsanopoulos, A.N.: Color image processing and applications. Springer, Berlin (2000)
2. Vertan, C., Boujemaa, N., Buzuloiu, V.: A Fuzzy Color Credibility Approach To Color Image filtering. In: International Conference on Image Processing, Canada, September 2000, vol. 2, pp. 808–811 (2000)
3. Koschan, A., Abidi, M.: A comparison of median filter techniques for noise removal in color images, University of Erlangen-Nurnberg, Institute of Computer Science 34(15), 69–79 (2001)
4. Andreadis, I., Louverdis, G.: Soft morphological color image processing: a fuzzy approach. In: IEEE 11th Mediterranean Conf. on Automation & Control, Rhodes, Greece, June 2003, vol. (7-008), pp. 1–5 (2003)
5. Andreadis, I., Louverdis, G., Chatizianagostou, S.: New fuzzy color median filter. Journal of Intelligent and Robotics Systems (41), 315–330 (2004)
6. Yan, L., Wang, L., Yap, K.: A noisy chaotic neural network approach to image denoising. In: IEEE International Conference on Image Processing, ICIP, Singapore, October 2004, vol. 2, pp. 1229–1232 (2004)
7. Tan, Y.P., Yap, K.H., Wang, L.P.: Intelligent Multimedia Processing with Soft Computing. Springer, Berlin (2004)
8. Morillas, S., Gregori, V., Peris-Fajarens, G., Latorre, P.: A New Median Filter Based on Fuzzy Metrics. In: Kamel, M.S., Campilho, A.C. (eds.) ICIAR 2005. LNCS, vol. 3656, pp. 82–91. Springer, Heidelberg (2005)
9. Morillas, S.: Fuzzy metrics and peer groups for impulsive noise reduction in color images. In: 14th European Signal and Image Processing Conference, Italy (2006)
10. Schulte, S., de Witte, V., Nachtegaal, M., van der Weken, D., Kerre, E.E.: Fuzzy Two-Steps Filter For Impulse Noise Reduction From Color Images. IEEE Transaction on Image Processing 15(11), 3568–3579 (2006)
11. Schulte, S., de Witte, V., Nachtegaal, M., van der Weken, D., Kerre, E.E.: Histogram-based fuzzy colour filter for image restoration, vol. (9), pp. 1377–1390 (September 2007)
12. Zhou, Y., Tang, Q., Jin, W.: Adaptive Fuzzy Median Filter for Image Corrupted by Impulse Noise. In: Image and Signal Processing, CISP, China, vol. 3, pp. 265–269 (2008)
13. Suteu, M.: A Novel Fuzzy Color Median Filter Based on a Cascade of Fuzzy Inference Systems. In: The 5th Symposium for Students in Electronics and Telecommunications SSET, Cluj-Napoca (May 2009)
14. Xu, Z., Wu, H., Qiu, B., Yu, X.: Geometric Features-Based Filtering for Suppression of Impulse Noise in Color Images. Transactins on Image Processing 18(8) (August 2009)

Automatic Modeling of Fuzzy Systems Using Particle Swarm Optimization

Sergio Oliveira Costa Jr.¹, Nadia Nedjah¹, and Luiza de Macedo Mourelle²

¹ Department of Electronics Engineering and Telecommunications

² Department of Systems Engineering and Computation
Faculty of Engineering, State University of Rio de Janeiro
{serol,nadia,ldmm}@eng.uerj.br

Abstract. Fuzzy systems are currently used in many kinds of applications, such as control, for their effectiveness and efficiency. However, these characteristics depend primarily on the model yield by human experts, which may or may not be optimized for the problem at hand. Particle swarm optimization (PSO) is a technique used in complex problems, including multi-objective problems. In this paper, we propose an algorithm that can generate fuzzy systems automatically for different kinds of problems by simply providing the objective function and the problem variables. This automatic generation is performed using PSO. To be able to do so and in order to avoid dealing with inconsistent fuzzy systems, we used some known techniques, such as the WM method, to help in developing meaningful rules and clustering concepts to generate membership functions. Tests using the sigmoid 3D curve have been carried out and the obtained results are presented.

1 Introduction

Fuzzy systems form an important tool to model complex problems based on imprecise informations and/or in situations where a precise result is not of interest and an approximation is sufficient [1]. The performance of a fuzzy system depends on the expert's interpretation, which leads to in the generation of the rule base and membership functions of the system. To minimize this dependency, some methods are being used in the attempt to automatically generate the components required in a fuzzy system. For the membership functions, clustering-based algorithms, such as Fuzzy C-means and its generalizations as *Pre-shaped C-means* [2], are usually used. Other approaches also exist [8]. The major difficulty in the development of fuzzy systems consists of the definition of membership functions and rules that provide the desired behavior of these systems.

Swarm Intelligence is an area of artificial intelligence based on collective and decentralized behavior of individuals that interact with each other, as well as with the environment [1]. PSO is a stochastic evolutionary algorithm, based on swarm intelligence, that searches for the solution of optimization problems in a specific search space and is able to predict the social behavior of individuals according to defined objectives [5].

Methods based on examples, such as the Wang-Mendel or WM method [10], are usually used for automatic rule generation. Also, there are many research works that exploit evolutionary algorithms (EA), both to optimize the rule base and the membership functions. In [3], genetic algorithms (GA) are used to generate the rule base, with candidate rules pre-selection. In [7], the authors use EA to generate fuzzy systems that are more compact and more interpretable by humans. In [9], the authors use clustering techniques and GA to define good sets of rules for classification problems. In [4], the authors use evolutionary technique and GA to generate fuzzy systems from some given knowledge bases.

In this paper, we developed an algorithm based on PSO to generate fuzzy systems for any kind of problem, provided an objective function that may be continuous or discrete. Using simple informations, such as variable names, the corresponding domains and the objective function, this algorithm can yield an appropriate fuzzy system. Some tests were performed with a known control surface to validate the effectiveness of the tool.

The rest of this paper is organized in five sections. Firstly, in the Section 2, we briefly describe the WM method of rules generation. After that, in the Section 3, we introduce the proposed method for the automatic modeling of fuzzy systems using PSO. For this purpose, we first define the structure of a particle and the coordinates used to position it within the search space as well as the fitness function of the represented system. Then, in the Section 4, we present the obtained results to model a commonly used control surface.

2 Rule Generation Methods

The rule generation method referred to as Wang-Mendel (WM) [10] uses an input-output data set for the problem at hand, to generate a rule set of fuzzy systems. The input-output data set is usually provided as $(x^p; y^p)$, $p = 1, 2, \dots, N$, wherein $x^p \in R^n$ and $y^p \in R$. This method extracts the rules that best describe how the output variable $y \in R$ is influenced by the n input variables $x = (x_1, \dots, x_n) \in R^n$, based on the provided examples.

For instance, assuming two data sets to a system with two input variables x_1 and x_2 and an output y , that are (x_1^1, x_2^1, y^1) and (x_1^2, x_2^2, y^2) , and the membership functions showed in the graphics of the Fig. 1. To obtain the rules represented by these two sets, first we must get the degree of confidence using the membership functions, for each data set. In this case, we have:

- x_1^1 : degree 0.67 in $A1$ and 0.11 in $A2$;
- x_2^1 : degree 0.16 in $B1$ and 0.80 in $B2$;
- y^1 : degree 0.66 in $C1$;
- x_1^2 : degree 0.25 in $A2$ and 0.68 in $A3$;
- x_2^2 : degree 0.10 in $B3$ and 0.58 in $B4$;
- y^2 : degree 0.39 in $C2$ and 0.51 in $C3$.

In order to generate the rules, we always keep the membership functions in which the variable has the highest degree, and so we discard the functions that have

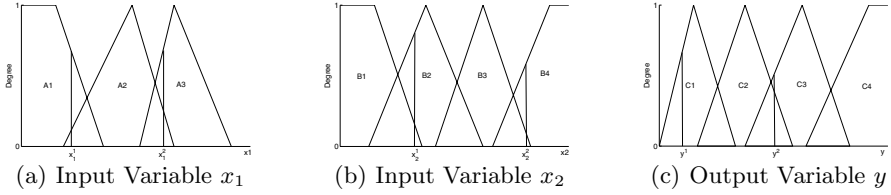


Fig. 1. Example of input-output data set for rules generation

lower degree. So, for the data sets defined in Fig. 1, we have (1). So the first rule would be “If x_1 is A1 and x_2 is B2 then y is C1” and the second, “If x_1 is A3 and x_2 is B4 then y is C3”.

$$\begin{aligned} (x_1^1, x_2^1, y^1) &= [x_1^1(0.67 \text{ in } A1), x_2^1(0.80 \text{ in } B2), y^1(0.66 \text{ in } C1)] \\ (x_1^2, x_2^2, y^2) &= [x_1^2(0.68 \text{ in } A3), x_2^2(0.58 \text{ in } B4), y^2(0.51 \text{ in } C3)] \end{aligned} \quad (1)$$

Note that each data set generates one single rule. Considering a real system, it is very possible that these rules can be conflicting rules. To overcome this problem, one can associate degrees of confidence to each generated rule, using the degree of relevance of each rule term. Equation 2 shows how this degree can be computed:

$$C(Rule) = \mu(x_1) \times \mu(x_2) \times \mu(y), \quad (2)$$

wherein C is degree of *Rule* and $\mu(x_1)$, $\mu(x_2)$ and $\mu(y)$ are the degree of relevance of each rule term. In the case of the first rule, the associated confidence degree would be $C(Rule_1) = 0.67 \times 0.80 \times 0.66 = 0.35376$.

There are also methods based on genetic algorithms. In [3], the authors use the WM method to generate the initial rule set and then apply their own genetic algorithm on some classification rate of the rules. This method is only used for classification problems.

3 Proposed Automatic Generation

In this work, PSO is used to evolve the fuzzy systems parameters of the *Mamdani* type, both for rules and membership functions. The search algorithm is based on these two elements and always tries to improve the solution at hand. However, the functions are not modified in every iteration, unlike the rules, whose modification obeys to pre-determined update rate, that is defined at the beginning of the evolutionary process. The purpose is to maintain the functions stable for some time, giving more time for the algorithm to search for more appropriate rules for those functions. At the end of each execution, when the algorithm reaches the stop criterion, it returns the best solution found.

There are four important aspect that define the performance of the PSO search. These are the particle representation, the position coordinates of a given particle in the search space, the fitness function that allows us to determine how appropriate is the fuzzy system associated with a given particle and how to update the system represented by the particle at hand.

3.1 Representation

Each particle is associated with a fuzzy system and a position in the search space, that is represented by an n -position vector, where n depends on the number of used variables. In this work, the fuzzy system is defined by an hierarchical structure as described in Fig. 2.

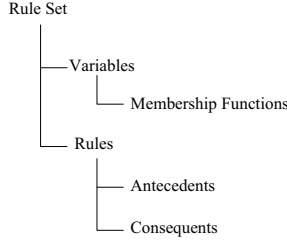


Fig. 2. Fuzzy representation structure

3.2 Particle Position and Movement

The position vector has one entry for the rules number, another one for the completeness factor, and m positions for the number of the functions, where m is the number of the system variables. Thus, the position dimension is dependent on the number of variables of the problem. The completeness factor is a criterion that measures the *discontinuity* between functions in the variables domain (see Section 3.3). The mutation operator determines how the update of each one these items is performed. This update promotes the movement of particles on the search space. In this work, we used three kinds of mutations:

- If the velocity relating to the number of rules is positive, then we increase the rules number. Otherwise, we decrease it.
- Changing the number of functions for each variable of the fuzzy system follows the same criterion, given above.
- The change of the completeness is performed increasing the width of a function. Thus, the tendency is to reduce the empty space in the domain, if any. Similarly, reducing the width of the function, we alter the distinctness between the available ones. The more positive the velocity is, the bigger the increase in domain of one of the functions. If the more negative the velocity is, the smaller the decrease in the domain.

3.3 Fitness Function

Inspired by the work reported in [7], the fitness of each particle is defined as in (3):

$$F = -100 \times \omega_1 \times \log(MSE) + 50 \times \omega_2(1 - C_r) + 50 \times \omega_3(1 - C_f) + 50 \times \omega_4(1 - P_D) + 50 \times \omega_5(1 - P_C), \quad (3)$$

wherein MSE is the mean-square error of the difference between the returned value by the objective function (y_h) and the returned value by the fuzzy model (y_h^F), as (4), wherein N_D is the number of data.

$$MSE = \frac{1}{N_D} \sum_{h=1}^N (y_h - y_h^F)^2, \quad (4)$$

The C_r term represents the relation between the amount of rules presents on the model and the total of possible rules, and C_f represents the relation between the amount of functions presents on the system and the total of possible functions, as in (5):

$$C_r = \frac{N_R}{N^{max}} \quad C_f = \frac{N_F}{N^{max}} \quad (5)$$

Term P_D is a criterion that measures the *distinctness* between the membership functions of the variables, defined in (6):

$$P_D = \frac{1}{N_V} \sum_{i=1}^{N_V} \left(\frac{1}{N_S^i} \sum_{h=1}^{N_S^i} \lambda_{ih} \right), \quad (6)$$

wherein N_V is the number of variables, N_S^i is the total possible interval of overlap between functions of the i -th variable, λ_{ih} is the width of the h -th overlap and χ_i is the width of the variable domain. In order to the determine λ_{ih} , it is necessary to define the level ξ_D , drawing a horizontal line, crossing all the functions, as showed in the Fig. 3(a). Term P_C is a criterion that represents the *completeness* of the membership functions, in relation to the domain, and is defined as in (7):

$$P_C = \frac{1}{N_V} \sum_{i=1}^{N_V} \left(\frac{1}{N_D^i} \sum_{h=1}^{N_D^i} \gamma_{ih} \right), \quad (7)$$

wherein N_D^i is the total possible number of discontinuity between functions of the i -th variable, γ_{ih} is the width of h -th discontinuity and χ_i is the width of the variable domain.

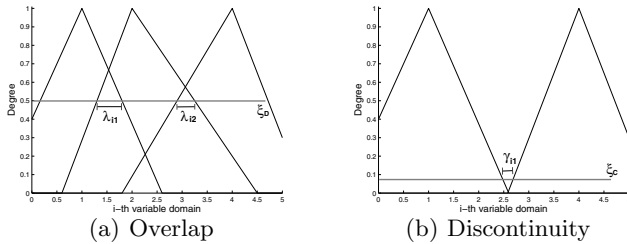


Fig. 3. Overlap and discontinuity illustrations

In order to determine γ_{ih} , is necessary to define the level ξ_C , in a similar way to ξ_D , as shown Fig. 3(b).

The coefficients $\omega_1, \omega_2, \omega_3, \omega_4$ and ω_5 , control the contribution of each term of (3) in the evaluation of the fuzzy system associated with the particle.

This evaluation function covers the many required criteria of a fuzzy system. These are the precision, by the error quantification; the compactness, by the relation between the number of the rules and functions of the model and the total possible number; and the interpretability, by measuring of the distinctness and completeness, providing a complete model evaluation.

4 Results and Tests

The WM method [10], introduced in Section 2, was used to initialize rules of the fuzzy systems of each particle. Besides, the concept of clustering was used in the membership functions generation, to decrease the possibility of yielding functions that are incompatible with the fuzzy system. In order to evaluate the implementation, reported throughout this paper, we performed some tests to generate the control surface of a water vehicle [6] shown in Fig. 4(a). Equation (8) shows the function used to generate this curve.

$$z = \frac{2}{1 + e^{-2x-2y}} - 1 \quad (8)$$

Initially, we used variable parameters: number of rules may vary from 1 to 50 and membership functions from 1 to 7 for each variable. However, the results were not satisfactory. The surfaces generated by the evolved fuzzy systems were too far from the original one, as shown in Fig. 4(b).

Several tests were performed, and the PSO parameters were adjusted many times with values near to those referenced in related literature [5]. Defining a small interval for the rules set and setting the values of the number of functions (number of rules varying from 1 to 4 and two membership functions for each variable), the algorithm evolved a fuzzy system that yielded curves that meet the original surface in several points. Table 1 shows some variations of parameters, used in the tests. The best result obtained so far is depicted in Fig. 4(c).

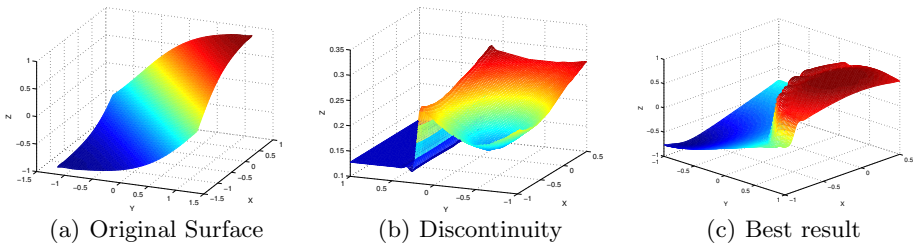
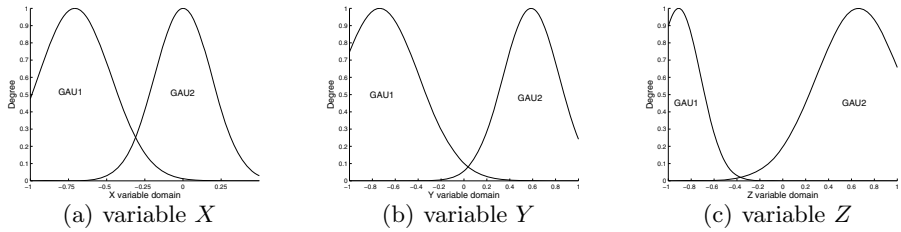


Fig. 4. Original surface and an initial result

Table 1. Algorithm parameters

Parameter	Values
w	0, 0.5 e 1
c_1, c_2	1, 1.1, 1.5, 1.9, 2, 3 e 4
Number of particles	10, 20, 40 e 100
Total of iterations	1000, 10000 e 50000
Minimum number of rules	1, 2 e 4
Maximum number of rules	1, 4, 16 e 50
Minimum number of functions	1 e 2
Maximum number of functions	2, 4 e 7
Kind of Function	different kind of functions Gaussian functions only Triangular functions only
Initialization	With WM e Without WM
Total Rules WM	0, 2 e 4
$K_1 - K_5$	0, 0.5 e 1

**Fig. 5.** Functions of the variable X , Y and Z

The set of rules evolved area as follows. The membership functions of the fuzzy systems that provided the best results are presented in Fig. [5\(a\)](#), [5\(b\)](#) e [5\(c\)](#).

1. if X is GAU_1 and Y is GAU_1 then Z is GAU_1 ;
2. if X is GAU_2 and Y is GAU_2 then Z is GAU_2 .

5 Conclusion

In this paper, we illustrated the use of PSO to automatically generate the fuzzy rules, fuzzy variables together with the corresponding membership functions of fuzzy systems. We described the particle representation, its movement in the search space and we provided a fitness function that allows us to assess the appropriateness of the evolved fuzzy system. This experience showed that the performance of the evolutionary process is very much dependent on the choice of the parameters, such as the number of membership functions per variable as well as on the number of rules allowed in the system. More tests are being carried out in order to reach a final conclusion on the subject.

References

1. Beni, G., Wang, J.: Robots and Biological Systems: Towards a New Bionics? Toscana, Italy. NATO ASI Series (1989)
2. Chen, L., Chen, C.L.P.: Pre-shaped fuzzy c-means algorithm (pfcmm) for transparent membership function generation. In: IEEE International Conference on Systems, Man and Cybernetics, pp. 789–794 (October 2007)
3. Cintra, M.E., de Arruda Camargo, H.: Fuzzy rules generation using genetic algorithms with self-adaptive selection. In: IEEE International Conference on Information Reuse and Integration, pp. 261–266 (August 2007)
4. Cordn, O., Herrera, F.: A hybrid genetic algorithm-evolution strategy process for learning fuzzy logic controller knowledge bases. In: Genetic Algorithms and Soft Computing, pp. 251–278. Physica-Verlag, Heidelberg (1996)
5. Engelbrecht, A.P.: Fundamentals of Computational Swarm Intelligence. John Wiley and Sons Ltd., England (2005)
6. Guo, B., Liang, X., Wang, B., Wan, L.: Sigmoid surface control for mini underwater vehicles by improved particle swarm optimization. In: International Conference on Robotics and Biomimetics (December 2007)
7. Kim, M.-S., Kim, C.-H., Lee, J.j.: Evolving compact and interpretable takagi-sugeno fuzzy models with a new encoding scheme. IEEE Transactions on Systems, Man and Cybernetics, Part B 36, 1006–1023 (2006)
8. Krone, A., Slawinski, T.: Data-based extraction of unidimensional fuzzy sets for fuzzy rulegeneration. In: IEEE World Congress on Computational Intelligence, IEEE International Conference on Fuzzy Systems, vol. 2, pp. 1032–1037 (May 1998)
9. Setnes, M., Roubos, H.: GA-fuzzy modeling and classification: complexity and performance. IEEE Transactions on Fuzzy Systems 8, 509–522 (2000)
10. Wang, L.-X.: The WM method completed: A flexible fuzzy system approach to data mining. IEEE Transactions on Fuzzy Systems 11, 768–782 (2003)
11. Zadeh, L.A.: Fuzzy sets. Information and Control 8, 338–353 (1965)

On Automatic Design of Neuro-fuzzy Systems

Krzysztof Cpałka^{1,2}, Leszek Rutkowski^{1,2}, and Meng Joo Er³

¹ Academy of Management (SWSPiZ), Institute of Information Technology, Poland

² Czestochowa University of Technology,

Department of Computer Engineering, Poland

³ Nanyang Technological University,

School of Electrical & Electronic Engineering, Singapore

{Krzysztof.Cpałka,Leszek.Rutkowski}@kik.pcz.pl, emjer@ntu.edu.sg

Abstract. In this paper we propose a new approach for automatic design of neuro-fuzzy systems. We apply evolutionary strategy to determine the number of rules, number of antecedents, number of inputs, and number of discretization points of neuro-fuzzy systems. Proper selection of these elements influences the accuracy of the system and its interpretability. The algorithm has been tested using well-known classification benchmarks.

1 Introduction

Neuro-fuzzy systems combine the natural language description of fuzzy systems and the learning properties of neural networks. In literature various structures of such systems and corresponding learning methods have been proposed (see e.g. [2]-[5], [7]-[10]). Recently several algorithms have been developed to increase interpretability and accuracy of these systems. For various methods of designing fuzzy rule-based systems the reader is referred e.g. to [1], [2], [6].

In this paper we propose a new approach for automatic selection of the number of rules, number of antecedents, number of inputs, and number of discretization points of neuro-fuzzy systems. Proper selection of these elements influences the accuracy of the system and its interpretability. Our method is based on the evolutionary strategy (μ, λ) and allows to design a structure of neuro-fuzzy systems and to learn parameters of their membership functions.

This paper is organized into five sections. In Section 2 the description of flexible neuro-fuzzy system is given. In Section 3 the evolutionary learning and designing of neuro-fuzzy systems is presented. Section 4 shows the simulation results. Conclusions are drawn in Section 5.

2 Description of Flexible Neuro-fuzzy System

We will consider multi-input, single-output a neuro-fuzzy system of the logical type (see e.g. [2], [3], [5], [9], [10]), mapping $\mathbf{X} \rightarrow \mathbf{Y}$, where $\mathbf{X} \subset \mathbf{R}^n$ and $\mathbf{Y} \subset \mathbf{R}$. The fuzzy rule base of the system consists of a collection of N fuzzy IF-THEN rules in the form

$$R^k : [\text{IF } x_1 \text{ is } A_1^k \text{ AND } \dots \text{ AND } x_n \text{ is } A_n^k \text{ THEN } y \text{ is } B^k] , \quad (1)$$

where $\mathbf{x} = [x_1, \dots, x_n] \in \mathbf{X}$, $y \in \mathbf{Y}$, $A_1^k, A_2^k, \dots, A_n^k$ are fuzzy sets characterized by membership functions $\mu_{A_i^k}(x_i)$, $\mathbf{A}^k = A_1^k \times A_2^k \times \dots \times A_n^k$, B^k are fuzzy sets characterized by membership functions $\mu_{B^k}(y)$, $k = 1, \dots, N$, and N is the number of rules, respectively.

Each of N rules (II) determines a fuzzy set $\bar{B}^k \subset \mathbf{Y}$ given by

$$\mu_{\bar{B}^k}(y) = \mu_{\mathbf{A}^k \rightarrow B^k}(\bar{\mathbf{x}}, y) = I_{fuzzy}(\mu_{\mathbf{A}^k}(\bar{\mathbf{x}}), \mu_{B^k}(y)), \quad (2)$$

where $I_{fuzzy}(\mu_{\mathbf{A}^k}(\bar{\mathbf{x}}), \mu_{B^k}(y))$ denotes a fuzzy implication (see eg. [10], [9]). Aggregating fuzzy sets \bar{B}^k , we get set B' with membership function given by

$$\mu_{B'}(y) = \bigvee_{k=1}^N \{\mu_{\bar{B}^k}(y)\}. \quad (3)$$

The defuzzification is realized by the COA method defined by the following formula

$$\bar{y} = \frac{\sum_{r=1}^R \bar{y}^r \mu_{B'}(\bar{y}^r)}{\sum_{r=1}^R \mu_{B'}(\bar{y}^r)}, \quad (4)$$

where \bar{y}^r are discretization points of the integrals in the continuous version of the center of area (COA) method, and R is the number of discretization points. Neuro-fuzzy architectures developed so far in the literature are based on the assumption that number of terms R in a formula (4) is equal to the number of rules N . In this paper we do not equalize number of fuzzy rules (N) with number of discretization points (R) and generally it is possible that $R > N$ or $R < N$ or $R = N$. Thanks to that we obtain a more general formula describing flexible neuro-fuzzy systems and we have much more opportunities, adjusting parameters N and R , to design and reduce such systems (for details see [2]).

In the paper we design the structure of system (4) and simultaneously the following parameters of this system are determined in the process of learning: (i) parameters $\bar{x}_{i,k}^A$ and $\sigma_{i,k}^A$ of input fuzzy sets A_i^k , $k = 1, \dots, N$, $i = 1, \dots, n$, and parameters \bar{y}_k^B and σ_k^B of output fuzzy sets B^k , $k = 1, \dots, N$, (ii) discretization points \bar{y}^r , $r = 1, \dots, R$.

3 Evolutionary Learning and Designing of Neuro-fuzzy Systems

In this section a new learning algorithm for designing of flexible neuro-fuzzy inference systems, given by formula (4), is proposed. In the process of evolution we find all parameters described in the previous section and the number of rules, antecedents, inputs, and discretization points of flexible neuro-fuzzy system (4).

3.1 Evolution of Parameters of Neuro-fuzzy Systems

We apply evolutionary strategy (μ, λ) for learning all parameters listed in Section 2. In the system described by formula (4) there are $L = N(2n + 2) + R$

parameters to be found in the evolution process. In a single chromosome, according to the Pittsburgh approach, a complete linguistic model is coded in the following way

$$\begin{aligned} \mathbf{X}_j^{\text{par}} &= \begin{pmatrix} \bar{x}_{1,1}^A, \sigma_{1,1}^A, \dots, \bar{x}_{n,1}^A, \sigma_{n,1}^A, \bar{y}_1^B, \sigma_1^B, \dots \\ \bar{x}_{1,N}^A, \sigma_{1,N}^A, \dots, \bar{x}_{n,N}^A, \sigma_{n,N}^A, \bar{y}_N^B, \sigma_N^B, \\ \bar{y}_1 \dots \bar{y}_R \end{pmatrix}, \\ &= \left(X_{j,1}^{\text{par}}, X_{j,2}^{\text{par}}, \dots, X_{j,L}^{\text{par}} \right) \end{aligned} \quad (5)$$

where $j = 1, \dots, \mu$ for parent population or $j = 1, \dots, \lambda$ for the temporary population. The self-adaptive feature of the algorithm is realized by assigning to each gene a separate mutation range given by the standard deviation

$$\sigma_j^{\text{par}} = \left(\sigma_{j,1}^{\text{par}}, \sigma_{j,2}^{\text{par}}, \dots, \sigma_{j,L}^{\text{par}} \right), \quad (6)$$

where $j = 1, \dots, \mu$ for the parent population or $j = 1, \dots, \lambda$ for the temporary population. Detailed description of the evolutionary strategy (μ, λ) , used to modify the parameters of the flexible neuro-fuzzy system (4), can be found in [3].

3.2 Evolution of Structure of Neuro-fuzzy Systems

Algorithm is based on the evolutionary strategy (μ, λ) and classical genetic algorithm. At the beginning we take the maximum number of rules, antecedents, inputs, and discretization points. In the next step, we reduce our system using the evolutionary strategy. For this purpose we use an extra chromosome $\mathbf{X}_j^{\text{red}}$. Its genes take binary values and indicate which rules, antecedents, inputs, and discretization points are selected. The chromosome $\mathbf{X}_j^{\text{red}}$ is given by

$$\begin{aligned} \mathbf{X}_j^{\text{red}} &= \left(A_1^1, \dots, A_n^1, \text{rule}_1, \dots, A_1^N, \dots, A_n^N, \text{rule}_N, \bar{x}_1, \dots, \bar{x}_n, \bar{y}^1, \dots, \bar{y}^R \right) \\ &= \left(X_{j,1}^{\text{red}}, \dots, X_{j,L^{\text{red}}}^{\text{red}} \right) \end{aligned} \quad (7)$$

where $L^{\text{red}} = n \cdot N + n + N + R$ is the length of the chromosome $\mathbf{X}_j^{\text{red}}$, where $j = 1, \dots, \mu$ for the parent population or $j = 1, \dots, \lambda$ for the temporary population. Its genes indicate which rules (rule_k , $k = 1, \dots, N$), antecedents of rules (A_i^k , $i = 1, \dots, n$, $k = 1, \dots, N$), inputs (\bar{x}_i , $i = 1, \dots, n$), and discretization points (\bar{y}^r , $r = 1, \dots, R$) are taken to the system.

3.3 Initialization of the Algorithm

Initial values of genes in the initial parent population are the following:

- Genes in chromosome $\mathbf{X}_j^{\text{par}}$ corresponding to the input fuzzy sets A_i^k , $k = 1, \dots, N$, $i = 1, \dots, n$, ($\bar{x}_{i,k}^A$ and $\sigma_{i,k}^A$) and genes corresponding to the output fuzzy sets B^k , $k = 1, \dots, N$, (\bar{y}_k^B and σ_k^B) were initialized based on the method described in [5].

- Genes in chromosome $\mathbf{X}_j^{\text{par}}$ corresponding to discretization points \vec{y}^r , $r = 1, \dots, R$, are chosen as random numbers.
- Genes in chromosome $\mathbf{X}_j^{\text{red}}$ are chosen as random numbers (0 or 1).

The components of the mutation range σ_j^{par} , $j = 1, \dots, \mu$, are equal to 1 before the evolution process.

3.4 Fitness Function

In our approach each individual of the parental and temporary populations is represented by a sequence of chromosomes $\langle \mathbf{X}_j^{\text{red}}, \mathbf{X}_j^{\text{par}}, \sigma_j^{\text{par}} \rangle$. The genes of the first chromosome take integer values, whereas the genes of the last two chromosomes take real values. Moreover, in the proposed algorithm we apply:

- Fitness function determined on the basis of the number of rules, number of antecedents, number of inputs, number of discretization points of neuro-fuzzy system and its accuracy, described by following formula

$$\text{ff}(\mathbf{X}_j) = T_a^* \left\{ \begin{array}{l} \frac{\text{ff}_A(\mathbf{X}_j)}{\text{ff}_{A \max}(\mathbf{X}_j)} \cdot (\text{ff}'_{A \max}(\mathbf{X}_j) - \text{ff}'_{A \min}(\mathbf{X}_j)) + \text{ff}'_{A \min}(\mathbf{X}_j), \\ \frac{\text{ff}_B(\mathbf{X}_j)}{\text{ff}_{B \max}(\mathbf{X}_j)} \cdot (\text{ff}'_{B \max}(\mathbf{X}_j) - \text{ff}'_{B \min}(\mathbf{X}_j)) + \text{ff}'_{B \min}(\mathbf{X}_j), \\ \frac{\text{ff}_C(\mathbf{X}_j)}{\text{ff}_{C \max}(\mathbf{X}_j)} \cdot (\text{ff}'_{C \max}(\mathbf{X}_j) - \text{ff}'_{C \min}(\mathbf{X}_j)) + \text{ff}'_{C \min}(\mathbf{X}_j), \\ \frac{\text{ff}_D(\mathbf{X}_j)}{\text{ff}_{D \max}(\mathbf{X}_j)} \cdot (\text{ff}'_{D \max}(\mathbf{X}_j) - \text{ff}'_{D \min}(\mathbf{X}_j)) + \text{ff}'_{D \min}(\mathbf{X}_j), \\ \frac{\text{ff}_E(\mathbf{X}_j)}{\text{ff}_{E \max}(\mathbf{X}_j)} \cdot (\text{ff}'_{E \max}(\mathbf{X}_j) - \text{ff}'_{E \min}(\mathbf{X}_j)) + \text{ff}'_{E \min}(\mathbf{X}_j), \\ \frac{\text{ff}_F(\mathbf{X}_j)}{\text{ff}_{F \max}(\mathbf{X}_j)} \cdot (\text{ff}'_{F \max}(\mathbf{X}_j) - \text{ff}'_{F \min}(\mathbf{X}_j)) + \text{ff}'_{F \min}(\mathbf{X}_j); \\ w_A^{\text{ff}}, w_B^{\text{ff}}, w_C^{\text{ff}}, w_D^{\text{ff}}, w_E^{\text{ff}}, w_F^{\text{ff}} \end{array} \right\}. \quad (8)$$

where T_a^* is an algebraic weighted triangular t-norm (see e.g. [10]). The weighted t-norm aggregates six arguments, each of which is related to another evolution criterion of a chromosome encoding a single neuro-fuzzy system:

- $\text{ff}_A(\mathbf{X}_j)$ denotes the number of discretization points.
- $\text{ff}_B(\mathbf{X}_j)$ denotes the number of inputs.
- $\text{ff}_C(\mathbf{X}_j)$ denotes the number of rules.
- $\text{ff}_D(\mathbf{X}_j)$ denotes the number of antecedents.
- $\text{ff}_E(\mathbf{X}_j)$ denotes the percentage of mistakes in the learning sequence.
- $\text{ff}_F(\mathbf{X}_j)$ denotes the percentage of mistakes in the testing sequence.

Other components in the formula describing fitness function ($\text{ff}_i(\mathbf{X}_j)$, $\text{ff}_{i \max}(\mathbf{X}_j)$, $\text{ff}'_{i \max}(\mathbf{X}_j)$, $\text{ff}'_{i \min}(\mathbf{X}_j)$, $\text{ff}'_{i \min}(\mathbf{X}_j)$, $i = A, \dots, F$) are used for normalization and rescaling arguments of the weighted t-norm to the interval $[\text{ff}'_{i \min}(\mathbf{X}_j), \text{ff}'_{i \max}(\mathbf{X}_j)]$, $i = A, \dots, F$. Their values for the specific simulation problems are given in Table 1.

- Crossover with replacement of the genes for chromosomes $\mathbf{X}_j^{\text{par}}$ and σ_j^{par} .
- Single-point crossover for chromosomes $\mathbf{X}_j^{\text{red}}$, with probability p_c chosen before the evolution process, analogous to the classical genetic algorithm.
- Mutation for chromosomes $\mathbf{X}_j^{\text{red}}$, with probability p_m chosen before the evolution process, analogous to the classical genetic algorithm.

Table 1. Simulation results

		i=A	i=B	i=C	i=D	i=E	i=F
w_i^H		0.60	0.50	0.40	0.30	0.95	1.00
$ff'_{i \min}$		0.10	0.10	0.10	0.10	0.10	0.10
$ff'_{i \max}$		1.00	1.00	1.00	1.00	1.00	1.00
GI	$ff_{i \max}$	4	9	4	36	100	100
	ff_i	4	6	3	15	2.34	0.00
WBC	$ff_{i \max}$	4	9	4	36	100	100
	ff_i	4	5	2	9	2.51	0.98

In our approach the evolutionary algorithm stops when the specified number of generations have evolved. For discussion on other convergence criteria the reader is referred to [11].

4 Experimental Results

The neuro-fuzzy system (4) is simulated on the glass identification problem and Wisconsin breast cancer problem (UCI repository of machine learning databases, available online: <http://ftp.ics.uci.edu/pub/machine-learning-databases/>). In the Glass identification problem all sets are divided into a learning sequence (171 sets) and a testing sequence (43 sets), and in the Wisconsin breast cancer problem all sets are divided into a learning sequence (478 sets) and a testing sequence (205 sets).

In our simulations we use neuro-fuzzy system (4) with Gaussian membership functions, algebraic triangular norms and the Reichenbach S-implication (see e.g. [9]). The evolution process is characterized by the following parameters: $\mu = 10$, $\lambda = 500$, $p_m = 0.077$, $p_c = 0.770$, and the number of generations = 1000 (for details see e.g. [3]).

The experimental results are depicted in Table 1. Neuro-fuzzy system obtained in evolutionary learning for glass identification problem is characterized by 4 discretization points, 3 rules, 6 inputs (RI: refractive index, Mg: magnesium, Si: silicon, K: potassium, Ca: calcium, Ba: barium, Fe: iron) and 15 antecedents. The neuro-fuzzy system obtained in evolutionary learning for Wisconsin breast cancer problem is characterized by 4 discretization points, 2 rules, 5 inputs (clump thickness, uniformity of cell shape, bare nuclei, bland chromatin, normal nucleoli) and 9 antecedents.

5 Summary

In the paper we described an evolutionary algorithm for the automatic designing of neuro-fuzzy systems. The task of the algorithm is not only learning the parameters of the system itself, but also determination of the number of inputs, the number of rules, the number of antecedents, and the number of discretization points. The algorithm automatically selected these numbers (for specific

problems), simultaneously maintaining a good accuracy. Our method allows to significantly automate the process of designing neuro-fuzzy systems.

Acknowledgment

This work was partly supported by the Polish Ministry of Science and Higher Education (Habilitation Project 2007-2010, Polish-Singapore Research Project 2008-2010, Research Project 2008-2011) and the Foundation for Polish Science (TEAM project 2010-2014).

References

1. Casillas, J., Cordon, O., Herrera, F., Magdalena, L. (eds.): *Interpretability Issues in Fuzzy Modeling*. Springer, Heidelberg (2003)
2. Cpałka, K.: A New Method for Design and Reduction of Neuro-Fuzzy Classification Systems. *IEEE Transactions on Neural Networks* 20(4), 701–714 (2009)
3. Cpałka, K.: On evolutionary designing and learning of flexible neuro-fuzzy structures for nonlinear classification. In: *Nonlinear Analysis Series A: Theory, Methods & Applications*, vol. 71. Elsevier, Amsterdam (2009)
4. Czogała, E., Łęski, J.: *Fuzzy and Neuro-Fuzzy Intelligent Systems*. Physica-Verlag, Springer, Heidelberg (2000)
5. Gabryel, M., Rutkowski, L.: Evolutionary Learning of Mamdani-Type Neuro-fuzzy Systems. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) *ICAISC 2006. LNCS (LNAI)*, vol. 4029, pp. 354–359. Springer, Heidelberg (2006)
6. Kumar, M., Stoll, R., Stoll, N.: A robust design criterion for interpretable fuzzy models with uncertain data. *IEEE Trans. Fuzzy Syst.* 14(2), 314–328 (2006)
7. Łęski, J.: A Fuzzy If-Then Rule-Based Nonlinear Classifier. *Int. J. Appl. Math. Comput. Sci.* 13(2), 215–223 (2003)
8. Mitra, S., Hayashi, Y.: Neuro-fuzzy rule generation: survey in soft computing framework. *IEEE Trans. Neural Networks* 11(3), 748–768 (2000)
9. Rutkowski, L.: *Computational Intelligence*. Springer, Heidelberg (2007)
10. Rutkowski, L., Cpałka, K.: Flexible neuro-fuzzy systems. *IEEE Trans. Neural Networks* 14(3), 554–574 (2003)
11. Sivanandam, S.N., Deepa, S.N.: *Introduction to Genetic Algorithms*. Springer, Heidelberg (2008)

An Efficient Adaptive Fuzzy Neural Network (EAFNN) Approach for Short Term Load Forecasting

Juan Du¹, Meng Joo Er¹, and Leszek Rutkowski^{2,3}

¹ School of EEE, Nanyang Technological University, Singapore 639798

² Department of Computer Engineering, Czestochowa University of Technology, al. Armii Krajowej 36, 42-200 Czestochowa, Poland

³ Academy of Management (SWSPiZ), Institute of Information Technology, ul. Sienkiewicza 9, 90-113 Lodz, Poland
lrutko@kik.pcz.czest.pl

Abstract. In this paper, an Efficient Adaptive Fuzzy Neural Network (EAFNN) model is proposed for electric load forecasting. The proposed approach is based on an ellipsoidal basis function (EBF) neural network, which is functionally equivalent to the TSK model-based fuzzy system. EAFNN uses the combined pruning algorithm where both Error Reduction Ratio (ERR) method and a modified Optimal Brain Surgeon (OBS) technology are used to remove the unneeded hidden units. It can not only reduce the complexity of the network but also accelerate the learning speed. The proposed EAFNN method is tested on the actual electrical load data from well-known EUNITE competition data. Results show the proposed approach provides the superior forecasting accuracy when applying in the real data.

Keywords: Fuzzy Neural Network; Short Term load forecasting; Pruning algorithm.

1 Introduction

Short-Term Electric Load Forecasting (STELF) has become increasingly important and plays a crucial role in power system. Therefore, improvements in the accuracy of short-term load forecasts can result in significant financial savings for utilities and co-generators. Various forecasting techniques have been proposed in the last few decades. In the past, most algorithms applied in a load forecast problem relied on statistical analysis, such as auto-regression [1] and time-series methods [2], [3]. However, they are basically linear devices, and the load series they try to explain are known to be distinctly nonlinear functions of the exogenous variables [4], [5].

Recently, the interest of the artificial intelligence techniques has been carried out to handle load forecast problems. Several research groups have studied the use of technologies of Artificial Neural Networks (ANN) [5], [6] and Fuzzy inference system (FIS) [7] for load forecasting. However, there still have some shortcomings to be resolved, such as low training speed, weaker searching capability

for the overall optimal solution and difficulty in selecting the parameters and structure of system. An improved method was proposed to use the fuzzy neural FNN methods to adjust the network structure and parameter regulation. FNN methods have the advantages of both neural networks (e.g., learning abilities, optimization abilities, and connectionist structures) and FIS (e.g., human-like IF-THEN rules thinking and ease of incorporating expert knowledge) [8]. The fuzzy neural system produces appropriate improvement for load forecasting [5], [9], [10]. In this paper, an Efficient Adaptive Fuzzy Neural Network (EAFNN) approach for the short term load forecasting is proposed. It is based on an ellipsoidal basis function (EBF) neural network, which is functionally equivalent to the TSK model-based fuzzy system. The proposed method can automatically adjust its model parameter and structure with new pruning techniques. After using the actual electrical load data to develop the forecasting model, the proposed method provides the excellent forecasting accuracy.

This paper is organized as follows. In Section 2, we describe learning algorithm of EAFNN. Section 3 describes the load which is used for training the EAFNN method and the task of short term load forecasting. In addition, results and observation of different models are showed in Section 3. Finally, Section 4 presents the conclusions.

2 Algorithm of EAFNN

2.1 Structure of EAFNN

EAFNN is a multi-input multi-output (MIMO) system which has inputs and outputs. With the structure based on the EBF neural network, it is functionally equivalent to the Takago-Sugeno-Kang (TSK) model-based fuzzy system. The functions of the various nodes in each of the four layers are depicted in Fig. 1.

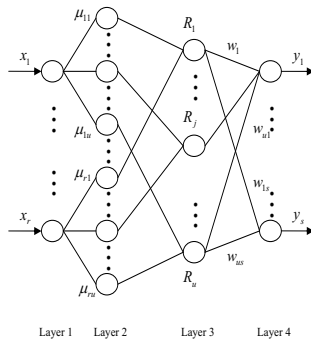


Fig. 1. Architecture of the EAFNN

Layer 1: Each node in layer 1 represents an input linguistic variable.

Layer 2: Each node in layer 2 represents a membership function (MF), which is governed by a Gaussian function:

$$\mu_{ij}(x_i) = \exp\left[-\frac{(x_i - c_{ij})^2}{2\sigma_{ij}^2}\right] \quad (1)$$

where $i = 1, 2, \dots, r$, $j = 1, 2, \dots, u$ and μ_{ij} is the j th membership function of the i th input variable x_i , c_{ij} is the center of the j th Gaussian membership of x_i , σ_{ij} is the width of the j th Gaussian membership of x_i .

Layer 3: Each node in layer 3 represents a possible IF-part for fuzzy rules. For the j th rule, its output is given by

$$\phi_j(x_1, x_2, \dots, x_r) = \exp\left[-\sum_{i=1}^r \frac{(x_i - c_{ij})^2}{2\sigma_{ij}^2}\right] = \exp[-md^2(j)] \quad (2)$$

$$j = 1, 2, \dots, u$$

where md can be regarded as a function of regularized Mahalanobis distance (M-distance).

Layer 4: Each node in layer 4 represents the output variable as a weighted summation of incoming signals and is given by

$$y(x_1, x_2, \dots, x_r) = \sum_{j=1}^u \omega_j \cdot \phi_j \quad (3)$$

where y is the value of an output variable and ω_j is the THEN-part (consequent parameters) or connection weight of the j th rule.

For the TSK model, the consequents are the polynomials in the input variables and are given by

$$\omega_{ij} = \alpha_{0j} + \alpha_{1j}x_1 + \dots + \alpha_{rj}x_r \quad j = 1, 2, \dots, u \quad (4)$$

Suppose that u fuzzy rules are generated for n observations. Equation (3) can be rewritten as follows in a more compact form

$$Y = W\phi \quad (5)$$

2.2 Learning Algorithm of EAFNN

I. Adding a Fuzzy Rule

EAFNN begins with no fuzzy rules. Two criteria are used in order to generate a new rule, namely system errors and ε -Completeness.

For each observation (X_k, t_k) , $k = 1, 2, \dots, n$, where n is the number of total training data, X_k is the k th desired output, compute the overall dynamic EAFNN output y_k of the existing structure using Equation (3). If

$$\|e_k\| = \|t_k - y_k\| > k_e \quad (6)$$

a new rule should be considered. Here, k_e is a predefined threshold that decays during the learning process.

The second criterion is ε -Completeness of fuzzy rules. When an observation (X_k, d_k) , $k = 1, 2, \dots, n$, enters the system, we calculate the M-distance $md_k(j)$ between X_k and the center C_j ($j = 1, 2, \dots, n$) of the existing EBF units. If

$$md_{k,\min} = md_k(J) > k_d(md_k(j)) \quad (7)$$

where k_d is a prespecified threshold that decays during the learning process, this implies that the existing system is not satisfied with ε -Completeness and a new rule should be considered.

II. Parameter Adjustment

A new Gaussian membership function is allocated whose width and center is set as follows:

$$c_{i(u+1)} = x_i^k \quad (8)$$

$$\sigma_i = \frac{\max\{|c_i - c_{i-1}|, |c_i - c_{i+1}|\}}{\sqrt{\ln(1/\varepsilon)}} \quad i = 1, 2, \dots, m \quad (9)$$

where c_{i-1} and c_{i+1} are the two centers of neighboring membership function of i th membership function.

Suppose that u fuzzy rules are generated according to the two criteria stated above for n observation with r number of input variables, the outputs of the N nodes can be obtained according to Equation (3). Writing in matrix form:

$$W\phi = Y \quad (10)$$

where $W \in \mathfrak{R}^{u(r+1)}$, $\phi \in \mathfrak{R}^{u(r+1) \times n}$, $Y \in \mathfrak{R}^n$

Assume that the desired output is $T = (t_1, t_2, \dots, t_n) \in \mathfrak{R}^n$. The problem of determining the optimal parameters W^* can be formulated as a linear problem of minimizing $\|W\phi - T\|_2$ and W^* is determined by the pseudoinverse technique

$$W^* = T(\phi^T \phi)^{-1} \phi^T \quad (11)$$

III. Pruning a Fuzzy Rule

The performance of an EAFNN not only depends on the number and location (in the input space) of the centers but also depends on determination of the network weights. In this paper, the Error reduction ratio method (ERR) and the modified Optimal Brain Surgeon (OBS) method are utilized as pruning strategy. ERR is utilized to select significant rules and a modified OBS relies on the sensitivity analysis of the weight parameter.

(1) Error Reduction Ratio Method

Given n input-output pairs $\{X(k), t(k), k = 1, 2, \dots, n\}$, consider Equation 5 in the following compact form:

$$D = H\theta + E \quad (12)$$

where $D = T^T \in \mathfrak{R}^n$ is the desired output, $H = \phi^T = (h_1 \dots h_v) \in \mathfrak{R}^{n \times v}$ are the regressors, with $v = u \times (r + 1)$, $\theta = W^T \in \mathfrak{R}^v$ contains real parameters and

$E \in \mathfrak{R}^n$ is the error vector that is assumed to be uncorrelated with the regressors $h_i (i = 1, 2, \dots, v)$.

The matrix H can be transformed into a set of orthogonal basis vectors if its row number is larger than the column number. H is decomposed into

$$H = PN \tag{13}$$

where $P = (p_1, p_2, \dots, p_v) \in \mathfrak{R}^{n \times v}$ has the same dimension as H with orthogonal columns and $N \in \mathfrak{R}^{n \times v}$ is an upper triangular matrix.

Substituting Equation (13) into Equation (12) yields

$$D = PN\theta + E = PG + E \tag{14}$$

The linear least square (LLS) solution of G is given by $G = (P^T P)^{-1} P^T D$

$$g_i = \frac{p_i^T D}{p_i^T p_i} \quad i = 1, 2, \dots, v. \tag{15}$$

As p_i and p_j are orthogonal for $i \neq j$, the sum of squares of D is given as follows:

$$D^T D = \sum_{i=1}^v g_i^2 p_i^T p_i + E^T E \tag{16}$$

Substituting g_i by Equation (15), and ERR due to p_i is defined as

$$err_i = \frac{(p_i^T D)^2}{P_i^T p_i D^T D} \quad i = 1, 2, \dots, v \tag{17}$$

The above equation offers a simple and effective means of seeking a subset of significant regressors. Define the ERR matrix $\Delta = (\rho_1, \rho_2, \dots, \rho_u) \in \mathfrak{R}^{(r+1) \times u}$ whose elements are obtained from Equation (17) and the j th rule column of Δ as the total ERR corresponding to the j th rule. Furthermore, define

$$\eta_j = \sqrt{\frac{\rho_j^T \rho_j}{r + 1}} \quad j = 1, 2, \dots, u \tag{18}$$

then η_j represents the significance of the j th rule. If $\eta_j < k_{err}$, $j = 1, 2, \dots, u$, where k_{err} is a prespecified threshold, then the j th rule is deleted.

(2) Modified OBS Pruning Method

In the learning process, the network reaches a local minimum in error and the third and all higher order terms can be ignored. The cost function ΔE of the system can be approximated simply as

$$\Delta E \approx \frac{1}{2} \Delta w^T H \Delta w \tag{19}$$

where Δw is the increase of weight, H is the Hessian matrix. The cost function also can be defined as the squared error as follows:

$$E(k) = \frac{1}{2} \sum_{i=1}^n [d(i) - p(i)^T \theta]^2 \quad (20)$$

where $p(i)^T = h_i$, $\theta = w$.

The Hessian matrix can be written as follows:

$$\frac{\partial^2 E}{\partial \theta^2} = H = \sum_{i=1}^n p(i)p^T(i) \quad (21)$$

Because the dimensions of Hessian matrix are equal to the number of hidden units in the network, S_i can be computed through w_i as follows:

$$S_i = \frac{\bar{w}_i^2}{2[H^{-1}]_{i,i}} \quad (22)$$

$$\bar{w}_i = \frac{\sum_{j=1}^m w_{ij}}{m} \quad (23)$$

where m is the number of weights which connect with the i th unit. [12] demonstrate that the premise behind this measure is that the higher this sum, the more important that unit is to the network. The smaller the value of the saliencies S , the less important is the neuron.

3 Short Term Load Forecasting

In this study, to evaluate the performance of the proposed load forecasting scheme, the EAFNN method is tested with data obtained from well-known EU-NITE competition data, to predict the maximum daily values of electrical loads. The load data includes electricity load demand recorded every half hour from January 1998 to January 1999. The training data used in this study covers the period from January 1998 to December 1998. The task is to supply the prediction of maximum daily values of electrical loads for January 1999.

To forecast the peak load L_i of the i th day, two load forecasting models are used. Model I is a forecasting model which inputs data is without temperature vectors and Model II is a forecasting model which inputs data is with temperature vectors. For the first model, the training input data includes seven load vectors: $L_{i-1}, L_{i-2}, L_{i-3}, L_{i-4}, L_{i-5}, L_{i-6}, L_{i-7}$. The training input of the second model has seven load vectors and two more temperature vectors: T_{i-1} and T_i . Mean Absolute Percentage Error (MAPE) and Maximal Error (ME) are used to evaluate the performance of the forecasting model as follows:

$$MAPE = \frac{1}{N} \sum_{i=1}^N \frac{|L_{actual,i} - L_{predicted,i}|}{L_{actual,i}} \times 100 \quad (24)$$

$$MAE = Max |L_{actual,i} - L_{predicted,i}| \quad (25)$$

where $L_{actual,i}$ and $L_{predicted,i}$ are the real and the forecast data of maximum daily electrical load on the i th day of the year 1999 respectively. N is the number of days in January 1999. When training the EAFNN models, there are some parameters, which influence the performance of the model, need to choose. In this paper, the training parameters are chosen as: $k_e = 0.9918, k_d = 0.9765, k_{err} = 0.01, Saliency_{exp} = 0.000004$.

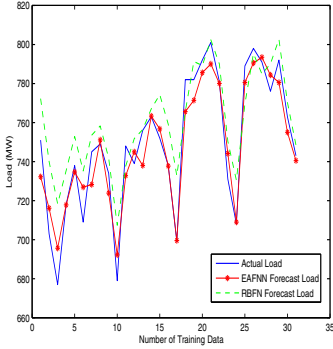


Fig. 2. Comparison In Model I

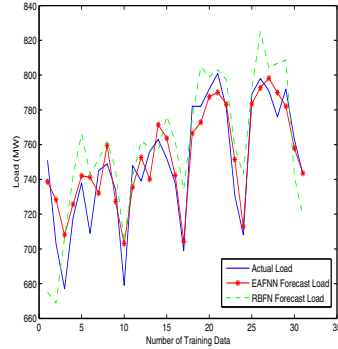


Fig. 3. Comparison In Model II

Fig. 2 and Fig. 3 show the comparison of the performance between RBF neural network based load forecaster and the proposed EAFNN based load forecaster. The RBF neural network used in this paper is based on gradient descent learning method. The number of hidden neurons is chosen as 10 neurons in advance. The same training set and validation set are utilized for the RBF neural network. In Table 1, the ME and MAPE comparison are made between the RBF and EAFNN based load forecaster. From Table 1 we can see that the EAFNN load forecaster has better performance for load forecast both in ME and MAPE. Moreover, for RBF forecasting model, the performance of the model using temperature input data is not as good as the model using only historical load information. It is because the fuzzy correlation between the temperature and the load are limited. But the EAFNN has good performance in both models because it can adapt to more complex data.

Table 1. Comparison of the performance of RBF and EAFNN forecasters

	Model I		Model II	
	ME (MW)	MAPE (%)	ME (MW)	MAPE (%)
RBF	41.3321	2.047	75.9176	2.878
EAFNN	25.0281	1.545	32.2087	1.552

4 Conclusion

An Efficient Adaptive Fuzzy Neural Network based forecasting scheme is developed for the short-term hourly electric load. In this algorithm, the determination of the fuzzy rules and adjustment of the premise and consequent parameters in fuzzy rules can be achieved simultaneously. In the determination of the fuzzy rules, the fuzzy rules can be allocated and removed automatically without predefining. The forecasting scheme uses the learning algorithm of the proposed EAFNN. The proposed short-term load forecasting method is tested on the actual electrical load data from EUNITE competition data. Results show the proposed approach has better performance in two short-term load forecast models.

Acknowledgments

This work was partly supported by the Polish Ministry of Science and Higher Education Polish-Singapore Research Project 2008-2010 and Research Project 2008-2011, and the Foundation for Polish Science TEAM project 2010-2014.

References

1. Vemuri, S., Huang, W.L., Nelson, D.L.: On-Line Algorithms for Forecasting Hourly Loads of an Electrical Utility. *IEEE Transactions on Power Apparatus and Systems* 100(8), 3775–3784 (1981)
2. Moghram, I., Ruhman, S.: Analysis and Evaluation Five of Load Forecasting Techniques. *IEEE Transactions on Power Systems* 14(4), 1484–1491 (1989)
3. Hagan, M.T., Behr, S.M.: The Time Series Approach to Short Term Load Forecasting. *IEEE Transactions on Power Systems* 2(3), 832–837 (1987)
4. Hippert, H.S., Pedreira, C.E., Souza, R.C.: Neural networks for short-term load forecasting: a review and evaluation. *IEEE Transactions on Power Apparatus and Systems* 16(1), 44–55 (2001)
5. Campbell, P.R.J.: A Hybrid Modelling Technique for Load Forecasting. In: *Electrical Power Conference, EPC 2007*, pp. 435–439. IEEE, Canada (2007)
6. Saini, L.M., Soni, M.K.: Power Systems. *IEEE Transactions on Power Apparatus and Systems* 17(3), 907–912 (2002)
7. Mori, H., Kobayashi, H.: Optimal Fuzzy Inference for Short-Term Load Forecasting. *IEEE Transactions on Power Systems* 11(1), 390–396 (1996)
8. Meneganti, M., Saviello, F., Tagliaferri, R.: Fuzzy neural networks for classification and detection of anomalies. *IEEE Transactions on Neural Networks* 9, 848–861 (1998)
9. Papadakis, S.E., Theocharis, J.B., Kiartzis, S.J., Bakirtzis, A.G.: A Novel Approach to Short-Term Load Forecasting Using Fuzzy Neural Networks. *IEEE Transactions on Power Systems* 13(2), 480–492 (1998)

10. Bakirtzis, A.G., Theocharis, J.B., Kiartzis, S.J., Satsios, K.J.: Short term load forecasting using fuzzy neural networks. *IEEE Transactions on Power Systems* 10(3), 1518–1524 (1995)
11. Hassibi, B., Stork, D.G.: Second-order derivatives for network pruning: Optimal brain surgeon. In: *PAdvances in Neural Information Processing*, vol. 4, pp. 665–684. Morgan Kaufman, Los Altos (1993)
12. Messer, K., Kittler, J.: Choosing an optimal neural network size to aid search through a large image database. In: *Proceedings of the British Machine Vision Conference, BMVC 1998*, pp. 235–244 (1998)

Fault Diagnosis of an Air-Handling Unit System Using a Dynamic Fuzzy-Neural Approach

Juan Du¹, Meng Joo Er¹, and Leszek Rutkowski^{2,3}

¹ School of EEE, Nanyang Technological University, Singapore 639798

² Department of Computer Engineering, Czestochowa University of Technology, al. Armii Krajowej 36, 42-200 Czestochowa, Poland

³ Academy of Management (SWSPiZ), Institute of Information Technology, ul. Sienkiewicza 9, 90-113 Lodz, Poland
lrutko@kik.pcz.czest.pl

Abstract. This paper presents a diagnostic tool to be used to assist building automation systems for sensor health monitoring and fault diagnosis of an Air-Handling Unit (AHU). The tool employs fault detection and diagnosis (FDD) strategy based on an Efficient Adaptive Fuzzy Neural Network (EAFNN) method. EAFNN is a Takagi-Sugeno-Kang (TSK) type fuzzy model which is functionally equivalent to the Ellipsoidal Basis Function (EBF) neural network neurons. An EAFNN uses the combined pruning algorithm where both Error Reduction Ratio (ERR) method and a modified Optimal Brain Surgeon (OBS) technology are used to remove the unneeded hidden units. Simulation works show the proposed diagnosis algorithm is very efficient which can not only reduce the complexity of the network but also accelerate the learning speed.

Keywords: fuzzy neural network; fault diagnosis ; Air-handling unit.

1 Introduction

It is generally accepted that the performance of HVAC systems often falls short of expectations [1]. In order to investigate methods of FDD and evaluate the most suitable modeling approaches, the international research project entitled International Energy Agency (IEA) annex 25 and annex 34 has been made. A number of methodologies and procedures for optimizing real-time performances, automated fault detection and fault isolation were developed in the Annexes, which concentrated on computer-aided fault detection and diagnosis, [2].

Artificial intelligence methods, which using neural networks and fuzzy set theory, have undergone rapid development in fault diagnosis for HVAC systems [1]. Previous research has investigated the monitoring and diagnosis of HVAC systems using solely fuzzy logic theory or neural network algorithm [3]. However, in the above studies, the fuzzy system and neural network designs have limitations. These are difficult and time-consuming tasks. To improve the accuracy of single-algorithm applications, various fusions of fuzzy logic and neural

networks, the so-called hybrid intelligent architecture, have been developed for better performance in decision making systems [4]. However, very few fuzzy-neural approaches are utilized for diagnosis of HVAC systems, while such kind of approaches are very well developed for other application fields such as for nuclear power plants, marine and power systems.

In HVAC operation, AHU plays an essential role for supplying treated air with specified temperature to the conditioned space [5]. In this paper, an EAFNN approach is introduced to deal with the problem of faults diagnosis in data generated by a pilot variable air volume (VAV) AHU system. EAFNN has the salient features of no predetermination of fuzzy rules and data space clustering and automatic and simultaneous structure and parameters learning automatically and simultaneously by online hierarchical learning algorithm.

This paper is organized as follows. Section 2 gives a brief description of the AHU and the residuals used in the fault diagnosis. The four faults and domain residuals are then described. The description of EAFNN and its learning algorithm are presented in Section 3. Section 4 shows the simulation results and some comparative studies with other learning algorithms. Lastly, conclusions are drawn in Section 5.

2 System and Model Description

A VAV AHU pilot plant was built for experimental purpose. As shown in Fig. 1, number 1, 2, 3 and 4 indicate the components of computer controller, air-conditioning pilot plant, signal process board and signal transmission cables, respectively. All motors (fans, pumps and compressors) in the system are controlled by variable speed drives (VSD).

A simplified system lay out diagram of AHU is shown in Fig.2. It can be seen from the figure that air enters the AHU through the outdoor air damper then mix with air passing through the re-circulation air damper. Air exit the mixed air plenum passes through the cooling coils. After being conditioned in the coils, the air is then distributed to the zones through the supply air duct. The supply air temperature is measured downstream of the supply fan and a static pressure sensor is settled on the supply air duct to measure the main supply air pressure. The objectives of the AHU are to maintain the supply air temperature at a constant set point value of 17.5°C by controlling the water pump and the supply air pressure at a constant set point value of 160 Pa(1.0 in. of water) by controlling the rotational speed of the supply fan.

There are two types of faults, namely complete faults (or abrupt failures) and performance degradations. Complete failures are severe and abrupt faults. Performance degradation is gradually evolving faults. In [3], the authors used seven different equipment and instrumentation faults to represent complete failures of various components in the AHU. By the limitation of real pilot plant of AHU system, here we only choose four main faults: two equipment faults and two sensor faults. The faults are shown in Fig. 2.



Fig. 1. Pilot Plant of VAV AHU

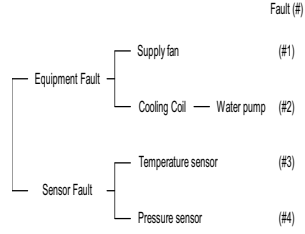


Fig. 2. Main faults in AHU

3 Algorithm of EAFNN

3.1 Structure of EAFNN

EAFNN is a multi-input multi-output (MIMO) system which has inputs and outputs. With the structure based on the EBF neural network, it is functionally equivalent to the TSK model-based fuzzy system.

Layer 1: Each node in layer 1 represents an input linguistic variable.

Layer 2: Each node in layer 2 represents a membership function (MF), which is governed by a Gaussian function.

$$\mu_{ij}(x_i) = \exp\left[-\frac{(x_i - c_{ij})^2}{2\sigma_{ij}^2}\right] \tag{1}$$

where $i = 1, 2 \dots r$, $j = 1, 2 \dots u$ and μ_{ij} is the j th membership function of the i th input variable x_i , c_{ij} is the center of the j th Gaussian membership of x_i , σ_{ij} is the width of the j th Gaussian membership of x_i .

Layer 3: Each node in layer 3 represents a possible IF-part for fuzzy rules. For the j th rule, its output is given by

$$\phi_j(x_1, x_2, \dots, x_r) = \exp\left[-\sum_{i=1}^r \frac{(x_i - c_{ij})^2}{2\sigma_{ij}^2}\right] = \exp[-md^2(j)] \tag{2}$$

$j = 1, 2, \dots, u$

where md can be regarded as a function of regularized Mahalanobis distance (M-distance).

Layer 4: Each node in layer 4 represents the output variable as a weighted summation of incoming signals and is given by

$$y(x_1, x_2, \dots, x_r) = \sum_{j=1}^u \omega_j \cdot \phi_j \tag{3}$$

where y is the value of an output variable and ω_j is the THEN-part (consequent parameters) or connection weight of the j th rule.

Equation (3) can be rewritten as follows in a more compact form

$$Y = W\phi \quad (4)$$

3.2 Learning Algorithm of EAFNN

I. Adding a Fuzzy Rule

EAFNN begins with no fuzzy rules. Two criteria are used in order to generate a new rule, namely system errors and ε -Completeness.

For each observation (X_k, t_k) , $k = 1, 2, \dots, n$, where n is the number of total training data, X_k is the k th desired output, compute the overall dynamic EAFNN output y_k of the existing structure using Equation (3). If

$$\|e_k\| = \|t_k - y_k\| > k_e \quad (5)$$

a new rule should be considered. Here, k_e is a predefined threshold that decays during the learning process.

The second criterion is ε -Completeness of fuzzy rules. When an observation (X_k, d_k) , $k = 1, 2, \dots, n$, enters the system, we calculate the M-distance $md_k(j)$ between X_k and the center C_j ($j = 1, 2, \dots, n$) of the existing EBF units. If

$$md_{k,\min} = md_k(J) > k_d(md_k(j)) \quad (6)$$

where k_d is a prespecified threshold that decays during the learning process. This implies that the existing system is not satisfied with ε -Completeness and a new rule should be considered.

II. Parameter Adjustment

A new Gaussian membership function is allocated whose width and center is set as follows:

$$c_{i(u+1)} = x_i^k \quad (7)$$

$$\sigma_i = \frac{\max\{|c_i - c_{i-1}|, |c_i - c_{i+1}|\}}{\sqrt{\ln(1/\varepsilon)}} \quad i = 1, 2, \dots, m \quad (8)$$

where c_{i-1} and c_{i+1} are the two centers of neighboring membership function of i th membership function.

Suppose that u fuzzy rules are generated according to the two criteria stated above for n observation with r number of input variables, the outputs of the N nodes can be obtained according to Equation (3). Writing in matrix form:

$$W\phi = Y \quad (9)$$

where $W \in \mathfrak{R}^{u(r+1)}$, $\phi \in \mathfrak{R}^{u(r+1) \times n}$, $Y \in \mathfrak{R}^n$

Assume that the desired output is $T = (t_1, t_2, \dots, t_n) \in \mathfrak{R}^n$. The problem of determining the optimal parameters W^* can be formulated as a linear problem of minimizing $\|W\phi - T\|_2$ and W^* is determined by the pseudoinverse technique

$$W^* = T(\phi^T \phi)^{-1} \phi^T \quad (10)$$

III. Pruning a Fuzzy Rule

The performance of an EAFNN not only depends on the number and location (in the input space) of the centers but also depends on determination of the network weights. In this paper, Error reduction ratio method (ERR) and a modified Optimal Brain Surgeon (OBS) method are utilized as pruning strategy.

(1) Error Reduction Ratio Method

Given n input-output pairs $\{X(k), t(k), k = 1, 2, \dots, n\}$, consider Equation (4) in the following compact form:

$$D = H\theta + E \quad (11)$$

where $D = T^T \in \mathfrak{R}^n$ is the desired output, $H = \phi^T = (h_1 \dots h_v) \in \mathfrak{R}^{n \times v}$ are the regressors, with $v = u \times (r + 1)$, $\theta = W^T \in \mathfrak{R}^v$ contains real parameters and $E \in \mathfrak{R}^n$ is the error vector that is assumed to be uncorrelated with the regressors $h_i (i = 1, 2, \dots, v)$.

The matrix H can be transformed into a set of orthogonal basis vectors if its row number is larger than the column number. H is decomposed into

$$H = PN \quad (12)$$

where $P = (p_1, p_2, \dots, p_v) \in \mathfrak{R}^{n \times v}$ has the same dimension as H with orthogonal columns and $N \in \mathfrak{R}^{n \times v}$ is an upper triangular matrix.

Substituting Equation (12) into Equation (11) yields

$$D = PN\theta + E = PG + E \quad (13)$$

The linear least square (LLS) solution of G is given by $G = (P^T P)^{-1} P^T D$

$$g_i = \frac{p_i^T D}{p_i^T p_i} \quad i = 1, 2, \dots, v. \quad (14)$$

As p_i and p_j are orthogonal for $i \neq j$, the sum of squares of D is given as follows:

$$D^T D = \sum_{i=1}^v g_i^2 p_i^T p_i + E^T E \quad (15)$$

Substituting g_i by Equation (14), and ERR due to p_i is defined as

$$err_i = \frac{(p_i^T D)^2}{P_i^T p_i D^T D} \quad i = 1, 2, \dots, v \quad (16)$$

The above equation offers a simple and effective means of seeking a subset of significant regressors. Define the ERR matrix $\Delta = (\rho_1, \rho_2, \dots, \rho_u) \in \mathfrak{R}^{(r+1) \times u}$ whose elements are obtained from Equation (16) and the j th rule column of Δ as the total ERR corresponding to the j th rule. Furthermore, define

$$\eta_i = \sqrt{\frac{\rho_j^T \rho_j}{r+1}} \quad j = 1, 2, \dots, u \quad (17)$$

then η_j represents the significance of the j th rule. If $\eta_j < k_{err}$, $j = 1, 2, \dots, u$, where k_{err} is a prespecified threshold, then the j th rule is deleted.

(2) Modified OBS Pruning Method

In the learning process, the network reaches a local minimum in error and the third and all higher order terms can be ignored. The cost function ΔE of the system can be approximated simply as

$$\Delta E \approx \frac{1}{2} \Delta w^T H \Delta w \tag{18}$$

where Δw is the increase of weight, H is the Hessian matrix. The cost function also can be defined as the squared error as follows:

$$E(k) = \frac{1}{2} \sum_{i=1}^n [d(i) - p(i)^T \theta]^2 \tag{19}$$

where $p(i)^T = h_i$, $\theta = w$.

Then the Hessian matrix can be written as follows:

$$\frac{\partial^2 E}{\partial \theta^2} = H = \sum_{i=1}^n p(i) p^T(i) \tag{20}$$

Because the dimension of Hessian matrix is equal to the number of hidden units in the network, S_i can be computed through w_i as follows:

$$S_i = \frac{\bar{w}_i^2}{2[H^{-1}]_{i,i}} \tag{21}$$

$$\bar{w}_i = \frac{\sum_{j=1}^m w_{ij}}{m} \tag{22}$$

where m is the number of weights which connect with the i th unit. The smaller the value of the saliencies S , the less important is the neuron.

4 Application of EAFNN to Fault Diagnosis

Faults in the AHU system are diagnosed by inputting vectors to the trained EAFNN. The training vectors are obtained by introducing faults in an AHU pilot plant and recording the subsequent response of the system. Data are calculated using system variables measured 150 seconds after the system begins running. In this study, 2047 data are used for training and testing of EAFNN. EAFNN is tested by using a jack-knife method [6]. In the jack-knife method, one half of the sample patterns are selected randomly from the database for training the EAFNN. Subsequently, the other half of the sample patterns is used for testing the trained EAFNN. Fig. 3 shows the growth of fuzzy rules. The assessment of the prediction performance of the different soft computing models was performed by quantifying the prediction obtained on an independent data set. The Root Mean

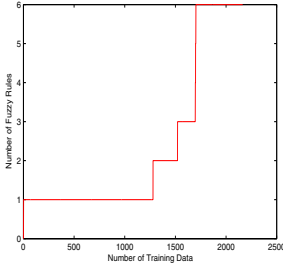


Fig. 3. Fuzzy rule generation

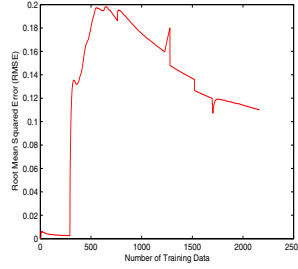


Fig. 4. The performance of DFNN

Square Error (RMSE) is used to evaluate the performance of fault diagnosis scheme. RMSE of EAFNN can be seen in Fig. 4.

$$RMSE = \sqrt{\frac{\sum_{k=1}^n (d(k) - y(k))^2}{n}} \tag{23}$$

To demonstrate the effectiveness of the novel algorithm, the results of the proposed method are compared with other earlier works, such as BP-based method, RBF-based method and GFNN method [7]. The number of hidden neurons and RBF function neurons need to be selected in advance for the BPNN and the RBFN separately. In this study, we choose the BPNN architecture as 4*10*5, corresponding to the number of inputs, the number of hidden neurons and the number of outputs respectively. A log-sigmoid activation function is used for both the hidden and output layers. The RBFN utilized in this research has 15 hidden neurons. Each of the 15 neurons in the hidden layer applies an activation function which is a function of the Euclidean distance between the input and prototype vector. Each hidden neuron contains its own prototype vector as a parameter. A comparison of information of fault diagnosis performances of BPNN, RBFNN, GFNN and EAFNN is shown in Table 1.

From Table 1, the GFNN and EAFNN have faster learning speed than the BPNN and RBFN because of the non-iterative learning. Moreover, The BPNN-based and RBFN-based methods need to predetermine the number of hidden layer neurons or the RBF neurons, while the GFNN and EAFNN can decide the system structure automatically without predetermination. The GFNN-based method and EAFNN-based method have less hidden neurons, which means they have better generalization ability and they can fit for different kinds of faults.

Table 1. Performance comparisons of different methods

	BPNN	RBFN	GFNN	EAFNN
Hidden layer neurons	10	15	4	6
Computing time (s)	679.453	390.156	98.453	101.345
Final RMSE	0.2585	0.1410	0.12	0.1101
Average RMSE	-	-	0.0634	0.0526

Finally, The RMSE of training and testing of the EAFNN are better than those of the other methods.

5 Conclusion

In this paper, the EAFNN is developed and applied to fault diagnosis in an AHU system. Firstly, four system variables which describe the dominant symptoms of fault modes of operation are chosen. Consequently, the proposed EAFNN is trained by dominant symptoms and the faults. The structure and parameter identification of EAFNN are done automatically and simultaneously without partitioning the input space and selecting initial parameters a priori. From the performance of the EAFNN in a real pilot, it is evident that the proposed EAFNN has high accuracy with a compact structure, and also has fast and efficient learning. The proposed approach can also be extended for on-line learning and can also be used to consider additional faults for more complex HVAC systems. We are currently developing EAFNN for AHU fault diagnosis in the condition of complex.

Acknowledgments

This work was partly supported by the Polish Ministry of Science and Higher Education Polish-Singapore Research Project 2008-2010 and Research Project 2008-2011, and the Foundation for Polish Science TEAM project 2010-2014.

References

1. Lee, W.Y., Park, C., Kelly, G.E.: Fault detection of an air-handling unit using residual and recursive parameter identification methods. *ASHRAE Transaction* 102, 528–539 (1996)
2. House, J.M., Kelly, G.E.: Overview of building diagnostics. In: *Overview of building diagnostics*, pp. 1–9 (2000)
3. Lee, W.Y., House, J.M., Park, C., Kelly, G.E.: Fault diagnosis of an air-handling unit using artificial neural networks. *ASHRAE Transaction* 102, 540–549 (1996)
4. Kuo, H.-C., Chang, H.-K.: A new symbiotic evolution-based fuzzy-neural approach to fault diagnosis of marine propulsion systems. *Engineering Applications of Artificial Intelligence* 17, 919–930 (2004)
5. House, J.M., Smith, T.F.: Optimal control of building and HVAC systems. *Optimal Control of Building and HVAC Systems* 6, 4326–4330 (1995)
6. Kim, J.K., Park, H.W.: Statistical textural feature for detection of microcalcification in digitized mammograms. *IEEE Transactions on Medical Imaging* 18, 231–238 (1999)
7. Wu, S.Q., Er, M.J.: A fast approach for automatic generation of fuzzy rules by generalized dynamic fuzzy neural networks. *IEEE Transactions on Fuzzy Systems* 9, 578–594 (2001)

An Interpretation of Intuitionistic Fuzzy Sets in the Framework of the Dempster-Shafer Theory

Ludmila Dymova and Pavel Sevastjanov

Institute of Comp.& Information Sci., Czestochowa University of Technology,
Dabrowskiego 73, 42-200 Czestochowa, Poland
sevast@icis.pcz.pl

Abstract. A new interpretation of Intuitionistic Fuzzy Sets in the framework of the Dempster-Shafer Theory is proposed. Such interpretation allows us to reduce all mathematical operations on the Intuitionistic Fuzzy values to the operations on belief intervals. The proposed approach is used for the solution of Multiple Criteria Decision Making (*MCDM*) problem in the Intuitionistic Fuzzy setting.

1 Introduction

Intuitionistic Fuzzy Set (*A-IFS*) proposed by Atanassov [1], is one of the possible generalizations of fuzzy sets theory and appears to be relevant and useful in some applications.

The concept of *A-IFS* is based on the simultaneous consideration of membership μ and non-membership ν of an element of a set to the set itself [1]. By definition $0 \leq \mu + \nu \leq 1$. The most important applications of *A-IFS* is the decision making problem [4]. In the framework of *A-IFS*, the decision making problem may be formulated as follows.

Let $X = \{x_1, x_2, \dots, x_m\}$ be a set of alternatives, $A = \{a_1, a_2, \dots, a_n\}$ be a set of local criteria, $W = \{w_1, w_2, \dots, w_n\}$ be the weights of local criteria. If μ_{ij} is the degree to which x_i satisfies the criterion a_j and ν_{ij} is the degree to which x_i does not satisfy a_j then alternative x_i may be presented by its characteristics as follows: $x_i = \{(w_1, < \mu_{i1}, \nu_{i1} >), (w_2, < \mu_{i2}, \nu_{i2} >), \dots, (w_n, < \mu_{in}, \nu_{in} >)\}$, $i = 1, \dots, m$.

To obtain the resulting alternative's evaluation, usually different real valued score functions are used [4,10]. Deschrijver and Kerre [8] established some interrelations between *A-IFS* and theories modeling imprecision such as Interval Valued Fuzzy Sets, Type 2 Fuzzy Sets and Soft Sets. In the current paper, we show that there exists also a strong link between *A-IFS* and the Dempster-Shafer Theory of evidence (*DST*). In the papers [17,18] proposed a set of *MCDM* models in *A-IFS* setting based on some operations on *IFVs* defined in [2]. The common limitation of these aggregation operators is that the weights w_i , $i=1$ to n , are supposed to be real values, although, in general, they may be presented by *IFVs* too.

In the current paper, we propose a new approach based on *DST*, which makes it possible to aggregate local criteria without this limitation. If the final scores

of alternatives are presented by *IFVs*, the problem of comparison of such values arises. Therefore, the specific methods were developed to compare *IFVs* [4,10], but they have some limitations.

In the current paper, we show that it is possible to get over these limitations using the *DST* semantics for *A-IFS*. In this *DST/IFS* approach, the problem of *IFVs* comparison reduces to the comparison of belief intervals and is solved using the methods of interval analysis.

Summarizing, we note that there exist two important problems in *MCDM* in the intuitionistic fuzzy setting: aggregation of local criteria without intermediate defuzzification in the case when criteria and their weights are *IFVs*; comparison of *IF* valued scores of alternatives basing on the degree to which one *IFV* is greater / smaller than the other. In the current paper we propose an approach to the solution of these problems based on the interpretation of *A-IFS* in the framework of the *DST*.

The rest of this paper is set out as follows. In Section 2, we present our interpretation of *A-IFS* in terms of the *DST*. Section 3 is devoted to the *MCDM* problem formulated in the spirit of the proposed treatment of *A-IFS* based on the *DST*. Finally, the concluding section summarizes the report.

2 Basic Concept

2.1 Dempster-Shafer Theory

The origins of the Dempster-Shafer theory go back to the work by A.P. Dempster [5,6] who developed a system of upper and lower probabilities. Following this work his student G. Shafer proposed a more thorough explanation of belief functions [15].

Assume A is a subsets of X . A *DS* belief structure has associated with it a mapping m , called basic assignment function, from subsets of X into a unit interval, $m : 2^X \rightarrow [0, 1]$ such that $m(\emptyset) = 0$, $\sum_{A \subseteq X} m(A) = 1$. The subsets of X for which the mapping does not assume a zero value are called focal elements. The null set is never a focal element. In [15], Shafer introduced a number of measures associated with this structure. The measure of belief is a mapping $Bel : 2^X \rightarrow [0, 1]$ such that for any subset B of X

$$Bel(B) = \sum_{\emptyset \neq A \subseteq B} m(A). \tag{1}$$

A second measure introduced by Shafer [15] is a measure of plausibility which is a mapping $Pl : 2^X \rightarrow [0, 1]$ such that for any subset B of X

$$Pl(B) = \sum_{A \cap B \neq \emptyset} m(A). \tag{2}$$

It is easy to see that $Bel(B) \leq Pl(B)$. *DS* provides an explicit measure of ignorance about an event B and its complementary \bar{B} as a length of an interval $[Bel(B), Pl(B)]$ called the belief interval (*BI*).

The core of the evidence theory is the Dempsters rule of combination of evidence from different sources. With two belief structures m_1, m_2 , the Dempster’s rule of combination is defined as

$$m_{12}(A) = \frac{\sum_{B \cap C = A} m_1(B)m_2(C)}{1 - K}, A \neq \emptyset, m_{12}(\emptyset) = 0, \tag{3}$$

where $K = \sum_{B \cap C = \emptyset} m_1(B)m_2(C)$. In [20], Zadeh has underlined that this normalization involves counter-intuitive behaviors in the case of considerable conflict. In order to solve this problem, other combination rules [7,9,13,16,19] have been proposed. Nevertheless, here we shall use only the classical Dempster’s rule (3) as it is more popular in applications [3,11] than other rules.

2.2 Interpretation of Intuitionistic Fuzzy Sets in the Framework of Dempster-Shafer Theory

In [1], Atanassov defined *A-IFS* as follows.

Definition 1. Let $X = \{x_1, x_2, \dots, x_n\}$ be a finite universal set. An intuitionistic fuzzy set A in X is an object of the following form: $A = \{ \langle x_j, \mu_A(x_j), \nu_A(x_j) \rangle \mid x_j \in X \}$, where the functions $\mu_A : X \rightarrow [0, 1], x_j \in X \rightarrow \mu_A(x_j) \in [0, 1]$ and $\nu_A : X \rightarrow [0, 1], x_j \in X \rightarrow \nu_A(x_j) \in [0, 1]$ define the degree of membership and degree of non-membership of the element $x_j \in X$ to the set $A \subseteq X$, respectively, and for every $x_j \in X, 0 \leq \mu_A(x_j) + \nu_A(x_j) \leq 1$. Following to [1], we call $\pi_A(x_j) = 1 - \mu_A(x_j) - \nu_A(x_j)$ the intuitionistic index (or the hesitation degree) of the element x_j in the set A . It is obvious that for every $x_j \in X$ we have $0 \leq \pi_A(x_j) \leq 1$.

Hong and Choi [10] proposed to use interval representation $[\mu_A(x_j), 1 - \nu_A(x_j)]$ of intuitionistic fuzzy set A in X instead of pair $\langle \mu_A(x_j), \nu_A(x_j) \rangle$ in context of *MCDM* problem. The first obvious advantage of such approach is that expression $[\mu_A(x_j), 1 - \nu_A(x_j)]$ represents a regular interval as its right bound always is not smaller than its left bound (this is a consequence of the condition $0 \leq \mu_A(x_j) + \nu_A(x_j) \leq 1$). Obviously, this approach is equivalent to the interval valued fuzzy sets interpretation of *A-IFS*. The second advantage is the possibility to redefine the basics of *A-IFS* in terms of *DST*. Here we show that convenient for the practical applications methods for *MCDM* can be developed using *DST* semantics for *A-IFS*.

Firstly, we show that in the framework of *DST* the triplet $\mu_A(x_j), \nu_A(x_j), \pi_A(x_j)$ represents the basic assignment function. Really, when analyzing any situation in context of *A-IFS*, we implicitly deal with the following three hypotheses: *Yes*: $x_j \in A, No$: $x_j \notin A, (Yes, No)$: both the hypotheses $x_j \in A$ and $x_j \notin A$ can not be rejected (the case of hesitation). In this context, $\mu_A(x_j)$ may be treated as the probability or evidence of $x_j \in A$, i.e., as the focal element of the basic assignment function: $m(Yes) = \mu_A(x_j)$. Similarly, we can assume that $m(No) = \nu_A(x_j)$. Since $\pi_A(x_j)$ is usually treated as the hesitation degree, a natural assumption is $m(Yes, No) = \pi_A(x_j)$. Taking into account that $\mu_A(x_j) + \nu_A(x_j) + \pi_A(x_j) = 1$

we come to the conclusion that triplet $\mu_A(x_j), \nu_A(x_j), \pi_A(x_j)$ represents a correct basic assignment function. According to the *DST* formalism we get $Bel_A(x_j)=m(Yes)=\mu_A(x_j)$ and $Pl_A(x_j)=m(Yes)+m(Yes, No)=\nu_A(x_j) + \pi_A(x_j)=1 - \nu_A(x_j)$.

Therefore, the following definition can be introduced:

Definition 2. Let $X = \{x_1, x_2, \dots, x_n\}$ be a finite universal set and x_j is an object in X presented by the functions $\mu_A(x_j), \nu_A(x_j)$ which represent the degree of membership and degree of non-membership of $x_j \in X$ to the set $A \subseteq X$ such that $\mu_A : X \rightarrow [0, 1], x_j \in X \rightarrow \mu_A(x_j) \in [0, 1]$ and $\nu_A : X \rightarrow [0, 1], x_j \in X \rightarrow \nu_A(x_j) \in [0, 1]$ and for every $x_j \in X, 0 \leq \mu_A(x_j) + \nu_A(x_j) \leq 1$. An intuitionistic fuzzy set A in X is an object having the following form: $A = \{ \langle x_j, BI_A(x_j) \rangle | x_j \in X \}$, where $BI_A(x_j) = [Bel_A(x_j), Pl_A(x_j)]$ is the belief interval, $Bel_A(x_j) = \mu_A(x_j)$ and $Pl_A(x_j) = 1 - \nu_A(x_j)$ are the measures of belief and plausibility that $x_j \in X$ belongs to the set $A \subseteq X$.

At first glance, this definition seems as a simple redefinition of *A-IFS* in terms of Interval Valued Fuzzy Sets, but we show that using the *DSF* semantics it is possible to enhance the performance of *A-IFS* when dealing with *MCDM* problems. Particularly, this approach allows us to use directly the Dempster’s rule of combination to aggregate local criteria presented by *IFVs* and develop a method for *MCDM* without intermediate defuzzification when local criteria and their weights are *IFVs*. As the result we get final alternative’s evaluations in the form of belief interval. Hence, an appropriate method for such intervals comparison is needed. In the following section we describe a new method (based on the *DST*) providing the results of interval comparison in the form of belief interval, i.e., without loss of information caused by intermediate type reductions.

3 MCDM Problem in the Framework of Intuitionistic/DST Approach

To make our consideration more transparent and comparable with the results obtained earlier by the other authors, we shall use here the example analyzed in [12] since only in these paper the *MCDM* problem is considered in the case when not only local criteria, but also their weights are *IFVs*: “Consider an air-condition system selection problem. Suppose there exist three air-condition systems x_1, x_2 and x_3 . Suppose three criteria a_1 (economical), a_2 (function) and a_3 (being operative) are taken into consideration. The degrees μ_{ij} and ν_{ij} for the alternatives with respect to the criteria representing the fuzzy concept “excellence” were presented in [12] as follows:

$$((\mu_{ij}, \nu_{ij}))_{3 \times 3} = \begin{matrix} & \begin{matrix} x_1 & x_2 & x_3 \end{matrix} \\ \begin{matrix} a_1 \\ a_2 \\ a_3 \end{matrix} & \begin{pmatrix} (0.75, 0.10) & (0.80, 0.15) & (0.40, 0.45) \\ (0.60, 0.25) & (0.68, 0.20) & (0.75, 0.05) \\ (0.80, 0.20) & (0.45, 0.50) & (0.60, 0.30) \end{pmatrix} \end{matrix}. \tag{4}$$

The degrees ρ_i of membership and the degrees τ_i of non-membership representing the fuzzy concept “importance” were presented criteria were presented in [12] as follows:

$$((\rho_i, \tau_i))_{1 \times 3} = \begin{pmatrix} a_1 & a_2 & a_3 \\ (0.25, 0.25) & (0.35, 0.40) & (0.30, 0.65) \end{pmatrix}. \quad (5)$$

To get the final alternative’s evaluations $FAE(x_i)$ on the base of data from the structures (14),(15) we use the *DST* interpretation of *A-IFS*. For each pair x_i, a_j there are two sources of information concerned with x_i goodness: the degree of the local criterion satisfaction and the weight (importance) of this criterion. Let us consider the first of them. As in subsection 2.2, in this case we deal with three hypotheses: *Yes*: the alternative x_i is good as it satisfies the local criterion a_j ; *No*: an alternative x_i is rather bad (not good) as it does not satisfies a_j ; (*Yes, No*): the compound hypothesis (we hesitate over a choice of *Yes* or *No*). The degree of local criterion satisfaction can be treated as the first source of evidence for estimation of x_i goodness. Therefore, for the pair x_i, a_j it can be presented by the basic assignment function as follows: $m_1^{ij}(Yes)$, $m_1^{ij}(No)$, $m_1^{ij}(Yes, No)$, where $m_1^{ij}(Yes) = \mu_{ij}$, $m_1^{ij}(No) = \nu_{ij}$, $m_1^{ij}(Yes, No) = 1 - \mu_{ij} - \nu_{ij} = \pi_{ij}$.

The other source of evidence of x_i goodness is the relative importance (weight) of the local criterion. The reason behind this is that we can qualify x_i as a good one if it satisfies well a local criterion of high importance. On the other hand, the degree (quality) of x_i estimation should be qualified as a low one when x_i satisfies poorly a local criterion of a low importance. So three hypothesis should be considered: *Yes*: the criterion a_j is important; *No*: the criterion a_j is not important; (*Yes, No*): the compound hypothesis (we hesitate over a choice of *Yes* or *No*). Then for the local criterion a_j the basis assignment function corresponding to its importance can be presented as follows: $m_2^j(Yes) = \rho_j$, $m_2^j(No) = \tau_j$, $m_2^j(Yes, No) = 1 - \rho_j - \tau_j$. To obtain the combined basic assignment function m_{com} based on these sources of evidence presented by particular assignment functions $m_1^{ij}(Yes)$, $m_1^{ij}(No)$, $m_1^{ij}(Yes, No)$ and $m_2^j(Yes)$, $m_2^j(No)$, $m_2^j(Yes, No)$, we have used the Dempster’s combination rule:

$$m_{com}^{ij}(A) = \frac{\sum_{B \cap C = A} m_1^{ij}(B)m_2^j(C)}{1 - K}, \quad (6)$$

where $K = \sum_{B \cap C = \emptyset} m_1^{ij}(B)m_2^j(C)$, $A, B, C \in \{Yes, No, (Yes, No)\}$. As the result, for each pair x_i, a_j the basic assignment function $m_{com}^{ij}(Yes)$, $m_{com}^{ij}(No)$, $m_{com}^{ij}(Yes, No)$ can be calculated.

Consequently, in the spirit of *DST*, the local criteria a_j can be treated as the particular sources of information (evidence) for the generalized estimation of x_i goodness. There are three local criteria in our example. Hence, each alternative x_i can be presented by the structure M_i as follows:

$$M_i = \begin{pmatrix} m_{com}^{i1}(Yes) & m_{com}^{i1}(No) & m_{com}^{i1}(Yes, No) \\ m_{com}^{i2}(Yes) & m_{com}^{i2}(No) & m_{com}^{i2}(Yes, No) \\ m_{com}^{i3}(Yes) & m_{com}^{i3}(No) & m_{com}^{i3}(Yes, No) \end{pmatrix}. \quad (7)$$

The final basic assignment function based on the particular evidences presented by the local criteria combined with their importances can be obtained using Dempster's rule. For a *MCDM* problem with more than two local criteria, we can first obtain the combination of focal elements of two assignment functions using the Dempster's rule and combine the obtained result with the third assignment function and so on. It is easy to show that in our case of three local criteria, this process leads to following expression:

$$m_{com}^i(A) = \frac{\sum_{B \cap C \cap D = A} m_{com}^{i1}(B) m_{com}^{i2}(C) m_{com}^{i3}(D)}{1 - K}, \quad (8)$$

where $K = \sum_{B \cap C \cap D = \emptyset} m_{com}^{i1}(B) m_{com}^{i2}(C) m_{com}^{i3}(D)$,

$A, B, C, D \in Yes, No, (Yes, No)$.

From this expression for each alternative x_i we obtain the final basic assignment function $m_{com}^i(Yes)$, $m_{com}^i(No)$, $m_{com}^i(Yes, No)$ and the bounds of believe interval $Bel(x_i) = m_{com}^i(Yes)$, $Pl(x_i) = m_{com}^i(Yes) + m_{com}^i(Yes, No)$. If it is needed, this result may be represented in terms of *A-IFS* theory since according to the Definition 2: $\mu^i = m_{com}^i(Yes)$, $\nu^i = m_{com}^i(No)$ and $\pi^i = m_{com}^i(Yes, No)$.

The final alternative's evaluation $FAE(x_i)$ can be presented both by the final basic assignment function m_{com}^i and by the belief interval $[Bel(x_i), Pl(x_i)]$. On the other hand, the last presentation seems to be more convenient as it allows us to compare the final alternative's evaluations $FAE(x_i) = [Bel(x_i), Pl(x_i)]$ using the method presented in [14].

The local criteria aggregation on the base of Dempster's combination rule is only one of the methods we can use for *MCDM* in the framework of *A-IFS/DST* approach. Therefore we will denote the corresponding final alternative's evaluation as $FAE_{com}(x_i)$.

Using above approach, the following results for the considered example (4),(5) have been obtained:

$$FAE_{com}(x_1) = [Bel(x_1), Pl(x_1)] = [0.9257, 0.9257],$$

$$FAE_{com}(x_2) = [Bel(x_2), Pl(x_2)] = [0.8167, 0.8168],$$

$$FAE_{com}(x_3) = [Bel(x_3), Pl(x_3)] = [0.7377, 0.7379].$$

Obviously, to select the best alternative, their final evaluations $FAE_{com}(x_i)$ presented by corresponding intervals should be compared. To obtain the final ranking on the set of comparing alternatives, the real valued criteria introduced in [14] (strong, weak and mixed preferences) could be used, but in the considered example we get the results of such comparison in the form of degenerated belief intervals:

$$BI_{comb}(x_1 > x_2) = 1, BI_{comb}(x_2 > x_3) = 1, BI_{comb}(x_1 > x_3) = 1,$$

$$BI_{comb}(x_1 < x_2) = 0, BI_{comb}(x_2 < x_3) = 0, BI_{comb}(x_1 < x_3) = 0,$$

$$BI_{comb}(x_1 = x_2) = 0, BI_{comb}(x_2 = x_3) = 0, BI_{comb}(x_1 = x_3) = 0.$$

Since in our example only non interval results (0 or 1) of $FAE_{com}(x_i)$ comparison have been obtained, it is easy to see that the final alternative's ranking is $x_3 \prec x_2 \prec x_1$. It is worthy to note that using the same example, the substantially different result $x_2 \prec x_3 \prec x_1$ has been obtained in [12].

We can explain such divergence of the results only by the absence of any intermediate type reduction in our method for $MCDM$ in $A-IFS$ setting that makes it possible to avoid the loss of important information.

4 Conclusion

It is shown that the DST may serve as a good methodological base for interpretation of $A-IFS$. The use of the DSF semantics makes it possible to enhance the performance of $A-IFS$ when dealing with the $MCDM$ problems. Particularly, when solving $MCDM$ problems, proposed approach allows us to use the Dempster's rule of combination directly to aggregate the local criteria presented by $IFVs$ when their weights are $IFVs$ too, without intermediate defuzzification.

References

1. Atanassov, K.T.: Intuitionistic fuzzy sets. *Fuzzy Sets and Systems* 20, 87–96 (1986)
2. Atanassov, K.T.: New operations defined over the intuitionistic fuzzy sets. *Fuzzy Sets and Systems* 61, 137–142 (1994)
3. Beynon, M., Curry, B., Morgan, P.: The Dempster-Shafer theory of evidence: an alternative approach to multicriteria decision modeling. *Omega* 28, 37–50 (2000)
4. Chen, S.M., Tan, J.M.: Handling multicriteria fuzzy decision-making problems based on vague set theory. *Fuzzy Sets and Systems* 67, 163–172 (1994)
5. Dempster, A.P.: Upper and lower probabilities induced by a multi-valued mapping. *Ann. Math. Stat.* 38, 325–339 (1967)
6. Dempster, A.P.: A generalization of Bayesian inference (with discussion). *J. Roy. Stat. Soc., Series B* 30, 208–247 (1968)
7. Denoeux, T.: Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence. *Artificial Intelligence* 172, 234–264 (2008)
8. Deschrijver, G., Kerre, E.E.: On the position of intuitionistic fuzzy set theory in the framework of theories modelling imprecision. *Information Sciences* 177, 1860–1866 (2007)
9. Dubois, D., Prade, H.: Representation and combination of uncertainty with belief functions and possibility measures. *Computational Intelligence* 4, 244–264 (1998)
10. Hong, D.H., Choi, C.-H.: Multicriteria fuzzy decision-making problems based on vague set theory. *Fuzzy Sets and Systems* 114, 103–113 (2000)
11. Hua, Z., Gong, B., Xu, X.: A DS-AHP approach for multi-attribute decision making problem with incomplete information. *Expert Systems with Applications* 34, 2221–2227 (2008)

12. Li, D.-F.: Multiattribute decision making models and methods using intuitionistic fuzzy sets. *Journal of Computer and System Sciences* 70, 73–85 (2005)
13. Murphy, C.K.: Combining belief functions when evidence conflicts. *Decision Support Systems* 29, 1–9 (2000)
14. Sevastjanov, P.: Numerical methods for interval and fuzzy number comparison based on the probabilistic approach and Dempster-Shafer theory. *Information Sciences* 177 (2007)
15. Shafer, G.: *A mathematical theory of evidence*. Princeton University Press, Princeton (1976)
16. Smets, P.: The combination of evidence in the transferable belief model. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12, 447–458 (1990)
17. Xu, Z.: Intuitionistic fuzzy aggregation operators. *IEEE Transactions on Fuzzy Systems* 15, 1179–1187 (2007)
18. Xu, Z., Yager, R.R.: Dynamic intuitionistic fuzzy multi-attribute decision making. *International Journal of Approximate Reasoning* 48, 246–262 (2008)
19. Yager, R.R.: On the Dempster-Shafer framework and new combination rules. *Information Sciences* 41, 93–138 (1987)
20. Zadeh, L.: A simple view of the Dempster-Shafer theory of evidence and its application for the rule of combination. *AI Magazine* 7, 85–90 (1986)

Evolutionary Learning for Neuro-fuzzy Ensembles with Generalized Parametric Triangular Norms

Marcin Gabryel^{1,2}, Marcin Korytkowski^{1,2}, Agata Pokropinska⁴,
Rafał Scherer^{1,5}, and Stanisław Drozd³

¹ Department of Computer Engineering, Częstochowa University of Technology
Al. Armii Krajowej 36, 42-200 Częstochowa, Poland
{marcin.gabryel,marcin.korytkowski,rafal.scherer}@kik.pcz.pl

<http://kik.pcz.pl>

² The Professor Kotarbinski Olsztyn Academy of Computer Science and Management
ul. Artyleryjska 3c, 10-165 Olsztyn, Poland
<http://www.owsiiz.edu.pl>

³ University of Warmia and Mazury in Olsztyn
The Faculty of Mathematics and Computer Sciences
ul. Zolnierska 14, 10-561 Olsztyn, Poland
<http://wmii.uwm.edu.pl>

⁴ Institute of Mathematics and Computer Science
Jan Długosz University
al. Armii Krajowej 13/15, 42-200 Częstochowa, Poland
<http://imi.ajd.czest.pl/>

⁵ Academy of Management (SWSPiZ), Institute of Information Technology
ul. Sienkiewicza 9, 90-113 Łódź, Poland
<http://www.swspiz.pl/>

Abstract. In this paper we present a method for designing neuro-fuzzy systems with Mamdani-type inference and parametric t-norm connecting rule antecedents. Hamacher product was used as t-norm. The neuro-fuzzy systems are used to create an ensemble of classifiers. After obtaining the ensemble by bagging, every neuro-fuzzy system has its t-norm parameters fine-tuned. Thanks to this the accuracy is improved and the number of parameters can be reduced. The proposed method is tested on a well known benchmark.

1 Introduction

Ensemble methods in classification are very popular as they allow to easily achieve very high accuracy at the price of higher computational complexity and worse interpretability. One of the most popular ensemble methods are Bagging and Boosting. In this paper we present a new method for designing an ensemble of Mamdani-type neuro-fuzzy systems created by the Bagging algorithm. Mamdani-type neuro-fuzzy systems are the most popular neuro-fuzzy systems because of the use of the simplest method of fuzzy inference. Presented approach

replaces ordinary triangular norms with parametric ones. Thanks to this we can better fit the systems to the learning data. This improvement is connected with genetic selection of the best ensemble members.

The paper is divided into three main sections. In the next section we will give a short description of Mamdani-type neuro-fuzzy systems and the Bagging algorithm to design the neuro-fuzzy ensemble. Next, for better performance we replace fuzzy t-norms with parametric t-norm in the neuro-fuzzy systems and we reduce the number of classifiers by genetic selecting the best performers. In the last section the result of simulations are shown.

2 Ensemble of Neuro-fuzzy Systems

2.1 Description of Fuzzy Systems

In this paper, we consider a multi input, single output neuro fuzzy system mapping $\mathbf{X} \rightarrow \mathbf{Y}$ where $\mathbf{X} \subset \mathbf{R}^n$ and $\mathbf{Y} \subset \mathbf{R}$. The fuzzy rule base consists of a collection of N fuzzy IF-THEN rules in the form:

$$R^{(r)} : \text{IF } x_i \text{ is } A_1^r \text{ AND } \dots \text{ AND } x_n \text{ is } A_n^r \text{ THEN } y \text{ is } B^r, \tag{1}$$

where $\mathbf{x} = [x_1, \dots, x_n]$ are input variables, n - number of inputs, y - output value, fuzzy sets $A_1^r, A_2^r, \dots, A_n^r$ and B^r are characterized by membership functions $\mu_{A_i^r}(x_i)$ and $\mu_{B^r}(y)$, respectively, $r = 1, \dots, N, i = 1, \dots, n$. This system is based on the Mamdani-type reasoning, where antecedents and consequences in the individual rules are connected by the product t-norm. We use the most common singleton fuzzifier for mapping crisp values of input variables into fuzzy sets [15]. The defuzzification process is made by the following formula

$$\bar{y} = \frac{\sum_{r=1}^N \bar{y}^r \cdot \mu_{\bar{B}^r}(\bar{y}^r)}{\sum_{r=1}^N \mu_{\bar{B}^r}(\bar{y}^r)}, \tag{2}$$

where membership functions fuzzy sets $\bar{B}^r, r = 1, \dots, N$, are defined by

$$\mu_{\bar{B}^r}(\bar{y}^r) = T \left\{ \prod_{i=1}^n (\mu_{A_i^r}(x_i)), \mu_{B^r}(\bar{y}^r) \right\}. \tag{3}$$

Using the fact that

$$\mu_{B^r}(\bar{y}^r) = 1 \tag{4}$$

and using one of the properties of t-norm

$$T(1, a) = a \tag{5}$$

we obtain

$$\mu_{\bar{B}^r}(\bar{y}^r) = \prod_{i=1}^n (\mu_{A_i^r}(x_i)). \tag{6}$$

We choose as membership functions $\mu_{A_i^r}(x_i)$ the Gaussian functions

$$\mu_{A_i^r}(x_i) = \exp \left[- \left(\frac{x_i - \bar{x}_i^r}{\sigma_i^r} \right)^2 \right]. \quad (7)$$

Combining formulas (2)-(7) it is easily seen that the described system takes the form

$$\bar{y} = \frac{\sum_{r=1}^N \bar{y}^r \cdot \prod_{i=1}^N (\mu_{A_i^r}(x_i))}{\prod_{i=1}^N (\mu_{A_i^r}(x_i))}. \quad (8)$$

This system has been trained using the backpropagation method [15]. In the next subsection we will shortly describe an evolutionary algorithm to train the system described by (8) and we design the ensemble of classifiers.

2.2 Designing Ensemble of Classifiers by Bagging Algorithm

The classifier ensemble is designed by combining the methods of evolutionary learning system parameters and the Bagging algorithm. A similar algorithms has been presented previously in papers [7][6]. As t-norm connecting the antecedents of the rules in system (8) we chose Hamacher product T_{H_0} [10]

$$T_{H_0}(a, b) = \begin{cases} 0 & \text{if } a = b = 0 \\ \frac{ab}{a+b-ab} & \text{otherwise} \end{cases} \quad (9)$$

Bagging is a procedure for combining classifiers generated using the same training set. Bagging (bootstrap aggregating) [9][11] produces replicates of the training set \mathbf{z} and trains a classifier D_k on each replicate S_k . Each classifier is applied to a test pattern \mathbf{x} which is classified on a majority vote basis, ties being resolved arbitrarily. We have a set of labels $\Omega = \{\omega_1, \omega_2, \dots, \omega_C\}$, where C is the number of possible classes, labeled ω_i , $i = 1, \dots, C$. We consider the ensemble of classifiers $\mathbf{D} = [D_1, \dots, D_J]$, where there are J base classifiers D_k , $k = 1, \dots, J$. We assume that the output of classifier D_k is $\mathbf{d}_k(\mathbf{x}) = [d_{k,1}(\mathbf{x}), \dots, d_{k,C}(\mathbf{x})]^T \in \{0, 1\}^C$, where $d_{k,j} = 1$ if D_k determines that \mathbf{x} belong to class ω_j , and $d_{k,j} = 0$ otherwise. The majority vote will result in an ensemble decision for class ω_k if

$$\sum_{i=1}^J d_{i,k}(\mathbf{x}) = \max_{j=1}^C \sum_{i=1}^J d_{i,j}(\mathbf{x}) \quad (10)$$

The Bagging algorithm consists of the following steps

1. Initialize the parameters
 - the ensemble $\mathbf{D} = \emptyset$
 - the number of classifiers to train J
2. For $k = 1, \dots, J$ repeat points 3-5

3. Take sample S_k from \mathbf{Z}
4. Build a classifier D_k using S_k as the training set by evolutionary strategy (μ, λ) [6]
5. Add the classifier to the current ensemble $\mathbf{D} = \mathbf{D} \cup D_k$
6. Return \mathbf{D} as algorithm outcome
7. Run \mathbf{x} on the input D_1, \dots, D_J
8. The vector \mathbf{x} is a member of class ω_k , if condition (10) is fulfilled.

3 Improvement of the Performance and Reduction of Neuro-fuzzy Ensembles

3.1 Improvement the Performance of Neuro-fuzzy Systems

The use of parametric t-norms allows to improve system performance. In our method, after complete Bagging algorithm, we replace in D_k each Hamacher product t-norm by Hamacher parametric t-norm [10]

$$T_{H_p}(a, b) = \begin{cases} 0 & \text{if } p = a = b = 0 \\ \frac{ab}{p+(1-p)(a+b-ab)} & \text{otherwise} \end{cases} \quad (11)$$

where $p > -1$ is a parameter of t-norm. For $p = 0$ we obtain formula (9). The system (8) can be written as follows

$$\bar{y} = \frac{\sum_{r=1}^N \bar{y}^r \cdot T_{H_{p_r}}^N(\mu_{A_i^r}(x_i))}{T_{H_{p_r}}^N(\mu_{A_i^r}(x_i))} \quad (12)$$

For tuning parameters p_r of each classifier D_k we use evolutionary strategy (μ, λ) [4]. Each parameters p_r are represented in chromosome $\mathbf{X}_j = [p_1, p_2, \dots, p_N]$, where $j = 1, \dots, \mu$, $j = 1, \dots, \lambda$. Fitness function calculates the percentage of classification effectiveness obtained by the tuned system.

3.2 Reduction of Neuro-fuzzy Ensembles

The algorithm described above is performed on each classifier D_k and most of them improve their effectiveness. This operation allows us to reduce the number of classifiers by choosing the best systems. For this purpose, a classic genetic algorithm with binary encoding will be used. The length of the chromosome is equal to the total number of classifiers J . Each gene corresponds to the presence (encoded as 1) or exclusion (encoded as 0) of the classifier with the ensemble. If the decoded chromosome obtained worse results than the ensemble of classifiers from the previous algorithm, a penalty function would be imposed by resetting value of the fitness function.

4 Simulation Results

The algorithm presented in this paper consists of three phases:

1. learning the individual classifiers D_k by evolutionary strategy (μ, λ) during Bagging algorithm (subsection 2.2),
2. replacing t-norms connecting the antecedents of the rules on the parametric Hamacher t-norm and tuning parameters p_r by evolutionary strategy (μ, λ) (subsection 3.1),
3. reduction of redundant classifiers using classical genetic algorithm (subsection 3.2).

For the individual steps we applied the following parameters for evolutionary algorithms:

1. $\mu = 30$, $\lambda = 200$, number of generations is 100, number of classifiers $J = 50$,
2. $\mu = 10$, $\lambda = 70$, number of generations is 100,
3. chromosome length is 50, number of generations is 200, population size is 100, $p_m = 0.1$ (probability of mutation), $p_c = 0.7$ (probability of crossover), uniform crossover.

We consider the Glass Identification problem [16], which contains 214 instances and each instance is described by nine attributes (RI: refractive index, Na: sodium, Mg: magnesium, Al: aluminium, Si: silicon, K: potassium, Ca: calcium, Ba: barium, Fe: iron). All attributes are continuous. There are two classes: the window glass and the non-window glass. In our experiments, the data set is divided into a learning sequence (150 sets) and a testing sequence (64 sets). All individual classifiers in the ensemble have 4 rules. Detailed accuracy for all subsystems for three phases of our algorithm are shown in Table 1. After second phase we obtained 97% classification accuracy for the training set and 96% for the testing sets, and the third phase reduces the system to 24 classifiers with the same performance.

Table 1. The experimental results for learning neuro-fuzzy ensembles for each stage of the proposed method

Phase of algorithm	Training	Testing	Number of classifiers
1	90%	88%	50
2	97%	96%	50
3	97%	96%	27

5 Final Remarks

We have presented a three-phase algorithm for designing ensemble of neuro-fuzzy systems. Using the parametric t-norms in the neuro-fuzzy systems can significantly improve the classification results. The presented approach allowed to reduce the number of individual systems and to improve accuracy.

Acknowledgments

This work was partly supported by the Polish Ministry of Science and Higher Education (Habilitation Project 2007-2010 Nr N N516 1155 33, Polish-Singapore Research Project 2008-2010 and Research Project 2008-2011) and the Foundation for Polish Science – TEAM project 2010-2014.

References

1. Breiman, L.: Bagging predictors. *Machine Learning* 26(2), 123–140 (1996)
2. Cordon, O., Herrera, F., Hoffman, F., Magdalena, L.: *Genetic Fuzzy System, Evolutionary Tunning and Learning of Fuzzy Knowledge Bases*. World Scientific, Singapore (2000)
3. Cordon, O., Gomide, F., Herrera, F., Hoffmann, F., Magdalena, L.: Ten years of genetic fuzzy systems: current framework and new trends. *Fuzzy sets and systems* 141, 5–31 (2004)
4. Eiben, A.E., Smith, J.E.: *Introduction to Evolutionary Computing*. Springer, Heidelberg (2003)
5. Gabryel, M., Cpalka, K., Rutkowski, L.: Evolutionary strategies for learning of neuro-fuzzy systems. In: *I Workshop on Genetic Fuzzy Systems*, Granada, pp. 119–123 (2005)
6. Gabryel, M., Rutkowski, L.: Evolutionary Learning of Mamdani-type Neuro-Fuzzy Systems. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) *ICAISC 2006. LNCS (LNAI)*, vol. 4029, pp. 354–359. Springer, Heidelberg (2006)
7. Gabryel, M., Rutkowski, L.: Evolutionary Methods for Designing Neuro-fuzzy Modular Systems Combined by Bagging Algorithm. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008. LNCS (LNAI)*, vol. 5097, pp. 398–404. Springer, Heidelberg (2008)
8. Korytkowski, M., Gabryel, M., Rutkowski, L., Drozda, S.: Evolutionary Methods to Create Interpretable Modular System. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008. LNCS (LNAI)*, vol. 5097, pp. 405–413. Springer, Heidelberg (2008)
9. Kuncheva, L.I.: *Fuzzy Classifier Design*. Physica Verlag, Heidelberg (2000)
10. Klement, E.P., Mesiar, R., Pap, E.: Triangular norms. Position paper II: general constructions and parametrized families. *Fuzzy Sets and Systems* 145, 411–438 (2004)
11. Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs*, 3rd edn. Springer, Heidelberg (1996)
12. Rutkowska, D., Nowicki, R.: Implication-Based Neuro-Fuzzy Architectures. *International Journal of Applied Mathematics and Computer Science* 10(4) (2000)
13. Rutkowska, D.: *Neuro Fuzzy Architectures and Hybrid Learning*. Springer, Heidelberg (2002)
14. Rutkowski, L.: *Computational Intelligence, Methods and Techniques*. Springer, Heidelberg (2008)
15. Rutkowski, L.: *Flexible Neuro Fuzzy Systems*. Kluwer Academic Publishers, Dordrecht (2004)
16. Mertz, C.J., Murphy, P.M.: UCI respository of machine learning databases, <http://www.ics.uci.edu/pub/machine-learning-databases>

Fuzzy Spatial Analysis Techniques for Mathematical Expression Recognition

Ray Genoe and Tahar Kechadi

University College Dublin, Belfield, Dublin 4, Ireland
{ray.genoe,tahar.kechadi}@ucd.ie

Abstract. One of the major issues encountered in the recognition of handwritten mathematical expressions, is the non-linear nature of the symbols. The spatial positioning of each symbol with respect to the others implies mathematical relationships, but the ambiguous nature of handwriting can often lead to the misrecognition of these relationships. This paper describes the results of comparative testing of spatial analysis techniques that determine these relationships, in an effort to discover the most appropriate technique to be adopted during the development of an efficient recognition system.

1 Introduction

Research into the area of machine recognition of handwritten input, has been driven by the recent and rapid development of pen-based interfaces and the desire to combine the natural advantages of handwritten input with the processing power and storage of modern computers. The main issues that arise when considering mathematical expression recognition, are the determination of individual symbol identities and the overall structure of the input. The multi-tiered nature of special symbols such as fractions and scripts, and the embedded mathematical relationship that is implied by the positioning of one symbol to another, can often lead to ambiguities when recognising handwritten input.

Determining the relationship between two mathematical symbols involves investigating the horizontal and vertical positions of the first symbol in relation to the second. A comparison of the size of each symbol can also be used to aid the discovery of implicit relationships. These factors must be considered with respect to the type of symbols. Special consideration is usually given to ascenders and descenders due to the obvious differences in placement and size, with respect to other symbols [1,2,3,4,5]. Furthermore, it is important to know that some symbols are not suitable for certain relationships, while others suit a variety of relationships. For example, a minus symbol can be the binary operator that defines a fraction, a unary minus operation or a binary minus operation, whereas naturally, the relationship between two digits can not be one of these operations.

The authors in [5], described a soft-decision approach to resolve ambiguity by generating alternative solutions to the expression. The output string for the

expression must be syntactically verified by each alternative generated, otherwise it is considered invalid. Afterwards, if there is more than one alternative, the user must select the correct solution. In [6], fuzzy logic is introduced to cope with the ambiguity of handwritten input when recognising symbols and relationships. Subsequently, in the structural analysis phase, multiple parses are pursued if ambiguities arise and the most likely expression tree is selected as the result.

The order of symbols entered is also a prevalent issue when considering the relationships between symbols and a conventionally defined “order-of-entry” method is sometimes imposed on a user when they are entering symbols [7,8]. For example when a users wishes to enter a fraction, the authors in [8], required that the user initially enter the numerator, then the dividing line and finally the denominator. This was due to the temporal nature of the input required for their HMM. Other methods check a relationship in reverse order when no relationship is found in the conventional order of symbol entry, to ensure that regressive entries are managed appropriately [1,2,3]. The “order-of-entry” issue particularly affects the development of online recognition systems, as these systems usually require an expression to be developed incrementally, each time the user finishes a stroke and lifts the pen. Most offline recognition systems will sort the symbols when the user has completed entering the expression, so the “order-of-entry” of symbols is not usually an issue [4,5,9].

The system presented here, adopts an online approach to mathematical expression recognition. It uses fuzzy logic techniques to resolve the ambiguity of handwritten input and provides alternative solutions to the user when available; the instant the pen has been lifted after drawing a stroke. These techniques form part of the spatial analysis phase of expression recognition, where the size and positioning of the symbols are investigated to determine the relationships that exist between them.

2 Spatial Analysis Techniques

In this section we shall discuss 3 techniques that have been developed to resolve ambiguity encountered during relationship discovery. The symbol recognition and structural development phases, described in [10] and [3] respectively, are used by the system described in the spatial analysis discussion that follows. Fuzzy Logic is used extensively throughout the system. Previous work by the authors [1,2,3,6,10,11], combined with the results of the extensive testing that can be seen later, have shown that the use of Fuzzy Logic techniques, is appropriate for resolving the ambiguity found in symbol recognition and spatial analysis. The fuzzy function, $g(x)$, consists of 6 thresholds that can be assigned with a range of values that depend on the relationship we are investigating:

$$T_{REL} = \{a_{REL}, b_{REL}, c_{REL}, d_{REL}, e_{REL}, f_{REL}\}$$

These thresholds determine the confidence between 0 and 1 for various spatial verification techniques such as vertical and horizontal confidence. The upper (a) and lower (f) limits of the set \mathbf{T} represent the limits of confidence for the specific

test, while the range $[c, d]$ represents the region of maximum confidence. The limits b and e are used to strengthen/weaken the area of high confidence. For example, when analysing the vertical placement of a superscript relationship, the value e would be close to d , to avoid confusion with concatenation, just as the value b will be closer to c with respect to subscript relationship discovery. The thresholds in \mathbf{T} are dynamically generated in each of the techniques, with the Bounding Box Approach being the only exception.

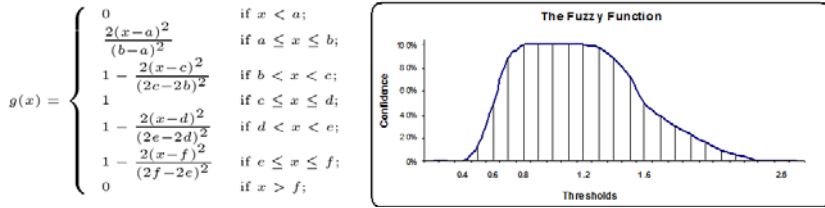


Fig. 1. The fuzzy function $g(x)$, where $T = \{0.4, 0.6, 0.8, 1.2, 1.6, 2.5\}$ and $x \geq 0$

2.1 The Bounding Box Approach

This approach was detailed in [11] and derived its name from the fact that the bounds of the bounding boxes of the symbols were heavily used when determining relationship confidence. In this approach the thresholds of the fuzzy function, used for determining vertical, horizontal and size confidences, were static for all types of symbols and relationships. This was achievable due to the fact that the input for the function was determined by the comparison of the positioning of one bounding box to another, as shown in Fig 2(a). Thus, for example, the vertical input, x , for $g_{superscript}(x) \leq 1$, $g_{subscript}(x) \geq 1$ and $g_{concatenation}(x) \approx 1$ and the static thresholds for $g(x)$ should reflect this where appropriate. Certain symbols such as ascenders, descenders and digits require their bounding boxes to be appropriately adjusted by a factor of 0.5 when comparing the vertical positioning and size of the symbols. Furthermore, the width of certain “vertical-line” symbols, such as the digit 1 or the letter l, are not appropriate measures to consider when comparing the horizontal positioning of symbols. As a result, half of the height of these symbol are used instead. A similar approach was adopted when dealing with the minus symbol. During preliminary testing it was discovered that there were some problems with this approach, especially when investigating implicit relationships. This was partly due to the fact that the factor of adjustment for special symbols (0.5) mentioned previously, did not accurately reflect the features of the symbols drawn. Figure 2(b) shows an example of a concatenation relationship entered by three different users. The values found for vertical input to $g_{concat}(x)$ for each sample were 1.47, 1.15 and 0.68 respectively, and $T_{VERT} = \{0.5, 0.7, 0.9, 1.2, 1.3, 1.5\}$. This resulted in a concatenation relationship being discovered for each example but further analysis of the first and third samples resulted in a slightly higher confidence in a subscript

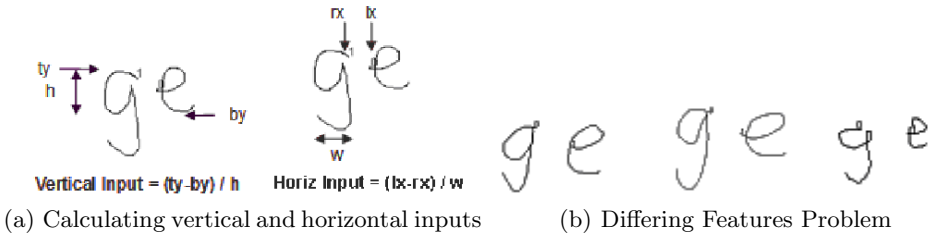


Fig. 2. Bounding Box Approach

and superscript relationship respectively. In addition to this adjustment problem, it was quite difficult to calibrate the thresholds and adjust the function due to the many factors that determine x . This led to the implementation of the Base Line Approach and subsequently the Feature-Based Approach.

2.2 The Base Line Approach

This approach derives its name from the fact that the baseline of the second symbol is used as input for $g_{REL}(x)$ for vertical confidence. The thresholds for $g(x)$ are determined by the size and position of the first symbol, $s1$, and the set of coefficients (C) that have been determined for the specific relationship:

$$C_{REL} = \{a_{REL}, b_{REL}, c_{REL}, d_{REL}, e_{REL}, f_{REL}\}$$

$$T_{REL} = v_{s1}(C_{REL}) = b_{s1} - (h_{s1} * C_{REL})$$

The function v_{s1} is used to calculate the thresholds of the vertical position for the desired relationship (rel). This example calculates vertical confidence, where the baseline (b) and height (h) of the first symbol determine the thresholds that will be used to establish the confidence that the second symbol is positioned correctly. A similar approach is taken when determining the horizontal confidence thresholds.

This method was much easier to implement than the Bounding Box Approach, due to the fact that there was a clear implication between the vertical confidence of an implicit relationship between symbols and the positioning of the second symbol in relation to the first. This resulted in a much easier calibration of the coefficients used when calculating the set T . As a result, initial testing showed significant improvements on the Bounding Box Approach. However, some of the problems previously identified were still evident in this approach due to the differing base lines of descenders as in Figure 2(b). This led to the development of the next approach in an effort to add more information regarding the features of certain symbols.

2.3 The Feature-Based Approach

The symbol recognition phase mentioned in [10,11] and used as part of the overall HMER system, analyses the features of each symbol recognised and stores

this information along with the identity of each symbol. The Feature-Based Approach utilises the feature information in an effort to add more accuracy to relationship determination. It also includes some added complexity, as it not only finds where the bottom of the key feature of a symbol is positioned in relation to another, but also finds where the top of the key feature is positioned. This eliminates the need to rely as heavily on a comparison of the size of the symbols for implicit relationship discovery, as this is implied by calculating the confidence that the upper and lower limits of the key features are positioned correctly. Top and bottom vertical confidence, VC_{TOP} and VC_{BTM} , is calculated using the following formulae, where t_{s1} and b_{s1} are the top and bottom y coordinates of symbol one's key feature and t_{s2} and b_{s2} represent the top and bottom y coordinates of symbol two's key feature.

$$T_{TOP} = v_{s1_{TOP}}(C_{REL_{TOP}}) = t_{s1} - (h_{s1} * C_{REL_{TOP}})$$

$$VC_{TOP} = g_{VERT_{TOP}}(t_{s2})$$

$$T_{BTM} = v_{s1_{BTM}}(C_{REL_{BTM}}) = b_{s1} - (h_{s1} * C_{REL_{BTM}})$$

$$VC_{BTM} = g_{VERT_{BTM}}(b_{s2})$$

This endeavour involved the creation of many more sets of threshold coefficients and significantly increased the complexity of the function that determines vertical confidence. It soon became apparent that the increase in the number of alternative threshold coefficients made it very difficult to calibrate the system. However, it was shown during testing that the added information regarding the features did improve the decision making process while reducing the error rate of relationship discovery.

3 Testing

The spatial analysis techniques mentioned in the previous section were tested on a dataset consisting of 2836 relationships. These relationships were created by 11 different users, who were originally asked to create a dataset of 25 detailed expressions each. All of the symbols that were recognised correctly by the symbol recogniser and that were part of a relationship in each expression, were separated from the original dataset with respect to the relationship, and stored in the test set. An example of this can be seen in Figure 3. The reason this was done was to preserve the natural positioning of the symbols that the users entered when constructing an entire expression, rather than for example, requesting them to enter 10 samples of a scripted relationship, 10 samples of an addition relationship, and so on. The latter method, in the opinion of the authors, may have led to homogeneity in the data samples. For the purpose of consistency, the relationship precedence method of structural analysis and development that was described in 3 was used in each of the spatial methods that were tested. This should not have been an issue as most of the relationships in the dataset consisted of two symbols and were fairly straightforward in their construction. There was however 0.035% error (one out of

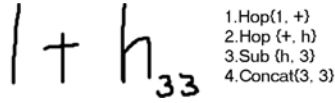


Fig. 3. An original user sample, broken down into 4 relationship samples

Table 1. Analysis of Spatial Relationship Techniques

Method	Correct	Alternatives	
	Solutions	Available	Errors
Bounding Box	96.01%	3.07%	0.92%
Base Line	99.02%	0.67%	0.32%
Feature Based	99.05%	0.85%	0.11%
Hybrid	99.44%	0.49%	0.07%

2836) due to this structural method inappropriately dealing with some unordered symbols in a fraction that was tested for a concatenation relationship. Table 1 describes the results of the testing for each method. The Hybrid Approach is a variation of the base line approach, whereby features are used when available, to improve the recognition rates of implicit relationships. Most of the misrecognition in the Bounding Box Approach can be appropriated to the ambiguity encountered while discovering implicit relationships, particularly when dealing with concatenation. This was partly due to users entering symbols at angles contrary to the horizontal manner of most users and also due to the base line problems discussed earlier, regarding descenders. Many of these relationships were discovered but were not considered by the system to be the best relationship available.

The Base Line Approach vastly improved the decision-making process of the system as it improved the recognition of concatenation relationships by 6.98% and superscript relationships by 1.1%. However, to improve the recognition rates of the former, it was necessary to sacrifice 2.5% of the subscript relationships being presented as the best solution. However, it should be said that 100% of the subscript relationships were discovered.

The Feature-Based Approach improved the relationship discovery process by reducing the errors where the desired relationship was not discovered at all. The primary area of improvement was mainly found when investigating concatenation relationships between descenders and fractions. The added information regarding the top feature of the descenders ensured that the relationship was not misrecognised as a superscript relationship, as was the case in the previous approaches. However, while the concatenation recognition rates were slightly increased, the number of correct relationships discovered for superscript relationships were significantly depleted. This was a disappointing outcome for an approach that was supposed to increase the accuracy of implicit relationships. It is the opinion of the authors, that the difficulties encountered when calibrating the system, as a result of the added complexity in the algorithm and the additional sets of threshold coefficients, was the most likely reason for the insignificant improvement in relation to the Base Line Approach.

The Hybrid Approach was based on the Base Line Approach but also used features, when available, to determine the base line of descenders. This approach resulted in the least error rates and only failed to discover one relationship out of the entire dataset. Furthermore, it returned the highest number of correct relationships for each set that was tested. The most significant area of improvement was in the discovery of subscript relationships, where 100% of the relationships were identified as the optimal solution.

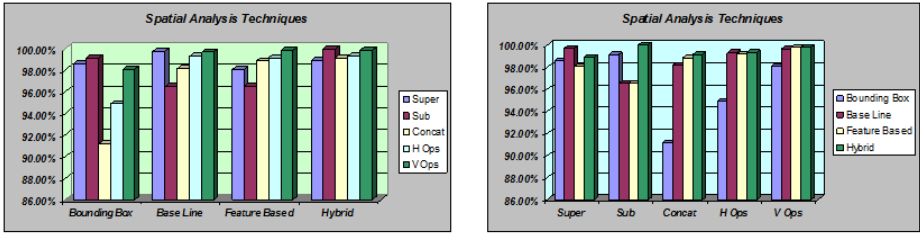


Fig. 4. Results of the spatial analysis techniques
(H Ops and V Ops are explicit horizontal and vertical operations)

4 Conclusion

The experimental results were limited by the current symbol recogniser, as important mathematical operators such as parenthesis and integral symbols are outside its recognition capabilities. Therefore the dataset was limited to implicit and explicit relationships between digits, letters and the plus and minus symbols. However the methods of relationship discovery can easily integrate additional relationships between new symbols that are added to the symbol recogniser's scope. We believe that the tests described in this publication, are a true reflection of the results that could be obtained from a complete dataset of mathematical operations.

The results have shown that the complexity of the spatial analysis techniques can have a direct influence over the performance of the system. The relatively simplistic nature of the Base Line Approach, proved to be just as viable as the complex Feature-Based Approach. Furthermore, the former approach can easily incorporate new relationships, due to the generic nature of the methods that determine positional confidence. The Hybrid Approach is also a suitable solution, due to the fact that it only uses the feature information when available. Not all symbol recognition software will provide such information; another disadvantage of the Feature-Based Approach. In the absence of feature information, the system simply reverts back to the Base Line Approach that boasts recognition rates of 99.05%, only 0.42% less than the Hybrid Approach.

The next step in the development of the spatial analysis phase, is to change or develop the symbol recogniser so that the complete set of mathematical relationships can be identified. The structural development phase will be addressed

with some proposed techniques, and after subsequent testing we hope to have developed a highly efficient and marketable recognition system.

References

1. Genoe, R., Fitzgerald, J., Kechadi, T.: A purely online approach to mathematical expression recognition. In: IWFHR 10, The Tenth International Workshop on Frontiers in Handwriting Recognition, La Baule, France (October 2006)
2. Genoe, R., Fitzgerald, J., Kechadi, T.: An online fuzzy approach to the structural analysis of handwritten mathematical expressions. In: WCCI 2006, The IEEE International Conference on Fuzzy Systems, Vancouver, BC, Canada (July 2006)
3. Genoe, R., Kechadi, T.: On the recognition of online handwritten mathematics using Feature-Based fuzzy rules and relationship precedence. In: WCCI 2008, The IEEE International Conference on Fuzzy Systems, Hong Kong (June 2008)
4. Zanibbi, R., Blostein, D., Cordy, J.R.: Baseline structure analysis of handwritten mathematics notation. In: ICDAR 2001, The Sixth International Conference on Document Analysis and Recognition, Seattle, WA, USA (September 2001)
5. Winkler, H., Fahrner, H., Lang, M.: A soft-decision approach for structural analysis of handwritten mathematical expressions. In: ICASSP 1995: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, Detroit, Michigan, vol. 4, pp. 2459–2462 (May 1995)
6. Fitzgerald, J.A., Geiselbrechtger, F., Kechadi, M.T.: Structural analysis of handwritten mathematical expressions through fuzzy parsing. In: ACST 2006: Proceedings of the Second IASTED International Conference on Advances in Computer Science and Technology, pp. 151–156. ACTA Press (2006)
7. Rhee, T.H., Kim, J.H.: Efficient search strategy in structural analysis for handwritten mathematical expression recognition. *Pattern Recognition Journal* 42(12), 3192–3201 (2009)
8. Kosmala, A., Rigoll, G.: Recognition of On-Line handwritten formulas. In: 6th Int. Workshop on Frontiers in Handwriting Recognition (IWFHR), Taejon, S. Korea (August 1998)
9. Garain, U., Chaudhuri, B.: Recognition of online handwritten mathematical expressions. *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics* 34(6), 2366–2376 (2004)
10. Fitzgerald, J.A., Geiselbrechtger, F., Kechadi, T.: Application of fuzzy logic to online recognition of handwritten symbols. In: IWFHR 2004: Proceedings of the Ninth International Workshop on Frontiers in Handwriting Recognition, Tokyo, Japan, October 2004, pp. 395–400. IEEE Computer Society, Los Alamitos (2004)
11. Fitzgerald, J.A., Geiselbrechtger, F., Kechadi, T.: Feature extraction of handwritten symbols using fuzzy logic. In: AI 2004: Proceedings of the The Seventeenth Canadian Conference on Artificial Intelligence, Ontario, Canada, pp. 493–498. Springer, Heidelberg (2004)

A Modified Pittsburg Approach to Design a Genetic Fuzzy Rule-Based Classifier from Data

Marian B. Gorzalczany and Filip Rudziński

Department of Electrical and Computer Engineering
Kielce University of Technology
Al. 1000-lecia P.P. 7, 25-314 Kielce, Poland
{m.b.gorzalczany,f.rudzinski}@tu.kielce.pl

Abstract. The paper presents a modification of the Pittsburg approach to design a fuzzy classifier from data. Original, non-binary crossover and mutation operators are introduced. No special coding of fuzzy rules and their parameters is required. The application of the proposed technique to design the fuzzy classifier for the well known benchmark data set (*Wisconsin Breast Cancer*) available from the <http://archive.ics.uci.edu/ml> is presented. A comparative analysis with several alternative (fuzzy) rule-based classification techniques has also been carried out.

1 Introduction

Fuzzy rule-based systems belong to the most important areas of applications of the theory of fuzzy sets and fuzzy logic. These applications include - to mention the most prominent ones - fuzzy control (see e.g. [12]), fuzzy classification (see e.g. [8]) and fuzzy identification and modelling (see e.g. [11]). Fuzzy systems, however, are not capable of learning fuzzy rules from available data. In order to diminish this significant drawback, hybrid solutions (belonging to the domain of computational intelligence [1]) have been developed. In particular, various implementations of fuzzy rules in neural-network-like structures - referred to as neuro-fuzzy systems - have been proposed (see e.g. [5], [7], [10]). Most of them, however, operates on predefined, “initial” fuzzy rule bases and uses gradient descent, backpropagation-like learning algorithms leading to local optima of the assumed system-quality indices.

The design process of a fuzzy rule-based system from a given input-output data set can be presented as a structure- and parameter-optimization problem. Therefore, only the application of global search and optimization methods in complex spaces can lead to optimal or sub-optimal solutions to the considered problem. Ideal methodologies to obtain such solutions are offered by evolutionary computations and, in particular, by genetic algorithms. There are three general approaches to design (fuzzy) rule-based system by means of evolutionary techniques (see e.g. [3]): Michigan, Pittsburg and iterative rule learning. The first and the last approaches encode one fuzzy rule of the rule base in a single chromosome. The Pittsburg approach encodes the entire fuzzy rule base in a single

chromosome and thus, the population consists of competing rule bases. One of the main concerns regarding the latter case is that of representation (see e.g. [4], [9]). In particular, binary representation may lead to a premature convergence of the system (see discussion in [3]).

This paper presents a modification of the Pittsburg approach to design a genetic fuzzy rule-based classifier from data. The essential issue of the problem representation is addressed simply by avoiding any special coding of fuzzy rules and their parameters; a direct representation and processing of fuzzy rules can be made. Original, non-binary crossover and mutation operators are introduced to secure the proper operation of the system. Also, a two-stage technique for the pruning of the obtained fuzzy rule bases is introduced. First, the problem of the fuzzy rule-based classifier design is formulated. Then, the details of the proposed genetic approach are presented. In turn, the application of the proposed technique to the well known benchmark data set (*Wisconsin Breast Cancer*) available from the <http://archive.ics.uci.edu/ml> is presented. Finally, a comparative analysis - from the point of view of accuracy versus interpretability criterion - with several alternative methods of designing (fuzzy) rule-based classifiers is carried out.

2 Fuzzy Classification Rules and Fuzzy Inference [5]

The considered classifier is a system with n inputs (attributes) x_1, x_2, \dots, x_n and an output, which has the form of a possibility distribution over the set $Y = \{y_1, y_2, \dots, y_c\}$ of class labels. Each input attribute x_i ($x_i \in X_i$), $i = 1, 2, \dots, n$, is described by numerical values. The “values” of symbolic attributes are encoded using integer numbers. The classifier is designed from the learning data in the form of K input-output records:

$$L_1 = \{x'_k, y'_k\}_{k=1}^K, \quad (1)$$

where $x'_k = (x'_{1k}, x'_{2k}, \dots, x'_{nk}) \in \mathbf{X} = X_1 \times X_2 \times \dots \times X_n$ (\times stands for Cartesian product of ordinary sets) is the set of input numerical attributes, and y'_k is the corresponding class label ($y'_k \in Y$) for data sample no. k . Expression (1) can be rewritten in a more general form:

$$L = \{x'_k, B'_k\}_{k=1}^K, \quad (2)$$

where x'_k is as in (1) and B'_k is a fuzzy set representing a possibility distribution defined over the set Y of class labels ($B'_k \in F(Y)$, where $F(Y)$ denotes the family of all fuzzy sets defined in the universe Y). The possibility distribution $F(Y)$ assigns to each class $y_j \in Y$, a number from the interval $[0,1]$, which indicates to what extent the object described by x'_k belongs to that class. In particular, when we deal with a nonfuzzy possibility distribution over Y , the fuzzy set B'_k is

reduced to fuzzy singleton $B'_{k(singl.)}$, which indicates one class, say y'_k , with the degree of belonging equal to 1, and the remaining classes with the degree equal to 0. This case is represented by expression (1).

Fuzzy classification rules that will be synthesized from the learning data \mathbf{L} (2) by the proposed later in the paper genetic technique have the following form (for the case when all n input attributes are involved):

$$\text{IF } (x_1 \text{ is } A_{1r}) \text{ AND } \dots \text{ AND } (x_n \text{ is } A_{nr}) \text{ THEN (possibility distr. } B_r), \quad (3)$$

where $A_{ir} \in F(X_i)$, $i = 1, 2, \dots, n$ is one of the S-, M-, or L-type fuzzy sets (see below), and $B_r \in F(Y)$ is the possibility distribution; all in the r -th fuzzy rule, $r = 1, 2, \dots, R$. If fuzzy classification rules are to be synthesized from the learning data \mathbf{L}_1 (1) - or, equivalently, from the data \mathbf{L} (2) but with singleton-type fuzzy sets B'_k - then the consequent fuzzy sets B_r in (3) will be replaced by appropriate fuzzy singleton possibility distributions $B_{r(singl.)}$.

As mentioned earlier, the input attributes are described by three types of fuzzy sets corresponding to verbal terms “*Small*” (S-type), “*Medium*” (M-type) and “*Large*” (L-type). Their membership functions have the following forms: $\mu_{M_i}(x_i) = \exp[-0.5(x_i - c_{M_i})^2 / \sigma_{M_i}^2]$, $\mu_{S_i}(x_i) = \exp[-0.5(x_i - c_{S_i})^2 / \sigma_{S_i}^2]$ only for $x_i \geq c_{S_i}$ and 1 elsewhere, and, analogously, $\mu_{L_i}(x_i) = \exp[-0.5(x_i - c_{L_i})^2 / \sigma_{L_i}^2]$ for $x_i \leq c_{L_i}$ and 1 elsewhere (see Fig. 2 later in the paper); $\sigma_{S_i} > 0$, $\sigma_{M_i} > 0$, $\sigma_{L_i} > 0$, $i = 1, 2, \dots, n$. In general, one S-type, one L-type and several M-type fuzzy sets can be considered for a given attribute x_i . In this paper, only one M-type fuzzy set for each attribute x_i is assigned providing - in the considered applications - both, high accuracy and high interpretability of the fuzzy systems.

In the course of the operation of the genetic algorithm, an evaluation of particular individuals (fuzzy rule bases) takes place in each generation. Therefore, a fuzzy-set-theory representation of fuzzy rule base (3) as well as fuzzy inference mechanism have to be employed. It can be shown (see e.g. our monograph [5]) that using the Mamdani fuzzy implication (with min-type t -norm), representing AND-connectives in the antecedent part of the rules also by min-type t -norms, and selecting max-type s -norm operator to combine fuzzy relations representing particular rules, one can obtain - for the input numerical data $\mathbf{x}^0 = (x_1^0, x_2^0, \dots, x_n^0)$ - a fuzzy response (possibility distribution B^0 represented by its membership function $\mu_{B^0}(y_j)$, $j = 1, 2, \dots, c$) of the fuzzy classifier (3):

$$\mu_{B^0}(y_j) = \max_{r=1,2,\dots,R} \mu_{B_r^0}(y_j) = \max_{r=1,2,\dots,R} \min[\alpha_r, \mu_{B_r}(y_j)], \quad (4)$$

where $\alpha_r = \min(\alpha_{1r}, \alpha_{2r}, \dots, \alpha_{nr}) = \min[\mu_{A_{1r}}(x_1^0), \mu_{A_{2r}}(x_2^0), \dots, \mu_{A_{nr}}(x_n^0)]$. α_r is the activation degree of the r -th fuzzy rule by the input numerical data \mathbf{x}^0 .

3 Genetic Learning of Fuzzy Classification Rules from Data

Two separate entities, typical for knowledge-based systems, are being processed in the course of the operation of the genetic algorithm in the considered approach: a rule base and a data base. As mentioned earlier, no special coding of fuzzy rules and their parameters has been introduced. Therefore, the rule base simply contains the information about the present structure of particular rules indicating which antecedents are presently in use (they can be switched on and off). The data base contains the parameters of membership functions of particular fuzzy antecedents, that is, $c_{S_{ir}}, \sigma_{S_{ir}}, c_{M_{ir}}, \sigma_{M_{ir}}, c_{L_{ir}}, \sigma_{L_{ir}}$ where i denotes the number of the antecedent, and r - the number of the rule. A fundamental role is played by the proposed, non-binary crossover and mutation operators.

The crossover operator, that processes two individuals (two rule bases), operates in two stages:

- C1 (labelled as *RuleBaseExchange*) - repeats a random number of times two following activities: C1.1 - randomly selects one fuzzy rule in each of two rule bases and then exchanges them between both systems, C1.2 - in each of fuzzy rules selected in C1.1, randomly selects one input attribute and then exchanges fuzzy sets that are assigned to those attributes,
- C2 (labelled as *DataBaseExchange*) - randomly selects two membership functions represented by parameters c_1, σ_1 and c_2, σ_2 ; then, it calculates linear combinations of those parameters to obtain their new values:

$$c_{1new} = pc_1 + (1 - p)c_2, \quad c_{2new} = pc_2 + (1 - p)c_1, \quad (5)$$

$$\sigma_{1new} = p\sigma_1 + (1 - p)\sigma_2, \quad \sigma_{2new} = p\sigma_2 + (1 - p)\sigma_1, \quad (6)$$

where $p \in [0, 1]$ is a randomly selected value.

The mutation operator, processing one individual (one rule base), operates in four stages:

- M1 (labelled as *RuleInsert*) - inserts into the rule base a new fuzzy rule with m input attributes $x_i, i = 1, 2, \dots, m$ where $m \leq n$ is a randomly selected positive integer number; then, randomly selected fuzzy sets from appropriate collections are assigned to particular input attributes,
- M2 (labelled as *RuleDelete*) - removes a randomly selected fuzzy rule from the rule base,
- M3 (labelled as *RuleFuzzySetChange*) - randomly selects one fuzzy rule and one input attribute in it and then replaces the fuzzy set describing that attribute by a randomly selected fuzzy set for that attribute,
- M4 (labelled as *FuzzySetChange*) - randomly selects the membership function from the data base, randomly selects one of its parameters, and assigns to it new, randomly selected value.

The fitness function ff has been defined as follows: $ff = const. - Q$, where $const.$ is a constant value selected in such a way that $ff > 0$, and Q is the cost function

$$Q = \frac{1}{Kc} \sum_{k=1}^K \sum_{j=1}^c [\mu_{B'_k}(y_j) - \mu_{B_k^0}(y_j)]^2, \tag{7}$$

where $\mu_{B'_k}(y_j)$ is the desired response of the fuzzy classifier for the k -th sample of the learning data \mathbf{L} (2), and $\mu_{B_k^0}(y_j)$ is the actual response of the classifier (calculated according to (4)).

After the completion of the learning, a two-stage technique for the pruning of the obtained fuzzy rule base can be applied. In the first stage (called “rule pruning”), “weak” fuzzy rules (with lowest numbers of records that activate them) can be removed. In the second stage (called “attribute pruning”), “weak” input attributes in a given rule (those, which can be deleted without decreasing the accuracy of the system) can be removed. The pruning increases the transparency and interpretability of the system.

4 Application to the Selected Classification Problem

The application of the proposed technique to design the genetic fuzzy rule-based classifier for the well known benchmark data set (*Wisconsin Breast Cancer*) will now be presented. The original data set (699 records) has been divided into the learning- and test-parts (465 and 234 records, respectively, randomly selected from the original data set preserving the proportion of the class occurrence).

In the reported experiments, the genetic algorithm with population of 300 individuals and tournament selection method (with the number of individuals participating in the competition [9] equal to 5) supported by elitist strategy as well as with crossover and mutation probabilities equal to 0.8 and 0.5, respectively, has been used. After the learning completion, the above-mentioned two-stage pruning technique has been applied.

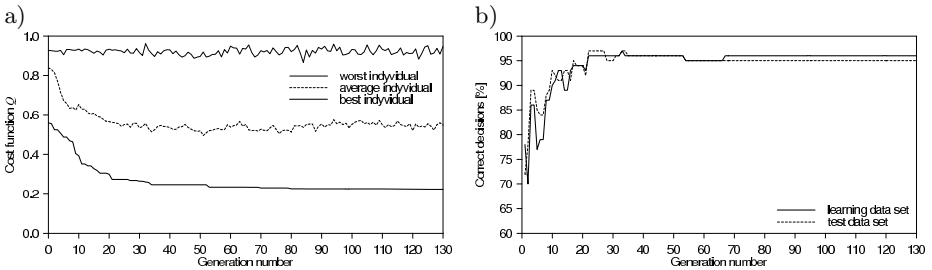


Fig. 1. Cost function Q (9) (a) and percentage of correct decisions (b) vs. generation number plots

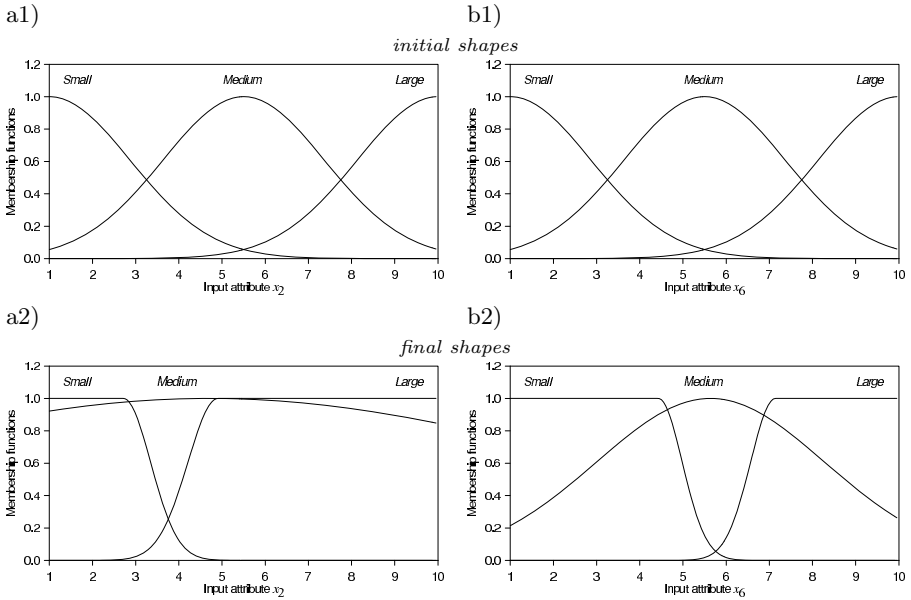


Fig. 2. The initial and final membership functions of fuzzy sets for input attributes: x_2 (a1, a2) and x_6 (b1, b2)

Table 1. Full fuzzy rule base

No.	The content of the fuzzy classification rule	Number of records activating the rule	
		learning data	test data
1	IF x_2 is Small AND x_5 is Large AND x_6 is Large AND x_9 is Small THEN Class 2 (malignant)	1	0
2	IF x_2 is Small AND x_4 is Medium AND x_6 is Small AND x_8 is Large AND x_9 is Small THEN Class 1 (benign)	1	0
3	IF x_2 is Small AND x_6 is Medium AND x_8 is Large THEN Class 2 (malignant)	3	0
4	IF x_2 is Small AND x_6 is Small THEN Class 1 (benign)	294	155
5	IF x_6 is Large THEN Class 2 (malignant)	113	55
6	IF x_2 is Medium AND x_4 is Large AND x_6 is Small AND x_7 is Small AND x_8 is Small THEN Class 2 (malignant)	1	0
7	IF x_2 is Large THEN Class 2 (malignant)	52	24
The learning and test data errors:		0.22071	0.21385
Number (and percentage) of correct decisions:		448 (96%)	224 (95%)

Fig. 1a presents the plot of the cost function Q (9) for the best, worst and average individuals and Fig. 1b shows the percentage of correct, crisp decisions made by the system for the learning and test data versus the number of generations. Fig. 2 presents initial and final shapes of the membership functions of fuzzy sets for input attributes x_2 (*Uniformity of Cell Size*) and x_6 (*Bare Nuclei*)

Table 2. Fuzzy rule base after the “rule pruning”

No.	The content of the fuzzy classification rule	Number of records activating the rule	
		learning data	test data
1	IF x_2 <i>is</i> Small AND x_6 <i>is</i> Small THEN Class 1 (benign)	298	155
2	IF x_6 <i>is</i> Large THEN Class 2 (malignant)	114	55
7	IF x_2 <i>is</i> Large THEN Class 2 (malignant)	53	24
The learning and test data errors:		0.22625	0.21329
Number (and percentage) of correct decisions:		445 (95%)	224 (95%)

Table 3. Fuzzy rule base after the “attribute pruning”

No.	The content of the fuzzy classification rule	Number of records activating the rule	
		learning data	test data
1	IF x_6 <i>is</i> Small THEN Class 1 (benign)	345	179
2	IF x_6 <i>is</i> Large THEN Class 2 (malignant)	210	55
The learning and test data errors:		0.35051	0.30037
Number (and percentage) of correct decisions:		410 (88%)	212 (90%)

Table 4. Comparative analysis with several alternative approaches - percentages of correct decisions for learning (CD_{learn}) and test (CD_{test}) data sets, for the whole data set (CD) and numbers of rules

GFRBC-Pitts ⁽¹⁾	GARS – 60 ⁽²⁾	GARS – 600 ⁽²⁾	GARC – IG ⁽³⁾	GARC – NIG ⁽³⁾
$CD_{learn} = 95\%$, $CD_{test} = 95\%$, 3 rules	$CD = 96.16\%$, 4.58 rules ⁽⁴⁾	$CD = 96.75\%$, 7.76 rules ⁽⁴⁾	$CD = 94.85\%$, 21 rules	$CD = 94.85\%$, 25 rules

⁽¹⁾GFRBC-Pitts - Genetic Fuzzy Rule-Based Classifier based on the Pittsburgh approach (presented in this paper), ⁽²⁾GARS-60, GARS-600 - Genetic Algorithm-based Rule Selection (two variants: 60 and 600 candidate fuzzy rules, respectively) [6], ⁽³⁾GARC-IG, GARC-NIG - Gain based Association Rule Classification systems [2], ⁽⁴⁾average number of rules.

that have been selected as the most important ones (they occur in the reduced rule bases - see Tables 2 and 3).

Table 1 presents the full rule base generated by the proposed approach. It is easy to see that 4 out of 7 rules are very “weak” (they are not activated at all by the test data and are activated by only 1 or 3 records from the learning data).

Therefore, they can be automatically removed from the rule base; new system is presented in Table 2 - it has almost the same accuracy as the full rule base but much higher transparency. It is worth noticing that “Medium”-type fuzzy set for attribute x_2 (see Fig. 2), that almost covers the neighbouring sets “Small” and “Large” occurs only once in a very “weak” rule no. 6 in the full rule base; this rule is then removed from the rule base. An analogous analysis can be carried out for “Medium”-type fuzzy set for attribute x_6 . Table 3 presents the further reduced rule base (after the “attribute pruning” stage).

Table 4 presents the results of comparative analysis with several alternative approaches to design (fuzzy) rule-based classifiers from data. It is evident that from the point of view of the accuracy versus interpretability criterion, the approach presented in this paper demonstrates its superiority over the considered alternative techniques.

5 Conclusions

The modification of the Pittsburg approach to design a fuzzy rule-based classifier from data has been presented in this paper. Original, non-binary crossover and mutation operators have been introduced. The proposed approach does not require any special coding of fuzzy rules and their parameters; a direct representation and processing of fuzzy rules can be made. Also, a two-stage pruning of the obtained fuzzy rule bases has been introduced. The applications of the proposed technique to design the fuzzy rule-based classifier for the well-known benchmark data set (*Wisconsin Breast Cancer*) available from the <http://archive.ics.uci.edu/ml> has been presented. A comparative analysis - from the point of view of the criterion of the accuracy versus interpretability ("measured" by the number of fuzzy rules and the number of input attributes in particular rules) - with several alternative (fuzzy) rule-based classification techniques has also been carried out demonstrating the superiority of the proposed technique.

References

1. Bezdek, J.C.: What is computational intelligence? In: Zurada, J.M., Marks II, R.J., Robinson, C.J. (eds.) *Computational Intelligence: Imitating Life*, pp. 1–12. IEEE Press, New York (1994)
2. Chen, G., Liu, H., Yu, L., Wei, Q., Zhang, X.: A new approach to classification based on association rule mining. *Decision Support Systems* 42(2) (2006)
3. Cordon, O., Herrera, F., Hoffmann, F., Magdalena, L.: *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. World Scientific, Singapore (2001)
4. DeJong, K.: Learning with genetic algorithms: An overview. *Machine Learning* 3(3), 121–138 (1988)
5. Gorzalczany, M.B.: *Computational Intelligence Systems and Applications. Neuro-Fuzzy and Fuzzy Neural Synergisms*. Physica-Verlag, Springer-Verlag Co., Heidelberg (2002)
6. Ishibuchi, H., Yamamoto, T.: Comparison of Heuristic Criteria for Fuzzy Rule Selection in Classification Problems. In: *Fuzzy Optimization and Decision Making*, vol. 3(2). Springer, Netherlands (2004)
7. Jang, J.-S.R., Sun, C.-T., Mizutani, E.: *Neuro-Fuzzy and Soft Computing*. In: *A Computational Approach to Learning and Machine Intelligence*. Prentice-Hall, Upper Saddle River (1997)
8. Kuncheva, L.I.: *Fuzzy Classifier Design*. Physica-Verlag, Springer-Verlag Co., Heidelberg (2000)

9. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. Springer, Heidelberg (1996)
10. Nauck, D., Klawonn, F., Kruse, R.: Foundations of Neuro-Fuzzy Systems. J. Wiley&Sons, Chichester (1997)
11. Pedrycz, W. (ed.): Fuzzy Modelling: Paradigms and Practice. Kluwer Academic Press, Boston (1996)
12. Wang, L.-X.: Adaptive Fuzzy Systems and Control: Design and Analysis. Prentice-Hall, New York (1994)

Automatic and Incremental Generation of Membership Functions

Narjes Hachani, Imen Derbel, and Habib Ounelli

Faculty of Science of Tunis, El Manar 1, 2092, Tunis, Tunisia

`Narjes_hachani@yahoo.fr`,

`imen.derbel@yahoo.fr`,

`habib.ounelli@fst.rnu.tn`

Abstract. The fuzzy set theory is well known as the most adequate framework to model and manage vague terms. In real-world fuzzy database applications, such terms allow to express flexible queries. Moreover, due to the frequent updates of databases, it is highly desirable to perform these updates incrementally. In this paper, we propose an automatic and incremental approach for the generation of membership functions describing vague terms. This approach uses a clustering algorithm to identify automatically the number of these membership functions. Also, it is based on a density function in order to derive the cores of fuzzy sets.

1 Introduction

The fuzzy set theory owes a great deal to human language. When we speak of temperature in terms such as "hot" or "cold" instead of in physical units such as degrees Fahrenheit or Celsius, we can see language becomes a fuzzy variable whose spatial denotation is imprecise. In this sense, fuzzy theory becomes easily understood because it can be made to resemble a high level language instead of a mathematical language. Recently, much research effort [1] have been devoted to develop fuzzy database systems. These systems handle imprecise and vague terms based on fuzzy set theory. A fuzzy set is fully defined by its membership function. Thus, how to determine the membership function is the first question that has to be tackled. During the past years, the generation of membership functions was a task that required knowledge acquisition from an expert [2]. However, using expert knowledge is a subjective behavior. Moreover, obtaining fuzzy knowledge base from an expert is often based on a tedious and unreliable trial and error approach. To cope with these limitations, several automatic approaches [3,4] have been proposed for generating membership functions. Nevertheless, addressing the incremental updating of databases hasn't been reported by none of these works. In this paper, we propose a new approach for incremental and automatic generation of trapezoidal membership functions. Our main contributions are the following:

- The proposed approach reflects the knowledge contained in the data by using an automatic clustering method.

- Our approach identifies automatically the optimal number of membership functions which is determined by a clustering method.
- The proposed approach can be used in fuzzy systems as a stand alone technique that is independent of fuzzy rules generation.
- The generated membership functions can be used in several applications such as flexible querying of databases.
- Our Approach is characterized by its power of handling incrementally the updates of databases.

This paper is organized as follows. In section 2, we describe the proposed approach for automatic generating of membership functions. In section 3, we tackle the incremental aspect of our approach in case of object insertion. In section 4, we report the experimental results conducting to evaluate our method. In section 5, we recall the main points of this paper.

2 Automatic Generation of Membership Functions

Our approach is based on a clustering method. Each obtained cluster will be represented by a fuzzy set. In this work, we assume trapezoidal membership functions as a model because of its popularity and its simplicity [2]. A fuzzy set is characterized by its core and its support. Among these parameters, we choose to determine first the core because it includes the most representative elements of the fuzzy set. Another reason is the possibility of estimating the supports of the fuzzy sets from the generated cores. Thus, our approach is composed of three main phases: generation of a partition, determination of the cores and derivation of the supports.

2.1 Generation of a Partition

A partition is constructed based on the algorithm CLUSTERDB* [5] which is an improvement version of CLUSTER algorithm [6] using a validity index DB* [7]. CLUSTERDB* constructs an initial relative neighborhood graph (RNG) and then tries to divide it into several subgraphs based on a threshold which is dynamically computed. This process is performed iteratively for each obtained subgraph until a stop criterion is reached. According to our clustering method, a cluster C can be defined as a subgraph (X, E) where X is the set of vertices and E is the set of edges. The weight of an edge connecting two vertices x_i and x_j is represented by the euclidian distance, noted $d(x_i, x_j)$.

2.2 Determination of the Cores

The core includes objects that best characterize a cluster. Therefore, it can be represented by the dense part of a cluster which is composed of the cluster's centroid Ce and the set of its dense neighbors. The centroid is represented by the average value of the cluster. If this object is not included in the cluster, we will consider its nearest neighbor.

Identification of centroid's neighbors. Before describing the determination of centroid's neighbors, we introduce some definitions.

1. The diameter of a cluster is defined as the maximum distance between two cluster objects.
2. Let x_i and x_j be two vertices of a cluster C . x_j is a direct neighbor of x_i if it exists an edge connecting x_i and x_j . The set of direct neighbors of a vertex x_i is defined as:

$$V(x_i) = \{x_j \in C \text{ such that } x_j \text{ is a direct neighbor of } x_i\}.$$

3. A density function of a vertex $x_i \in C$ is defined by the following expression [\[8\]](#):

$$De(x_i) = \frac{Diam_C - \frac{1}{Dg(x_i)} \sum_{x_j \in V(x_i)} d(x_i, x_j)}{Diam_C} \quad (1)$$

Where $Dg(x_i)$ is the cardinality of $V(x_i)$. $De(x_i)$ has a high value when the elements of $V(x_i)$ are close to x_i .

4. The density threshold is defined as follows:

$$thresh = \frac{(Dmin_C + Dmax_C)}{2}$$

Where $Dmin_C$ represents the minimal density in C and $Dmax_C$ is the maximal density in C .

5. A dense vertex of a cluster C is an object having a density value greater than the density's threshold of C .

The core includes first the centroid of the cluster. Then, it is extended iteratively with other vertices. Each inserted vertex x_i must satisfy the following properties:

- x_i is a direct neighbor of a vertex included in the core.
- x_i is a dense vertex.

2.3 Supports Generation

The last step involved by our approach is the derivation of supports based on the generated cores. Suppose we intend to construct the membership functions of fuzzy sets for the j^{th} quantitative attribute with a range from min_j to max_j . $[b_{ij}, c_{ij}]$ represents the core of the fuzzy set S_{ij} for this j^{th} attribute. Supports of membership functions are determined as follows. Let the fuzzy set S_{1j} be the fuzzy set associated to the first cluster C_{1j} and having as core $[b_{1j}, c_{1j}]$. Its support is given by :

$$\text{Support of } S_{1j} = [min_j, b_{2j}]$$

For the fuzzy set S_{ij} with the core $[b_{ij}, c_{ij}]$, the support is defined as follows:

$$\text{Support of } S_{ij} = [c_{(i-1)j}, b_{(i+1)j}]$$

For the last fuzzy set S_{kj} with the core $[b_{kj}, c_{kj}]$, the support of the membership function is :

$$\text{Support of } S_{kj} = [c_{(k-1)j}, max_j]$$

3 Handling with Incremental Insertion

The insertion of an object can affect the current partition and membership functions parameters. For this reason, when inserting an object p into a database D , we determine first the appropriate cluster for p based on a cluster property. In a second step, we determine the new membership functions parameters. We will first introduce the definition of a coherent partition.

Coherent partition. let a partition CK composed of K clusters $CK = \{C_1, \dots, C_k\}$. CK is a coherent partition if it satisfies the following property (CP): if two objects x_i^j and x_{i+1}^k belong to two distinct clusters respectively C_i and C_{i+1} , then $d(x_i^j, x_{i+1}^k) > d(x_i^j, x_i^l) \forall l \in [1, |C_i|], l \neq i$.

3.1 Identification of the Appropriate Cluster

Let CK be a partition composed of K clusters. Our purpose is to insert an object p into CK such that the property CP is still verified. According to the value of p , we distinguish the following cases:

1. p is between minimal and maximal border of a cluster C_i . In this case, C_i is the appropriate cluster for p . In fact, if we insert p into the cluster C_i , the obtained partition satisfies the property CP .
2. p is between the maximal border of a cluster C_i and the minimal border of a cluster C_{i+1} . We can distinguish two subclasses:
 - $d(p, C_i) = d(p, C_{i+1})$. In other words, p is equidistant from C_i and C_{i+1} . The property CP is not satisfied when either inserting p into C_i or affecting it to C_{i+1} . Thus, we propose to re-apply the clustering to the whole data set.

Proof. If we suppose that p belong to C_i , let C'_i be the cluster C_i including the object p . p is equidistant from his left neighbor in the cluster C'_i and the first object of the cluster C_{i+1} . Therefore, the property CP is not verified.

- $d(p, C_i) \neq d(p, C_{i+1})$. Let suppose that C_i is the nearest cluster to p . If CP is verified then the appropriate cluster for p is C_i else we re-apply the clustering algorithm.

Proof. If we suppose that p belongs to the farthest cluster C_{i+1} then the distance between p and its right neighbor in C_{i+1} is greater than the distance between the two clusters C_i and C_{i+1} . Consequently, the cluster property CP is not satisfied. So, C_{i+1} is not the appropriate cluster for p .

3.2 Incremental Generation of Membership Functions

After inserting the object p in the appropriate cluster C_i , we check if the membership function parameters will be modified. Algorithm [1](#) describes the proposed approach based on the following assumptions:

- C : the cluster including the inserted object p .
- Nc : the centroid of C after insertion of p .
- $Oinf$, $Osup$: the lower bound and the upper bound of the core before inserting p .
- $Ninf$, $Nsup$: the lower bound and the upper bound of the core after inserting p .
- $Othresh$, $Nthresh$: the density threshold before and after the insertion of p .
- $LOinf$: the left direct neighbor of $Oinf$.
- $ROsup$: the right direct neighbor of $Osup$.

This algorithm generates the new bounds of the core associated to a cluster C . When inserting a new object p , updates depends on threshold's density value, centroid's position and the position of the inserted object p . If the new cluster's centroid do not belong to the old core of the cluster, we must re-apply the core generation algorithm. Else, the new core will either be extended or reduced depending on threshold's density. We distinguish three cases:

1. The threshold is not modified after p insertion: the extension of the core is performed only in two cases:
 - p is the direct left neighbor of $Oinf$. In this case, the function $NewLbound(C, p)$ extends the core with p and its left direct neighbor Lp if they are dense.
 - p is the direct right neighbor of $Osup$. In this case, the function $NewRbound(C, p)$ extends the core with p and its right direct neighbor Rp if they are dense.
2. The threshold is reduced: in this case, the density of some elements at the left neighborhood of $Oinf$ and the right neighborhood of $Osup$ can be modified. In our algorithm, the function $Lneighbors(C, Oinf, Nthresh)$ searches the dense objects at the left neighborhood of $Oinf$. So, it determines the new lower bound of the core. As the same, the function $Rneighbors(C, Osup, Nthresh)$ identifies the dense objects at the right neighborhood of $Osup$. Therefore, it generates the right bound of the core.
3. The threshold increased: the core generation algorithm is re-applied. Indeed, some elements can become not dense because their density will be greater than the old threshold but lower than the new threshold.

4 Experiments

4.1 Experimental Setup

To evaluate the proposed approach, we apply it on the following databases.

Algorithm:Coreupdating**Input:** the cluster C , Nc , p , $Oinf$, $Osup$ **Output:** $Ninf, Nsup$

```

begin
   $Ninf \leftarrow Oinf$ 
   $Nsup \leftarrow Osup$ 
  if  $Nc \in [Oinf, Osup]$  then
    if  $Nthresh = Othresh$  then
      if  $p \text{ not } \in [Oinf, Osup]$  then
        if  $p = LOinf$  then
           $Ninf \leftarrow \text{Newlbound}(C, p)$ 
        else
          if  $p = ROsup$  then
             $Nsup \leftarrow \text{Newupbound}(C, p)$ 
      else
        if  $Nthresh < Othresh$  then
           $Ninf \leftarrow \text{Lneighbors}(C, Oinf, Nthresh)$ 
           $Nsup \leftarrow \text{Rneighbors}(C, Osup, Nthresh)$ 
        else
          Coregeneration( $C$ )
    else
      Coregeneration( $C$ )
end

```

1. *Census Income* DB [9] includes 606 objects. We are interested in the value of age attribute which allows to identify three clusters.
2. *Hypothyroid* DB [9] includes 1000 objects. We are interested in the values of the TSH which allows to identify two clusters.
3. *Thyroid* DB [9] includes 5723 objects. We are interested in the values of the TSH which allows to identify two clusters.

4.2 Experiments on Membership Functions Generation

The experimental results are presented in tables 1 and 2.

Table 1. Membership functions parameters for *Hypothyroid* DB

Clusters	Domain	Core	MF parameters
Cluster1	[0.005, 28]	[0.005, 18]	0.005, 18, 143
Cluster2	[143, 199]	[143, 160]	18, 143, 199

Table 1 shows that the algorithm *CLUSTERDB** identifies the appropriate number of clusters in the database *Hypothyroid*. Consequently, it allows to

determine the adequate number of membership functions. We have found similar results for the other described DB. However, due to space limit, we have presented only the result associated to the DB *Hypothyroid*. Table 2 shows that our approach for Membership Function Generation is not time-consuming.

Table 2. Running time in milliseconds

DB	<i>CensusIncome</i>	<i>Hypothyroid</i>	<i>Thyroid</i>
objects's number	606	1000	5723
Time in ms	66	190	736

4.3 Experiments on Incremental Insertion

We perform the following experiments, after inserting some objects in the DB, using the proposed incremental algorithm. For each DB, objects are added successively.

Table 3. Updates after insertion in Hypothyroid DB

p	Appcluster Core	MF's parameters
50	$C1$	$C1 : [0.005, 27]$ $C2 : [143, 160]$
100	$C2$	$C1 : [0.005, 143]$ $C2 : [27, 199]$ $C1 : [0.005, 143]$ $C2 : [143, 199]$

Table 4. Updates after insertion in Census Income DB

p	Appcluster Core	MF's parameters
77	$C1$	$C1 : [1, 78]$ $C2 : [83, 86]$ $C3 : [89, 90]$
87	$C2$	$C1 : [0.005, 27]$ $C2 : [143, 199]$ $C3 : [89, 90]$ $C1 : [0.005, 143]$ $C2 : [27, 199]$ $C3 : [87, 90]$

Tables 3 and 4 show that the inserted values are affected to a cluster of the initial partition. They also show the core and the membership functions parameters for each cluster. Table 5 illustrates the case of reclustering. In order to evaluate the effectiveness of the reclustering:

- We suppose that the inserted value p is included into a cluster C of the initial partition and we compute DB^* index (DB^*1) of the obtained partition $P1$. The cluster C is determined as follows: if p is equidistant from two clusters then we use the silhouette index [7] to choose a cluster for p . Else, p is inserted in the nearest cluster.

Table 5. Reclustering after insertion in Hypothyroid DB

p	Newpartition	DB*1	DB*2
85	$C1 : [0.005, 100]$ $C2 : [143, 199]$	0,267	0,157

- We reapply the clustering algorithm and we compute the value of DB* (DB*2) associated to the new partition $P2$.

According to table 4, the quality of the partition $P2$ is better than $P1$'s quality (DB*2 is lower than DB*1). Consequently, the reclustering is the appropriate choice.

5 Conclusion

In this paper, we have proposed an automatic and incremental approach for generating trapezoidal membership functions. Our approach allows to identify the appropriate number of membership functions. The major advantage is the handling of incremental aspect. Thus, when the database is updated, there is no need to redo all the steps of our algorithm from the scratch.

References

- Galindo, J., Urrutia, A., Piattini, M.: Fuzzy Databases: Modeling, Design and Implementation. IGI Publishing (2006)
- Bilgic, T., Turksen, I.B.: Measurement of Membership Functions: Theoretical and Empirical Work. Handbook of Fuzzy Sets and Systems (1997)
- Chen, S., Tsai, F.: A new method to construct membership functions and generate fuzzy rules from training instances. J. Inf. Manag. Sc. 16(2), 47–72 (2005)
- Tudorie, C., Frangu, L., Stefanscu, D.: Automatic Modelling of the Linguistic Values for Databases Fuzzy Querying. The Annals of Dunaria De JOS University of Galati. (2007)
- Hachani, N., Ounelli, H.: Improving Cluster Method Quality by Validity Indices. In: FLAIRS Conference, pp. 479–483 (2007)
- Bandyopadhyay, S.: An automatic shape independent clustering techniques. Pattern Recognition 37, 33–45 (2004)
- Halkidi, M., Vazirgiannis, M.: Clustering Validity Assessment: Finding the optimal partitioning of a data set. In: Proceeding of ICDM (2001)
- Guenoche, A.: Clustering by vertex density in the graph. In: Proceeding of IFCS Congress Classification, pp. 15–24 (2004)
- Blake, C., Merz, C.: UCI repository of machine learning databases, <http://www.ics.uci.edu/~mllearn/MLRepository.html>

A Multi-criteria Evaluation of Linguistic Summaries of Time Series via a Measure of Informativeness

Anna Wilbik* and Janusz Kacprzyk

Systems Research Institute, Polish Academy of Sciences
ul. Newelska 6, 01-447 Warsaw, Poland
{kacprzyk,wilbik}@ibspan.waw.pl

Abstract. We extend our works of deriving linguistic summaries of time series using a fuzzy logic approach to linguistic summarization. We proceed towards a multicriteria analysis of summaries by assuming as a quality criterion Yager's measure of informativeness that combines in a natural way the measures of truth, focus and specificity, to obtain a more advanced evaluation of summaries. The use of the informativeness measure for the purpose of a multicriteria evaluation of linguistic summaries of time series seems to be an effective and efficient approach, yet simple enough for practical applications. Results on the summarization of quotations of an investment (mutual) fund are very encouraging.

1 Introduction

This paper is an extension of the our previous works (cf. Kacprzyk, Wilbik and Zadrozny [1], and Kacprzyk and Wilbik [2,3,4,5], Kacprzyk and Zadrozny [6,7]) in which fuzzy logic, computing with words, and natural language generation were employed to derive a linguistic summary of a time series in the sense of a verbalization of how the time series behaves in regards to both the temporal evolution of values, their variability, etc. It is a different approach to the analysis of time series than the usual forecasting/prediction analyses, and is rather focused on providing tools and techniques for supporting decision making by a human analyst. The use of linguistic summaries is an example of verbalization of the results of data analysis which is less common than visualization.

First, one should notice that in virtually all non-trivial practical problems, notably in finance (to be more specific, investments in mutual funds considered by us) a decision support paradigm is employed, i.e. a decision is made by a human analyst based on some results of data analysis, modeling, calculations, etc. provided by the system. Here we consider some verbal summary of the past, with respect to the time series (more specifically, quotations of a mutual fund), as additional information that may be of much use to the analyst.

* Supported by the Polish Ministry of Science and Higher Education under Grant No. NN516 4309 33.

There is an ample rationale for this approach. On the one hand, in any mutual fund information leaflet, there is a disclaimer like “Past performance is no indication of future returns”, which is true. However, on the other hand (cf. “Past Performance Does Not Predict Future Performance” [8]), they also state: “. . . according to an Investment Company Institute study, about 75% of all mutual fund investors mistakenly use short-term past performance as their primary reason for buying a specific fund”. Similarly, in “Past performance is not everything” [9], there is “. . . disclaimers apart, as a practice investors continue to make investments based on a scheme’s past performance. To make matters worse, fund houses are only too pleased to toe the line by actively advertising the past performance of their schemes leading investors to conclude that it is the single-most important parameter (if not the most important one) to be considered while investing in a mutual fund scheme”. Moreover, in “New Year’s Eve: Past performance is no indication of future return” [10], they say “. . . if there is no correlation between past performance and future return, why are we so drawn to looking at charts and looking at past performance? I believe it is because it is in our nature as human beings . . . because we don’t know what the future holds, we look toward the past . . .”, or in [11]: “. . . Does this mean you should ignore past performance data in selecting a mutual fund? No. But it does mean that you should be wary of how you use that information . . . While some research has shown that consistently good performers continue to do well at a better rate than marginal performers, it also has shown a much stronger predictive value for consistently bad performers . . . *Lousy performance in the past is indicative of lousy performance in the future. . .*”. And, further: in [12], we have: “. . . there is an important role that past performance can play in helping you to make your fund selections. While you should disregard a single aggregate number showing a fund’s past long-term return, you can learn a great deal by studying the *nature of its past returns*. Above all, look for consistency.” There are a multitude of similar opinions expressed by top investment theorists, practitioners and advisors.

We use a slightly unorthodox approach to the summarization of the past performance of an investment fund by using a natural language summary exemplified by “most of long trends are slightly increasing”. We use Yager’s [14,15] approach to linguistic summarization of numerical data that is based on a calculus of linguistically quantified propositions. An important new directions, initiated by Kacprzyk and Zadrozny [16] is here a suggestion that a proper setting in which to derive linguistic data summaries may be within natural language generation (NLG), a modern, rapidly developing field of computer science and computational linguistics. This will not be discussed in more details here.

The analysis of time series data involves different elements but we concentrate on the specifics of our approach. First, we need to identify the consecutive parts of time series within which the data exhibit some uniformity as to their variability. Here, they are called trends, and described by straight line segments. That is, we perform first a piece-wise linear approximation of a time series and present time series data as a sequence of trends. The (linguistic) summaries of time series refer to the (linguistic) summaries of (partial) trends as meant above.

The next step is an aggregation of the (characteristic features of) consecutive trends over an entire time span (horizon) assumed. We follow the idea initiated by Yager [14,15] and then shown more profoundly and in an implementable way in Kacprzyk and Yager [20], and Kacprzyk, Yager and Zadrożny [21,22], that the most comprehensive and meaningful will be a linguistic quantifier driven aggregation resulting in linguistic summaries exemplified by “Most trends are short” or “Most long trends are increasing” which are easily derived and interpreted using Zadeh’s fuzzy logic based calculus of linguistically quantified propositions. A new quality, and an increased generality was obtained by using Zadeh’s [23] protoforms as proposed by Kacprzyk and Zadrożny [7].

Here we employ the classic Zadeh’s fuzzy logic based calculus of linguistically quantified propositions in which the degree of truth (validity) is the most obvious and important quality indicator. Some other indicators like a degree of specificity, focus, fuzziness, etc. have also been proposed by Kacprzyk and Wilbik [23,4,5]. The results obtain clearly indicate that multiple quality criteria of linguistic summaries of time series should be taken into account, and this makes the analysis obviously much more difficult.

As the first step towards an intended comprehensive multicriteria assessment of linguistic summaries of time series, we propose here a very simple, effective and efficient approach, namely to use quite an old, maybe classic Yager’s [24] proposal on an informativeness measure of a linguistic summary which combines, via an appropriate aggregation operator, the degree of truth, focus and specificity.

We illustrate our analysis on a linguistic summarization of daily quotations over an 8 year period of an investment (mutual) fund. We present the characteristic features of trends derived under some reasonable granulations, variability, trend duration, etc.

The paper is in line with some other modern approaches to linguistic summarization of time series. First, one should refer to the *SumTime* project [25]. A relation between linguistic data summaries and NLG is discussed by Kacprzyk and Zadrożny [16,26].

2 Linguistic Data Summaries

As a *linguistic summary of data (base)* we understand a (usually short) sentence (or a few sentences) that captures the very essence of the set of data, that is numeric, large, and because of its size is not comprehensible for human being.

We use Yager’s basic approach [14]. A linguistic summary includes: (1) a summarizer P (e.g. *low* for attribute *salary*), (2) a quantity in agreement Q , i.e. a linguistic quantifier (e.g. *most*), (3) truth (validity) T of the summary and optionally, (4) a qualifier R (e.g. *young* for attribute *age*).

Thus, a linguistic summary may be exemplified by “*Most of employees earn low salary*” $T = 0.7$ or in richer (extended) form, including a qualifier (e.g. *young*), by “*Most of young employees earn low salary*” $T = 0.82$. Thus, basically

the core of a linguistic summary is a linguistically quantified proposition in the sense of Zadeh [23] which may be written, respectively as

$$Qy's \text{ are } P \qquad QRy's \text{ are } P \qquad (1)$$

3 Linguistic Summaries of Trends

In our first approach we summarize the trends (segments) extracted from time series. Therefore as the first step we need to extract the segments. We assume that segment is represented by a fragment of straight line, because such segments are easy for interpretation. There are many algorithms for the piecewise linear segmentation of time series data, including e.g. on-line (sliding window) algorithms, bottom-up or top-down strategies (cf. Keogh [18,19]).

We consider the following three features of (global) trends in time series: (1) dynamics of change, (2) duration, and (3) variability. By *dynamics of change* we understand the speed of change of the consecutive values of time series. It may be described by the slope of a line representing the trend, represented by a linguistic variable. *Duration* is the length of a single trend, and is also represented by a linguistic variable. *Variability* describes how “spread out” a group of data is, computed using some statistical measures, e.g. (1) the range, (2) the interquartile range (IQR), (3) the variance, (4) the standard deviation, and (5) the mean absolute deviation (MAD). This is also a linguistic variable.

For practical reasons for all we use a fuzzy granulation (cf. Bathyrshin et al. [27,28]) to represent the values by a small set of linguistic labels as, e.g.: increasing, constant, decreasing, etc. These values are equated with fuzzy sets.

For clarity and convenience we employ Zadeh’s [29] protoforms for dealing with linguistic summaries [7]. We have two types of protoforms of linguistic summaries of trends:

– a short form:

$$\text{Among all segments, } Q \text{ are } P \qquad (2)$$

e.g.: “Among all segments, *most* are *slowly increasing*”.

– an extended form:

$$\text{Among all } R \text{ segments, } Q \text{ are } P \qquad (3)$$

e.g.: “Among all *short* segments, *most* are *slowly increasing*”.

The quality of linguistic summaries can be evaluated in many different ways, eg. using the degree of truth, specificity, appropriateness or others. Yager [24] proposed measure of informativeness, a measure that evaluates the amount of information hidden in the summary. This measure is interesting as it aggregates some quality criteria, namely the truth value, degree of specificity and degree of focus in the case of extended form summaries.

Truth value

The truth value (a degree of truth or validity), introduced by Yager in [14], is the basic criterion describing the degree of truth (from [0, 1]) to which a linguistically quantified proposition equated with a linguistic summary is true.

Using Zadeh’s calculus of linguistically quantified propositions [23] it is calculated in dynamic context using the same formulas as in the static case. Thus, the truth value is calculated for the simple and extended form as, respectively:

$$T(\text{Among all } y\text{'s, } Q \text{ are } P) = \mu_Q \left(\frac{1}{n} \sum_{i=1}^n \mu_P(y_i) \right) \tag{4}$$

$$T(\text{Among all } Ry\text{'s, } Q \text{ are } P) = \mu_Q \left(\frac{\sum_{i=1}^n \mu_R(y_i) \wedge \mu_P(y_i)}{\sum_{i=1}^n \mu_R(y_i)} \right) \tag{5}$$

where \wedge is the minimum operation (more generally it can be another appropriate operator, notably a t -norm cf. Kacprzyk, Wilbik and Zadrozny [30]).

Degree of specificity

The concept of specificity provides a measure of the amount of information contained in a fuzzy subset or possibility distribution. The specificity measure evaluates the degree to which a fuzzy subset points to one and only one element as its member [31].

We will consider the original Yagers proposal [31], in which specificity measures the degree to which a fuzzy subset contains one and only one element.

In [32] Yager proposed a measure of specificity as

$$Sp(A) = \int_0^{\alpha_{max}} \frac{1}{card(A_\alpha)} d\alpha \tag{6}$$

where α_{max} is the largest membership grade in A , A_α is the α -level set of A , (i.e. $A_\alpha = \{x : A(x) \geq \alpha\}$) and $cardA_\alpha$ is the number of elements in A_α .

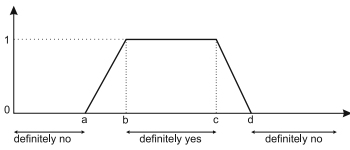


Fig. 1. A trapezoidal membership function of a set

In our summaries to define the membership functions of the linguistic values we use trapezoidal functions, as they are sufficient in most applications [33]. Moreover, they can be very easily interpreted and defined by a user not familiar with fuzzy sets and logic, as in Fig. 1.

To represent a fuzzy set with a trapezoidal membership function we need to store only four numbers, a , b , c and d . Usage such a definition of a fuzzy set is a compromise between cointension and computational complexity. In such a case measure of specificity of a fuzzy set A is

$$Sp(A) = 1 - \frac{c + d - (a + b)}{2} \tag{7}$$

Degree of focus

The extended form of linguistic summaries (3) limit by itself the search space as the search is performed in a limited subspace of all (most) trends that fulfill an additional condition specified by qualifier R . The very essence of the degree of focus introduced in 5 is to give the proportion of trends satisfying property R to all trends extracted from the time series.

The degree of focus is similar in spirit to a degree of covering 4, however it measures how many trends fulfill property R . That is, we focus our attention on such trends, fulfilling property R . The degree of focus makes obviously sense for the extended form summaries only, and is calculated as:

$$d_f(\text{Among all } R\text{'s } Q \text{ are } P) = \frac{1}{n} \sum_{i=1}^n \mu_R(y_i) \quad (8)$$

If the degree of focus is high, then we can be sure that such a summary concerns many trends, so that it is more general. However, if the degree of focus is low, then such a summary describes a (local) pattern seldom occurring.

Measure of informativeness

The idea of the measure of informativeness(cf. Yager 24) may be summarized as follows. Suppose we have a data set, whose elements are from measurement space X . One can say that the data set itself is its own most informative description, and any other summary implies a loss of information. So, a natural question is whether a particular summary is informative, and to what extent.

Yager 24 proposed the following measure for a simple form summary

$$I(\text{Among all } y\text{'s } Q \text{ are } P) = (T \cdot Sp(Q) \cdot Sp(P)) \vee ((1 - T) \cdot Sp(Q^c) \cdot Sp(P^c)) \quad (9)$$

where P^c is the negation of P , i.e. $\mu_{P^c}(\cdot) = 1 - \mu_P(\cdot)$ and Q^c is the negation of Q . $Sp(\cdot)$ is specificity of that fuzzy set.

For the extended form summary we propose the following measure

$$I(\text{Among all } R\text{'s } Q \text{ are } P) = (T \cdot Sp(Q) \cdot Sp(P) \cdot Sp(R) \cdot d_f) \vee ((1 - T) \cdot Sp(Q^c) \cdot Sp(P^c) \cdot Sp(R) \cdot d_f) \quad (10)$$

where d_f is the degree of focus and the rest is defined as previously.

4 Numerical Results

The method proposed in this paper was tested on data on quotations of an investment (mutual) fund that invests at least 50% of assets in shares listed at the Warsaw Stock Exchange.

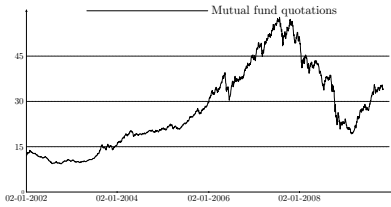


Fig. 2. Daily quotations of an investment fund in question

PLN 2.32, while the biggest daily decrease was equal to PLN 3.46. We illustrate the method proposed by analyzing the absolute performance of a given investment fund, and not against benchmarks, for illustrativeness.

We obtain 353 extracted trends, with the shortest of 1 time unit only the longest – 71. We assume 3 labels only for each attribute.

Data shown in Fig. 2 were collected from January 2002 until the end of October 2009 with the value of one share equal to PLN 12.06 in the beginning of the period to PLN 33.29 at the end of the time span considered (PLN stands for the Polish Zloty). The minimal value recorded was PLN 9.35 while the maximal one during this period was PLN 57.85. The biggest daily increase was equal to

Table 1. Some results obtained for 3 labels only for each attribute

linguistic summary	\mathcal{T}	d_f	\mathcal{I}
Among all low y 's, most are short	1.0000	0.7560	0.2736
Among all decreasing y 's, almost all are short	1.0000	0.2720	0.1166
Among all increasing y 's, almost all are short	1.0000	0.2668	0.1143
Among all short and increasing y 's, most are low	1.0000	0.2483	0.1444
Among all decreasing y 's, most are low	0.9976	0.2720	0.0596
Among all short and decreasing y 's, most are low	0.9969	0.2645	0.1533
Among all increasing y 's, most are short and low	0.9860	0.2668	0.1352
Among all y 's, most are short	0.9694	–	0.5012
Among all decreasing y 's, most are short and low	0.9528	0.2720	0.1333
Among all y 's, most are low	0.9121	–	0.3512
Among all short and constant y 's, most are low	0.8408	0.2741	0.1597
Among all moderate y 's, most are short	0.8274	0.2413	0.0619
Among all constant y 's, most are low	0.8116	0.4612	0.1239
Among all medium and constant y 's, most are low	0.7646	0.1265	0.0650
Among all medium y 's, most are low	0.7167	0.1524	0.0372

The summaries in the table are ordered according to the truth value, and then by the degree of focus. Generally, the simple form summaries, (e.g. Among all y 's, most are short) have a higher measure of informativeness, as they describe whole data set. The measure of informativeness of the extended form summaries is smaller, because they describe only a subset of the data.

This measure considers also number and quality of the adjectives used. For instance, for “Among all decreasing y 's, almost all are short”, with $\mathcal{I} = 0.1166$, and “Among all decreasing y 's, most are short and low”, with $\mathcal{I} = 0.1333$, the latter, although it has a bit smaller truth value, is more informative, as it provides additional information. This is a more general property resulting from our experiments.

It seems that the measure of informativeness is a good evaluation of the amount of information carried by the summary. Moreover, as it combines the measure of truth, focus and specificity in a intuitively appealing yet simple way, may be viewed as an effective and efficient tools for a multi-criteria assessment of linguistic summaries of times series.

5 Concluding Remarks

We extended our approach to the linguistic summarization of time series towards a multicriteria analysis of summaries by assuming as a quality criterion Yager's measure of informativeness that combines in a natural way the measures of truth, focus and specificity. Results on the summarization of quotations of an investment (mutual) fund are very encouraging.

References

1. Kacprzyk, J., Wilbik, A., Zadrożny, S.: Linguistic summarization of time series using a fuzzy quantifier driven aggregation. *Fuzzy Sets and Systems* 159(12), 1485–1499 (2008)
2. Kacprzyk, J., Wilbik, A.: Linguistic summarization of time series using fuzzy logic with linguistic quantifiers: a truth and specificity based approach. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008*. LNCS (LNAI), vol. 5097, pp. 241–252. Springer, Heidelberg (2008)
3. Kacprzyk, J., Wilbik, A.: Linguistic summarization of time series using linguistic quantifiers: augmenting the analysis by a degree of fuzziness. In: *Proceedings of 2008 IEEE WCCI*, pp. 1146–1153. IEEE Press, Los Alamitos (2008)
4. Kacprzyk, J., Wilbik, A.: A new insight into the linguistic summarization of time series via a degree of support: elimination of infrequent patterns. In: Dubois, D., et al. (eds.) *Soft Methods for Handling Variability and Imprecision*, pp. 393–400. Springer, Heidelberg (2008)
5. Kacprzyk, J., Wilbik, A.: Towards an efficient generation of linguistic summaries of time series using a degree of focus. In: *Proceedings of NAFIPS 2009* (2009)
6. Kacprzyk, J., Zadrożny, S.: FQUERY for Access: fuzzy querying for a windows-based dbms. In: Bosc, P., et al. (eds.) *Fuzziness in Database Management Systems*, pp. 415–433. Springer, Heidelberg (1995)
7. Kacprzyk, J., Zadrożny, S.: Linguistic database summaries and their proto-forms: toward natural language based knowledge discovery tools. *Information Sciences* 173, 281–304 (2005)
8. Past performance does not predict future performance, http://www.freemoneyfinance.com/2007/01/past_performanc.html
9. Past performance is not everything, <http://www.personalfn.com/detail.asp?date=9/1/2007&story=3>
10. New year's eve: past performance is no indication of future return stockcasting, blogspot.com/2005/12/new-years-evepast-performance-is-no.html
11. Myers, R.: Using past performance to pick mutual funds. *Nation's Business* (October 1997), findarticles.com/p/articles/mi_m1154/is_n10_v85/ai_19856416
12. Bogle, J.C.: *Common Sense on Mutual Funds: New Imperatives for the Intelligent Investor*. Wiley, New York (1999)
13. Securities, U., Commission, E.: Mutual fund investing: Look at more than a fund's past performance, <http://www.sec.gov/investor/pubs/mfperform.htm>
14. Yager, R.R.: A new approach to the summarization of data. *Information Sciences* 28, 69–86 (1982)
15. Yager, R.R.: On linguistic summaries in data. In: Piatetsky-Shapiro, G., et al. (eds.) *Knowledge Discovery in Databases*, pp. 347–363. MIT Press, Cambridge (1991)

16. Kacprzyk, J., Zadrozny, S.: Data mining via protoform based linguistic summaries: some possible relations to natural language generation. In: 2009 IEEE Symposium Series on Computational Intelligence Proceedings, pp. 217–224 (2009)
17. Sklansky, J., Gonzalez, V.: Fast polygonal approximation of digitized curves. *Pattern Recognition* 12(5), 327–331 (1980)
18. Keogh, E., Chu, S., Hart, D., Pazzani, M.: An online algorithm for segmenting time series. In: Proceedings of the 2001 IEEE ICDM (2001)
19. Keogh, E., Chu, S., Hart, D., Pazzani, M.: Segmenting time series: A survey and novel approach. In: Last, M., et al. (eds.) *Data Mining in Time Series Databases*, World Scientific Publishing, Singapore (2004)
20. Kacprzyk, J., Yager, R.R.: Linguistic summaries of data using fuzzy logic. *International Journal of General Systems* 30, 33–154 (2001)
21. Kacprzyk, J., Yager, R.R., Zadrozny, S.: A fuzzy logic based approach to linguistic summaries of databases. *International Journal of Applied Mathematics and Computer Science* 10, 813–834 (2000)
22. Kacprzyk, J., Yager, R.R., Zadrozny, S.: Fuzzy linguistic summaries of databases for an efficient business data analysis and decision support. In: Abramowicz, W. (ed.) *Knowledge Discovery for Business Information Systems*, pp. 129–152. Kluwer, Dordrecht (2001)
23. Zadeh, L.A.: Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy Sets and Systems* 9(2), 111–127 (1983)
24. Yager, R.R., Ford, K.M., Cañas, A.J.: An approach to the linguistic summarization of data, pp. 456–468. Springer, Heidelberg (1990)
25. Sripada, S., Reiter, E., Davy, I.: Sumtime-mousam: Configurable marine weather forecast generator. *Expert Update* 6(3), 4–10 (2003)
26. Kacprzyk, J., Zadrozny, S.: Computing with words is an implementable paradigm: fuzzy queries, linguistic data summaries and natural language generation. *IEEE Transactions on Fuzzy Systems* (forthcoming)
27. Batyrshin, I.: On granular derivatives and the solution of a granular initial value problem. *International Journal Applied Mathematics and Computer Science* 12(3), 403–410 (2002)
28. Batyrshin, I., Sheremetov, L.: Perception based functions in qualitative forecasting. In: Batyrshin, I., et al. (eds.) *Perception-based Data Mining and Decision Making in Economics and Finance*. Springer, Heidelberg (2006)
29. Zadeh, L.A.: A prototype-centered approach to adding deduction capabilities to search engines – the concept of a protoform. In: Proceedings of NAFIPS 2002, pp. 523–525 (2002)
30. Kacprzyk, J., Wilbik, A., Zadrozny, S.: Linguistic summarization of time series under different granulation of describing features. In: Kryszkiewicz, M., Peters, J.F., Rybiński, H., Skowron, A., et al. (eds.) *RSEISP 2007. LNCS (LNAI)*, vol. 4585, pp. 230–240. Springer, Heidelberg (2007)
31. Yager, R.R.: On measures of specificity. In: Kaynak, O., et al. (eds.) *Computational Intelligence: Soft Computing and Fuzzy-Neuro Integration with Applications*, pp. 94–113. Springer, Heidelberg (1998)
32. Yager, R.R.: Measuring tranquility and anxiety in decision making: An application of fuzzy sets. *International Journal of General Systems* 8, 139–146 (1982)
33. Zadeh, L.A.: Computation with imprecise probabilities. In: *IPMU 2008* (2008)

Negative Correlation Learning of Neuro-fuzzy System Ensembles

Marcin Korytkowski^{1,2} and Rafał Scherer^{1,3}

¹ Department of Computer Engineering, Częstochowa University of Technology
al. Armii Krajowej 36, 42-200 Częstochowa, Poland

<http://kik.pcz.pl>

² Olsztyn Academy of Computer Science and Management
ul. Artyleryjska 3c, 10-165 Olsztyn, Poland

<http://www.owskiiz.edu.pl/>

³ Academy of Management (SWSPiZ), Institute of Information Technology,
ul. Sienkiewicza 9, 90-113 Łódź, Poland

<http://www.swspiz.pl/>

marcin.korytkowski@kik.pcz.pl, rafal@ieee.org

Abstract. Ensembles of classifiers are sets of machine learning systems trained for the same task. The outputs of the systems are combined by various methods to obtain the classification result. Ensembles are proven to perform better than member weak learners. There are many methods for creating the ensembles. Most popular are Bagging and Boosting. In the paper we use the negative correlation learning to create an ensemble of Mamdani-type neuro-fuzzy systems. Negative correlation learning is a method which tries to decorrelate particular classifiers and to keep accuracy as high as possible. Neuro-fuzzy systems are good candidates for classification and machine learning problems as the knowledge is stored in the form of the fuzzy rules. The rules are relatively easy to create and interpret for humans, unlike in the case of other learning paradigms e.g. neural networks.

1 Introduction

There are many methods to improve accuracy of learning systems. One family of such methods are ensemble techniques, in which accuracy is improved by combining several weak learners. Most of such techniques are metalearning methods and require standard machine learning systems and learning algorithms as members of the ensemble. In the paper we use neuro-fuzzy systems [4][7][9][11][12] as the base systems. Neuro-fuzzy systems (NFS) are synergistic fusion of fuzzy logic and neural networks, hence NFS can learn from data and retain the inherent interpretability of fuzzy logic. The knowledge in the form of fuzzy rules is transparent and high accuracy performance is achieved. NFS can use almost the same learning methods and achieve the same accuracy as neural networks yet the knowledge in the form of fuzzy rules is easily interpretable for humans. Most popular NFS are Mamdani type linguistic NFS, where consequents and antecedents are related by the min operator or generally by a t-norm. In the paper we use the most popular gradient technique – the backpropagation algorithm connected with the negative correlation learning (NFS) [5][6] which is a meta-learning algorithm for

creating negatively correlated ensembles. The parameters of the neuro-fuzzy systems are modified by the backpropagation influenced by the NFS to make the ensemble to be uncorrelated.

2 Mamdani-Type Neuro-fuzzy Systems

In this section logical-type neuro fuzzy systems will be described. We consider multi-input-single-output fuzzy system mapping $\mathbf{X} \rightarrow Y$, where $\mathbf{X} \subset R^n$ and $Y \subset R$. Theoretically, the system is composed of a fuzzifier, a fuzzy rule base, a fuzzy inference engine and a defuzzifier. The fuzzifier performs a mapping from the observed crisp input space $\mathbf{X} \subset R^n$ to a fuzzy set defined in X . The most commonly used fuzzifier is the singleton fuzzifier which maps $\bar{\mathbf{x}} = [\bar{x}_1, \dots, \bar{x}_n] \in X$ into a fuzzy set $A' \subseteq X$ characterized by the membership function

$$\mu_{A'}(x) = \begin{cases} 1 & \text{if } x = \bar{x} \\ 0 & \text{if } x \neq \bar{x} \end{cases} \quad (1)$$

Equation (1) means that, in fact, we get rid of the fuzzifier. The knowledge of the system is stored in the fuzzy rule base which consists of a collection of N fuzzy IF-THEN rules in the form

$$R^{(k)} : \begin{cases} \text{IF} & x_1 \text{ is } A_1^k \text{ AND} \\ & x_2 \text{ is } A_2^k \text{ AND } \dots \\ & x_n \text{ is } A_n^k \\ \text{THEN} & y \text{ is } B^k \end{cases} \quad (2)$$

or

$$R^{(k)}: \text{IF } \mathbf{x} \text{ is } A^k \text{ THEN } y \text{ is } B^k \quad (3)$$

where $\mathbf{x} = [x_1, \dots, x_n] \in \mathbf{X}$, $y \in Y$, $A^k = A_1^k \times A_2^k \times \dots \times A_n^k$, $A_1^k, A_2^k, \dots, A_n^k$ are fuzzy sets characterized by membership functions $\mu_{A_i^k}(x_i)$, $i = 1, \dots, n$, $k = 1, \dots, N$, whereas B^k are fuzzy sets characterized by membership functions $\mu_{B^k}(y)$, $k = 1, \dots, N$. The firing strength of the k -th rule, $k = 1, \dots, N$, is defined by

$$\tau_k(\bar{\mathbf{x}}) = \prod_{i=1}^n \left\{ \mu_{A_i^k}(\bar{x}_i) \right\} = \mu_{A^k}(\bar{\mathbf{x}}) \quad (4)$$

The defuzzification is realized by the following formula

$$\bar{y} = \frac{\sum_{r=1}^N \bar{y}^r \cdot \mu_{\bar{B}^r}(\bar{y}^r)}{\sum_{r=1}^N \mu_{\bar{B}^r}(\bar{y}^r)}. \quad (5)$$

The membership functions of fuzzy sets \bar{B}^r , $r = 1, 2, \dots, N$, are defined using the following formula:

$$\mu_{\bar{B}^r}(y) = \sup_{\mathbf{x} \in \mathbf{X}} \left\{ \mu_{A^r}(\mathbf{x}) \ast \mu_{A^r \rightarrow B^r}(\mathbf{x}, y) \right\}. \quad (6)$$

With singleton type fuzzification, the formula takes the form

$$\mu_{\bar{B}^r}(y) = \mu_{A^r \rightarrow B^r}(\bar{\mathbf{x}}, y) = T(\mu_{A^r}(\bar{\mathbf{x}}), \mu_{B^r}(y)). \quad (7)$$

Since

$$\mu_{A^r}(\bar{x}) = \prod_{i=1}^n (\mu_{A_i^r}(\bar{x}_i)), \tag{8}$$

we have

$$\mu_{\bar{B}^r}(y) = \mu_{A^r \rightarrow B^r}(\bar{x}, y) = T \left[\prod_{i=1}^n (\mu_{A_i^r}(\bar{x}_i)), \mu_{B^r}(y) \right], \tag{9}$$

where T is any t -norm. Because

$$\mu_{B^r}(\bar{y}^r) = 1 \tag{10}$$

and

$$T(a, 1) = a, \tag{11}$$

we obtain the following formula

$$\mu_{\bar{B}^r}(\bar{y}^r) = \prod_{i=1}^n (\mu_{A_i^r}(\bar{x}_i)). \tag{12}$$

Finally we obtain

$$\bar{y} = \frac{\sum_{r=1}^N \bar{y}^r \cdot \prod_{i=1}^n (\mu_{A_i^r}(\bar{x}_i))}{\sum_{r=1}^N \prod_{i=1}^n (\mu_{A_i^r}(\bar{x}_i))}. \tag{13}$$

Input linguistic variables are described by means of Gaussian membership functions, that is

$$\mu_{A_i^r}(x_i) = \exp \left[- \left(\frac{x_i - \bar{x}_i^r}{\sigma_i^r} \right)^2 \right], \tag{14}$$

If we apply the Larsen (product) rule of inference, we will get the following formula

$$\bar{y} = \frac{\sum_{r=1}^N \bar{y}^r \left(\prod_{i=1}^n \exp \left[- \left(\frac{\bar{x}_i - \bar{x}_i^r}{\sigma_i^r} \right)^2 \right] \right)}{\sum_{r=1}^N \left(\prod_{i=1}^n \exp \left[- \left(\frac{\bar{x}_i - \bar{x}_i^r}{\sigma_i^r} \right)^2 \right] \right)}. \tag{15}$$

The output of the single Mamdani neuro-fuzzy system, shown in Fig. 1 is defined

$$h_t = \frac{\sum_{r=1}^{N_t} \bar{y}_t^r \cdot \tau_t^r}{\sum_{r=1}^{N_t} \tau_t^r}, \tag{16}$$

where $\tau_t^r = \prod_{i=1}^n (\mu_{A_i^r}(\bar{x}_i))$ is the activity level of the rule $r = 1, \dots, N_t$ of the classifier $t = 1, \dots, T$. Structure depicted by (16) is shown in Fig. 1 (index t is omitted for clarity).

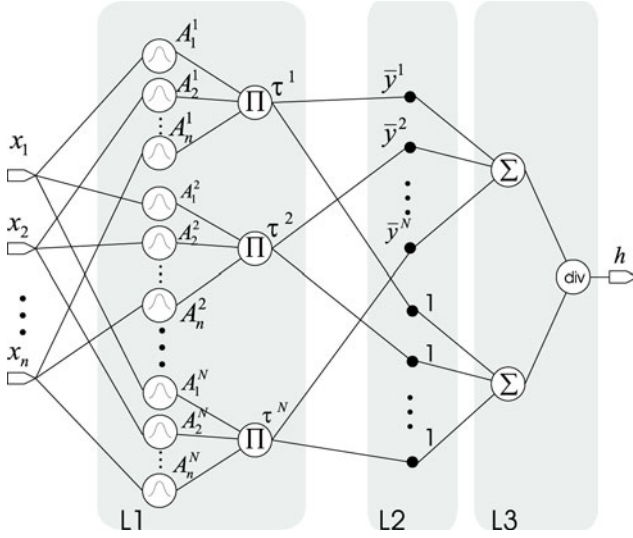


Fig. 1. Single Mamdani neuro-fuzzy system

3 Negative Correlation Learning

Negative correlation learning [5] [6] is a meta-learning algorithm for creating negatively correlated ensembles. Let us denote the l -th learning vector by $\mathbf{z}^l = [x_1^l, \dots, x_n^l, y^l]$, $l = 1 \dots m$ is the number of a vector in the learning sequence, n is the dimension of input vector \mathbf{x}^l , and y^l is the learning class label. The overall output of the ensemble of classifiers is computed by averaging outputs of all hypothesis

$$f(\mathbf{x}) = \frac{1}{T} \sum_{t=1}^T h_t(\mathbf{x}), \quad (17)$$

where $h_t(\mathbf{x})$ is the response of the hypothesis t on the basis of feature vector $\mathbf{x} = [x_1, \dots, x_n]$. All neuro-fuzzy parameters, i.e. antecedent and consequent fuzzy sets parameters, are tuned by the backpropagation algorithm [13]. Having given learning data set of pair (\mathbf{x}^l, y^l) , where y^l is the desired response of the system we can use the following error measure

$$E(\mathbf{x}^l, y^l) = \frac{1}{2} [h_t(\mathbf{x}) - y^l]^2 \quad (18)$$

Every relational neuro-fuzzy system parameter, denoted for simplicity as w , can be determined by minimizing the error measure in the iterative procedure. For every iteration t , the parameter value is computed by

$$w(t+1) = w(t) - \eta \frac{\partial E(\mathbf{x}^l, y^l; t)}{\partial w(t)} \quad (19)$$

where η is a learning coefficient. This is standard gradient learning procedure. As we build an ensemble of negatively correlated neuro-fuzzy systems, the error measure is modified by introducing a penalty term $p_t(l)$ and determining error after whole epoch

$$E_t = \frac{1}{m} \sum_{l=1}^m E_t(l) = \frac{1}{m} \sum_{l=1}^m \frac{1}{2} (h_t(l) - y^l)^2 + \frac{1}{m} \sum_{l=1}^m \lambda p_t(l) \quad (20)$$

where λ is a coefficient responsible for the strength of decorrelation. The penalty term is defined

$$p_t(l) = (h_t(l) - f(\mathbf{x})) \sum_{k \neq l} (h_t(k) - f(\mathbf{x})) \quad (21)$$

NCL metalearning tries to keep responses of the member neuro-fuzzy systems as different as possible, retaining at the same time classification accuracy.

4 Numerical Simulations

We used two well known dataset taken from [11] to show the ability of the proposed systems to fit to data. The systems were initialized randomly by the fuzzy c -means clustering and then trained the backpropagation algorithm.

4.1 Wisconsin Breast Cancer Database

Wisconsin Breast Cancer Database [11] consists of 699 instances of binary classes (benign or malignant type of cancer). Classification is based on 9 features (clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, mitoses). From the data set 205 instances were taken into testing data and 16 instances with missing features were removed. The classification accuracy is 95.3%.

4.2 Glass Identification

In this section we test the efficiency of our methods on the Glass Identification problem [11]. The goal is to classify 214 instances of glass into window and non-window glass basing on 9 numeric features. We took out 43 instances for a testing set. We obtained 94.23% classification accuracy. We used 3 neuro-fuzzy classifiers.

4.3 Ionosphere

Ionosphere problem [11] consists of radar data collected by a system in Goose Bay, Labrador. System of 16 high-frequency antennas targeted on free electrons in the ionosphere returned either evidence or no evidence of some type of structure in the ionosphere. Each object is described by 34 continuous attributes and belongs to one of two classes. The data were divided randomly into 246 learning and 105 testing instances. We used 4 neuro-fuzzy classifiers. We obtained 94.03% classification accuracy.

5 Conclusions

In the paper, we presented an application of the negative correlation learning [5][6], which is a meta-learning algorithm for creating negatively correlated ensembles, to an ensemble of Mamdani-type neuro-fuzzy systems. Thanks to neuro-fuzzy systems, apart from purely data-driven designing, we can use some expert knowledge in the form of fuzzy rules. All system parameters were determined by the backpropagation algorithm. NCL penalty factor influenced the backpropagation learning by inhibiting fuzzy system parameters modification if the system was correlated with the ensemble. Simulations on some popular benchmarks show great accuracy of the proposed method.

Acknowledgments

This work was partly supported by the Polish Ministry of Science and Higher Education (Habilitation Project 2007-2010 Nr N N516 1155 33, Polish-Singapore Research Project 2008-2010 and Research Project 2008-2011) and the Foundation for Polish Science – TEAM project 2010-2014.

References

1. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine (2007), <http://www.ics.uci.edu/~mllearn/MLRepository.html>
2. Babuska, R.: Fuzzy Modeling For Control. Kluwer Academic Press, Boston (1998)
3. Monirul Islam, M., Yao, X.: Evolving Artificial Neural Network Ensembles. *IEEE Computational Intelligence Magazine*, 31–42 (February 2008)
4. Jang, R.J.-S., Sun, C.-T., Mizutani, E.: Neuro-Fuzzy and Soft Computing. In: A Computational Approach to Learning and Machine Intelligence. Prentice Hall, Upper Saddle River (1997)
5. Liu, Y., Yao, X.: Ensemble learning via negative correlation. *Neural Networks* 12, 1399–1404 (1999)
6. Liu, Y., Yao, X.: Simultaneous training of negatively correlated neural networks in an ensemble. *IEEE Trans. Syst., Man, Cybern. B* 29, 716–725 (1999)
7. Nauck, D., Klawon, F., Kruse, R.: Foundations of Neuro - Fuzzy Systems. John Wiley, Chichester (1997)
8. Nauck, D., Kruse, R.: How the Learning of Rule Weights Affects the Interpretability of Fuzzy Systems. In: Proceedings of 1998 IEEE World Congress on Computational Intelligence, FUZZ-IEEE, Alaska, pp. 1235–1240 (1998)
9. Nowicki, R.: Nonlinear modelling and classification based on the MICOOG defuzzification. *Nonlinear Analysis* 71, e1033–e1047 (2009)
10. Pedrycz, W.: Fuzzy Control and Fuzzy Systems. Research Studies Press, London (1989)
11. Pedrycz, W., Gomide, F.: An Introduction to Fuzzy Sets, Analysis and Design. The MIT Press, Cambridge (1998)
12. Rutkowski, L.: Flexible Neuro Fuzzy Systems. Kluwer Academic Publishers, Dordrecht (2004)
13. Wang, L.-X.: Adaptive Fuzzy Systems And Control. PTR Prentice Hall, Englewood Cliffs (1994)

A New Fuzzy Approach to Ordinary Differential Equations

Witold Kosiński¹, Kurt Frischmuth², and Dorota Wilczyńska-Sztyma³

¹ Department of Computer Science, Polish-Japanese Institute of Information Technology, ul. Koszykowa 86, 02-008 Warsaw, Poland

wkos@pjwstk.edu.pl

² Institute of Mathematics, Rostock University, 18051 Rostock, Germany
kurt.frischmuth@uni-rostock.de

³ Institute of Mechanics and Applied Computer Science, Kazimierz Wielki University, ul. Chodkiewicza 30, 85-072 Bydgoszcz, Poland
dorotaws@ukw.edu.pl

Abstract. In real-life problems, both parameters and data used in mathematical modeling are often vague or uncertain. In fields like system biology, diagnosis, image analysis, fault detection and many others, fuzzy differential equations and stochastic differential equations are an alternative to classical, or in the present context crisp, differential equations. The aim of the paper is to propose a new formulation of fuzzy ordinary differential equations for which the Hukuhara derivative is not needed. After a short review of recent results in the theory of ordered fuzzy numbers, an exemplary application to a dynamical system in mechanics is presented. Departing from the classical framework of dynamics, equations describing the fuzzy representation of the state are introduced and solved numerically for chosen fuzzy parameters and initial data.

1 Introduction

The study of fuzzy differential equations (FDEs) forms a suitable setting for mathematical modeling in the presence of vagueness, typical of many real world problems. The theory of FDEs has been mainly developed in the framework of functions mapping closed intervals into the metric space of convex fuzzy numbers [21,2] equipped with the Hausdorff metric and possessing the Hukuhara derivative [6] at each point [7,20]. Recently an alternative framework has been proposed, which leads to ordinary but multivalued differential equations, i.e. to differential inclusions [20]. A general reference to the classical approach of fuzzy differential equations is the book [20] and references therein. The necessity of using the complex framework of differential inclusions follows from the well known fact that the metric space of convex fuzzy numbers with the algebraic operations addition and scalar nonnegative multiplication is only a semi-linear metric space. This space can be embedded as a complete cone into a corresponding Banach space [2,5]. The lack of full scalar multiplication, i.e. positive and negative, is one of the main drawbacks of the theory of convex fuzzy numbers [5].

The main aim of the present paper is to propose a new formulation of fuzzy ordinary differential equations for which the Hukuhara derivative is not needed.

Fuzzy differential equations have been formulated within the concept of convex fuzzy numbers, that has been introduced by Nguyen [21] in order to improve calculation and implementation properties of fuzzy numbers. However, the results of multiple arithmetic operations on convex fuzzy numbers lead regularly to an unacceptable large increase of the fuzziness. Furthermore, the order of operations affects results since the distributive law does not for the product of sums. These are two classical ways to define arithmetic operations on fuzzy numbers, the extension principle of Zadeh [25] and the α -cut and interval arithmetic method (cf. [1]). As long as we work with fuzzy numbers featuring continuous membership functions, both approaches give the same results. Generalizations have been discussed by Sanchez [22] and Klir [8], as well as by Drewniak [3] and Wagenknecht [23,24]. Even in the very particular case of fuzzy numbers, which are called (L, R) -numbers [4], approximations of fuzzy operations and functions are needed in order to combine extension principle and algebraic closedness. These approximations, again, lead to some drawbacks [24].

In this paper we try a new approach to differential equations in the presence of vagueness and uncertainties. To this end we recall recent concepts related to the arithmetics of so-called ordered fuzzy numbers. The latter have become an efficient tool in dealing with unprecise, fuzzy quantitative terms. Next, defuzzification operators on ordered fuzzy numbers will be defined. In a final section an exemplary application to a dynamical system in mechanics is presented. Departing from the classical framework of dynamics, equations describing the fuzzy representation of the state are introduced and solved numerically for chosen fuzzy parameters and initial data.

2 Ordered Fuzzy Numbers

In our first attempt at a new model of fuzzy numbers we have made the observation in [19]: a kind of quasi-invertibility of membership functions is crucial and one has to define arithmetic operations on their inverse parts to be in agreement with operations on the crisp real numbers. Consequently, assuming this, the invertibility of membership functions of convex fuzzy number A makes it possible to define two functions a_1, a_2 on $[0, 1]$ that give lower and upper bounds of each α -cut of the membership function μ_A of the number A

$$A[\alpha] = \{x : \mu_A(x) \geq \alpha\} = [a_1(\alpha), a_2(\alpha)] \quad (1)$$

where boundary points are given for each α by $a_1(\alpha) = \mu_A|_{incr}^{-1}(\alpha)$ and $a_2(\alpha) = \mu_A|_{decr}^{-1}(\alpha)$ where the subscripts $|_{incr}$ and $|_{decr}$ denote the restrictions of the function μ_A to its subdomains on which is increasing or decreasing, respectively. In the series of papers [16,17,9,18] we have introduced and then developed main concepts of the space of ordered fuzzy numbers (OFNs). In our approach the concept of membership functions has been weakened by requiring a mere *membership relation*.

Definition 1. *By an ordered fuzzy number A we mean an ordered pair (f, g) of functions such that $f, g : [0, 1] \rightarrow \mathbb{R}$ are continuous.*

Notice that f and g do not need to be inverse functions of some membership functions. On the other hand, if an ordered fuzzy number (f, g) is given in which f is an increasing function and g – decreasing, both on the unit interval I , and such that $f \leq g$, then one can attach to this pair a continuous function μ with a convex graph. This function μ can be regarded as a membership function of a convex fuzzy number with an extra arrow, which denotes the orientation of the number. The assignment can be defined by an inverse formula, namely $f^{-1} = \mu|_{incr}$ and $g^{-1} = \mu|_{decr}$.

Notice that pairs (f, g) and (g, f) represent two different ordered fuzzy numbers (OFNs), unless $f = g$. They differ in their orientations.

Definition 2. *Let $A = (f_A, g_A), B = (f_B, g_B)$ and $C = (f_C, g_C)$ be ordered fuzzy numbers. The scalar multiplication $r \cdot A$ by real $r \in \mathbb{R}$ is defined in a natural way: $r \cdot A = (rf_A, rg_A)$, sum $C = A + B$, subtraction $C = A - B$, product $C = A \cdot B$, and division $C = A \div B$ are defined by the formula*

$$f_C(y) = f_A(y) \star f_B(y) \quad g_A(y) \star g_B(y) \tag{2}$$

where “ \star ” works for “+”, “−”, “ \cdot ”, and “ \div ”, respectively, and where $A \div B$ is defined, if the functions $|f_B|$ and $|g_B|$ are bigger than zero. Notice that in order to subtract B from A we have to add to A the number $(-1) \cdot B$, and consequently $B - B = 0$, where $0 \in \mathbb{R}$ is the crisp zero. This means that subtraction of OFNs is not compatible with the Zadeh extension principle, if we confine OFNs to convex fuzzy numbers. However, according to Definition 2, adding ordered fuzzy numbers with given membership functions¹ of the same orientation, gives the same result as the standard addition on convex fuzzy numbers.

Operations introduced in the space \mathcal{R} of all ordered fuzzy numbers (OFN) make it a linear space (and an algebra), which can be equipped with a sup norm $\|A\| = \max(\sup_{s \in I} |f_A(s)|, \sup_{s \in I} |g_A(s)|)$ if $A = (f_A, g_A)$. In \mathcal{R} any algebraic equation $A + X = C$ for X , with arbitrarily given fuzzy numbers A and C , can be solved. Moreover, \mathcal{R} becomes a Banach space², isomorphic to a cartesian product of $C(0, 1)$ – the space of continuous functions on $[0, 1]$. Continuous, linear functionals on \mathcal{R} give a class of defuzzification functionals. Each of them, say ϕ , has a representation by a sum of two Stieltjes integrals with respect to functions h_1 and h_2 of bounded variation,

$$\phi(f, g) = \int_0^1 f(s)dh_1(s) + \int_0^1 g(s)dh_2(s) . \tag{3}$$

¹ Such numbers are convex fuzzy numbers equipped, however, additionally with orientation.

² In [5] the authors for the first time introduced a linear structure to convex fuzzy numbers.

Notice that if we put $h_1(s) = h_2(s) = 1/2H(s)$ with $H(s)$ as the Heaviside function with the unit jump at $s = 1$ then the defuzzification functional in (3) will represent the classical MOM – middle of maximum

$$\phi(f, g) = 1/2(f(1) + g(1)) . \tag{4}$$

The new model gives a continuum of defuzzification operators, both linear and nonlinear, which map ordered fuzzy numbers into reals. An example of a nonlinear functional is center of gravity defuzzification functional (COG) calculated at (f, g)

$$\bar{\phi}(f, g) = \int_0^1 \frac{f(s) + g(s)}{2} [f(s) - g(s)] ds \left\{ \int_0^1 [f(s) - g(s)] ds \right\}^{-1} \tag{5}$$

provided $\int_0^1 [f(s) - g(s)] ds \neq 0$. Other defuzzification operators can be defined, for example the method of defuzzification by the geometrical mean defined by

$$\phi_{GM}(f, g) = \frac{g(1)g(0) - f(0)f(1)}{g(1) + g(0) - (f(0) + f(1))} . \tag{6}$$

It is worthwhile to point out that the class of ordered fuzzy numbers (OFNs) represents the whole class of convex fuzzy numbers with continuous membership functions. Some generalization of the present Def. 1 to include discontinuous functions has been proposed in [10], and interpretations are given in [13].

3 Dynamic System Equations

The new model provides a challenge to modeling economical as well as physical situations [10,11], since calculation on elements from the space \mathcal{R} is most natural and compatible with calculations on real (crisp) numbers. Now, we will use small letters for elements of \mathcal{R} , and for example a pair of functions, so instead of writing $A = (f, g) \in \mathcal{R}$ we will write $a = (a_{up}, a_{down}) \in \mathcal{R}$ (compare the denotation from [16,17,9]).

Let us consider an n -d dynamical system described by the ODE in \mathbb{R}^n

$$\dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{x}, t) , \text{ with } \mathbf{x}(0) = \mathbf{x}_0 . \tag{7}$$

This equation has an immediate counterpart in the space \mathcal{R} if the variable $\mathbf{x} = (x^1, \dots, x^n)$ from \mathbb{R}^n is substituted by the variable $(\mathbf{x}_{up}, \mathbf{x}_{down})$ from \mathcal{R}^n . Equation (7) forms an evolution equations in two n -fold products of the Banach space $C([0, 1])$:

$$\dot{\mathbf{x}}_{up} = \mathbf{F}(\mathbf{x}_{up}, t), \quad \dot{\mathbf{x}}_{down} = \mathbf{F}(\mathbf{x}_{down}, t) \tag{8}$$

with initial conditions $\mathbf{x}_{up}(0) = \mathbf{x}_{up0}(s), \mathbf{x}_{down}(0) = \mathbf{x}_{down0}(s), s \in [0, 1]$.

Remark 1. Solutions to (8) are functions of two variables since the initial conditions are already functions of $s \in [0, 1]$.

For example if

$$\mathbf{x}_{up} = (x_{up}^1, \dots, x_{up}^n), \quad \mathbf{x}_{down} = (x_{down}^1, \dots, x_{down}^n),$$

and the function $F(\mathbf{x})$ is linear, equal to $A\mathbf{x}$, with $n \times n$ matrix A , then the fuzzy dynamical system is given by

$$\dot{\mathbf{x}}_{up} = A\mathbf{x}_{up}, \quad \dot{\mathbf{x}}_{down} = A\mathbf{x}_{down} \tag{9}$$

with initial conditions $\mathbf{x}_{up}(0) = \mathbf{x}_{up0}(s)$, and $\mathbf{x}_{down}(0) = \mathbf{x}_{down0}(s)$ when $s \in [0, 1]$. Solutions to (9) with $t \geq 0, s \in [0, 1]$ are

$$\mathbf{x}_{up}(t, s) = \mathbf{x}_{up0}(s) \exp(\mathbf{A}t), \quad \mathbf{x}_{down}(t, s) = \mathbf{x}_{down0}(s) \exp(\mathbf{A}t). \tag{10}$$

Remark 2. Solution $\mathbf{x} = (\mathbf{x}_{up}, \mathbf{x}_{down})$ is a vector function of the variable t with values being n ordered fuzzy numbers. In order to obtain a real valued vector function we need to superpose a defuzzification functional.

Assume that our defuzzification functional is linear, i.e. of the form (3) and calculate its value at the solution (10). Then we get, for any $t \geq 0$,

$$\phi(\mathbf{x})(t) = \int_0^1 \mathbf{x}_{up}(t, s) dh_1(s) + \int_0^1 \mathbf{x}_{down}(t, s) dh_2(s). \tag{11}$$

Hence we get $\hat{\mathbf{x}}(t) := \phi(\mathbf{x}) = \phi(\mathbf{x}_0) \exp(\mathbf{A}t)$.

Remark 3. If a linear dynamical system has fuzzy initial conditions then any linear defuzzification of its solution is the solution of the system with the defuzzified initial condition.

Here we have listed main properties of fuzzy ODEs and defuzzification in the space \mathcal{R} . One can go further and use this approach in modeling deformable mechanical systems with fuzziness.

Consider the 1d case, then the matrix A will transform to a fuzzy number $a = (a_{up}, a_{down}) \in \mathcal{R}$, and (9) will simplify to

$$\dot{x}_{up} = x_{up}a_{up}, \quad \dot{x}_{down} = x_{down}a_{down}, \tag{12}$$

with solutions $x_{up}(t, s) = x_{0up}(s) \exp(a_{up}(s)t), x_{down}(t, s) = x_{0down}(s) \exp(a_{down}(s)t)$, and $x_0 = (x_{0up}, x_{0down})$ as initial conditions. If we perform the defuzzification of the solution then, even in the linear case, we get

$$\hat{x}(t) = \int_0^1 x_{0up}(s) \exp(a_{up}(s)t) dh(s) + \int_0^1 x_{0down}(s) \exp(a_{down}(s)t) dh_2(s) \tag{13}$$

which shows that the result $\hat{x}(t) = \phi(x_{up}, x_{down})$ is a nonlinear function (in general, operator) of the defuzzified coefficient a .

Sample Problem: Nonlinear Pendulum

Consider a pendulum governed by the system of equations

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = -\frac{g}{\ell} \sin x_1, \quad (14)$$

where g is gravity, x_1 – the position (inclination) of the pendulum, the velocity $v = \dot{x}_1 = x_2$, and we set the length $\ell \in \mathbb{R}$ of the pendulum equal to $\ell = 0.1$. Now, we regard the parameter $p = \ell$ as a fuzzy number. We know it is more than 0.08 and less than 0.12. What can we say now about the solution? In Fig. 1 we present a fan of solution curves starting from the equilibrium position at an initial velocity $\dot{x}_1 = 1$. The parameter of the pencil $p = \ell$ is the length of the pendulum.

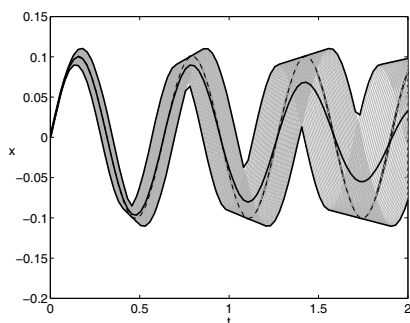


Fig. 1. Pendulum. Fan of solutions for $x(0) = 0$, $\dot{x}(0) = 1.0$, $\ell \in [0.08, 0.12]$.

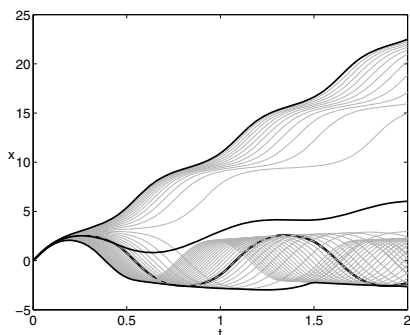


Fig. 2. Pendulum. Fan of solutions for $x(0) = 0$, $\dot{x}(0) = 19.0$, $\ell \in [0.08, 0.12]$.

Now we defuzzify the obtained solution and assume the fuzzy length $p = \ell$ in terms of two different OFN being convex fuzzy numbers, triangular and quadratic. Of the two standard defuzzification functionals MOM (4), and COG (5) we apply here the COG variant.

The triangular membership function $\mu_T(p)$ is described by the ordered fuzzy number $(f_T(s), g_T(s))$, with $s \in [0, 1]$ and $p \in [0.08; 0.12]$, is $f_T(s) = \frac{s+4}{50}$, $g_T(s) = \frac{6-s}{50}$, and $\mu_T(p) = 50p - 4$, $p \in [0.08; 0.1]$, $\mu_T(p) = -50p + 6$, $p \in [0.1; 0.12]$.

The results of applying the COG functional to fuzzy solutions of (14) with the pendulum length as fuzzy parameter are presented on Figs 1 and 2 by a continuous line, while the classical (crisp, with $\ell = 0.1$) trajectory is given by a dashed line.

4 Conclusions

The approach to ordinary differential equations presented here takes into account fuzzy initial conditions or fuzzy coefficients. The use of ordered fuzzy numbers leads to fuzzy differential equations, which become functional differential equations in the Banach space of real-valued continuous functions. Two defuzzification functionals are studied, the classical mean of maxima (MOM) and the OFN counterpart of the center of gravity (COG) operators, which are also known as Mamdani controllers. Application to fuzzy solutions to the nonlinear pendulum problem gives us numerical results. The first defuzzification functional does not require to pass to functional differential equations. It is enough to solve initial value problems for ODEs with two different values of ordered fuzzy numbers, representing the fuzzy length of the pendulum, namely for $\ell_1 = f(1)$ and $\ell_2 = g(1)$. In the case of the geometrical mean (6) only four different values are needed. The use of COG requires, however, the knowledge of the solution to (14) for the whole range of ℓ . Going further, we can use this approach in modeling deformable mechanical systems, governed in the crisp case by PDEs, which have to be equipped with fuzzy material coefficients or fuzzy initial or boundary conditions in the the case of vague data. This is the subject of a separate paper [12].

References

1. Buckley James, J., Eslami, E.: An Introduction to Fuzzy Logic and Fuzzy Sets. Physica-Verlag, Springer, Heidelberg (2005)
2. Diamond, P., Kloeden, P.: Metric Spaces of Fuzzy Sets. World Scientific, Singapore (1993)
3. Drewniak, J.: Fuzzy numbers (in Polish). In: Chojcan, J., Łęski, J. (eds.) Zbiory rozmyte i ich zastosowania, Wydawnictwo Politechniki Śląskiej, Gliwice, pp. 103–129 (2001)
4. Dubois, D., Prade, H.: Operations on fuzzy numbers. Int. J. System Science 9, 576–578 (1978)
5. Goetschel Jr., R., Voxman, W.: Elementary fuzzy calculus. Fuzzy Sets and Systems 18, 31–43 (1986)
6. Hukuhara, M.: Intégration des applications mesurables dont la valeur est un compact convexe (in French). Funkcial. Ekvac. 10, 205–223 (1967)
7. Kaleva, O.: Fuzzy differential equations. Fuzzy Sets and Systems 24, 301–317 (1987)

8. Klir, G.J.: Fuzzy arithmetic with requisite constraints. *Fuzzy Sets and Systems* 91, 165–175 (1997)
9. Kosiński, W.: On defuzzification of ordered fuzzy numbers. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) *ICAISC 2004. LNCS (LNAI)*, vol. 3070, pp. 326–331. Springer, Heidelberg (2004)
10. Kosiński, W.: On fuzzy number calculus. *Int. J. Appl. Math. Comput. Sci.* 16(1), 51–57 (2006)
11. Kosiński, W.: On soft computing and modelling. *Image Processing Communications* 11(1), 71–82 (2006)
12. Kosiński, W., Frischmuth, K., Piasecki, W.: Fuzzy approach to hyperbolic heat conduction equations. In: Kuczma, M., Wilmański, K., Szajna, W. (eds.) *18th Intern. Confer. on Computational Methods in Mechanics, CMM 2009, Short Papers*, Zielona Góra, May 2009, pp. 249–250 (2009)
13. Kosiński, W., Prokopowicz, P., Kacprzak, D.: Fuzziness - representation of dynamic changes by ordered fuzzy numbers. In: Seising, R. (ed.) *Views of Fuzzy Sets and Systems from Different Perspectives. Studies in Fuzziness and Soft Computing*, vol. (243), pp. 485–508. Springer, Heidelberg (2009)
14. Kosiński, W., Prokopowicz, P., Ślęzak, D.: Fuzzy numbers with algebraic operations: algorithmic approach. In: Kłopotek, M., Wierzchoń, S.T., Michalewicz, M. (eds.) *Intelligent Information Systems 2002, Proc. IIS 2002*, Sopot, Poland, June 3–6, pp. 311–320. Physica Verlag, Heidelberg (2002)
15. Kosiński, W., Prokopowicz, P., Ślęzak, D.: On algebraic operations on fuzzy reals. In: Rutkowski, L., Kacprzyk, J. (eds.) *Advances in Soft Computing, Proc. of the Sixth Int. Conference on Neural Network and Soft Computing, Zakopane, Poland*, June 11–15 (2002), pp. 54–61. Physica-Verlag, Heidelberg (2003)
16. Kosiński, W., Prokopowicz, P., Ślęzak, D.: Ordered fuzzy numbers. *Bulletin of the Polish Academy of Sciences, Ser. Sci. Math.* 51(3), 327–338 (2003)
17. Kosiński, W., Prokopowicz, P.: Algebra of fuzzy numbers (in Polish). *Matematyka Stosowana* 5(46), 37–63 (2004)
18. Kosiński, W., Prokopowicz, P., Ślęzak, D.: Calculus with fuzzy numbers. In: Bolc, L., Michalewicz, Z., Nishida, T. (eds.) *IMTCI 2004. LNCS (LNAI)*, vol. 3490, pp. 21–28. Springer, Heidelberg (2005)
19. Kosiński, W., Słysz, P.: Fuzzy reals and their quotient space with algebraic operations. *Bull. Pol. Acad. Sci., Sér. Techn. Scien.* 41(30), 285–295 (1993)
20. Lakshmikantham, V., Mohapatra, R.N.: *Theory of Fuzzy Differential Equations and Inclusions*. Taylor and Francis, Abington (2003)
21. Nguyen, H.T.: A note on the extension principle for fuzzy sets. *J. Math. Anal. Appl.* 64, 369–380 (1978)
22. Sanchez, E.: Solutions of fuzzy equations with extended operations. *Fuzzy Sets and Systems* 12, 237–248 (1984)
23. Wagenknecht, M.: On the approximate treatment of fuzzy arithmetics by inclusion, linear regression and information content estimation. In: Chojcan, J., Łęski, J. (eds.) *Zbiory rozmyte i ich zastosowania*, Wydawnictwo Politechniki Śląskiej, Gliwice, pp. 291–310 (2001)
24. Wagenknecht, M., Hampel, R., Schneider, V.: Computational aspects of fuzzy arithmetic based on archimedean t -norms. *Fuzzy Sets and Systems* 123(1), 49–62 (2001)
25. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning, Part I. *Information Sciences* 8, 199–249 (1975)

K2F – A Novel Framework for Converting Fuzzy Cognitive Maps into Rule-Based Fuzzy Inference Systems

Lars Krüger*

Institute of Applied Microelectronics and Computer Engineering,
Rostock University, D-18119 Rostock, Germany
l.krueger@technik.uni-rostock
<http://www.imd.uni-rostock.de>

Abstract. This paper focuses on a novel methodological framework for converting a *Fuzzy Cognitive Map* into a network of rule-based *Fuzzy Inference Systems*. Furthermore, it allows to obtain a crisp value representing an arbitrary parameter of the complex system's model. This way the system provides a quantitative answer without employing an exact mathematical model. This paper also outlines a first possible application area: the valuation of investments in high-technology ventures. A field in which usually conventional quantitative and retrospective measures usually do not deliver satisfying results due to the complexity of future-oriented risk prognosis and the lack of quantitative data.

Keywords: Fuzzy Cognitive Map, Fuzzy Inference System, Fuzzy Rules.

1 Introduction

Complex real-world systems such as Politics, Economics or Finance consist of several interrelated concepts and actors. Initially, *Cognitive Maps* [12], [13] were a first to attempt to intuitively represent those complex systems. This approach was one of the first to rely on natural language information expressed by experts rather than on mathematical models that may be difficult or even impossible to solve. But, real-world relations are not binary, they are fuzzy. Hence, a model that tries to describe real-world systems adequately has to incorporate fuzziness. A subsequent development was proposed by Kosko: *Fuzzy Cognitive Maps* (FCM) [10], [11]. Although describing complex systems and their behavior, FCMs still can not be used to quantitatively analyze a complex system. Though, a rule-based *Fuzzy Inference System* (FIS) could achieve this, but its efficacy mainly depends on the 'intelligence' of the integrated rule base [9]. So, it is obvious to use the expert knowledge captured in an FCM to design a rule-base. But, until now only a few ways of how to do that has been proposed. Each of these

* The author wishes to thank *engage - Key Technology Ventures* for material and intellectual support.

approaches, however, suffers from certain shortcomings that hinder a practical and unambiguous conversion.

This paper proposes a novel methodological framework for the transformation of an FCM to a system of rule-based FISs. As a result, expert knowledge captured in an FCM can unambiguously be utilized to quantitatively analyze a complex system. Section 2 gives a brief overview over the general concept of FCMs and their limitations. Subsequently, Section 3 provides an introduction to the novel methodological framework. First preliminary results of experiments based on implementation of the framework are reported in Section 4. Finally, Section 5 concludes this paper with some remarks on current challenges and needs for further research.

2 State-of-the-Art

FCMs are fuzzy graph structures for representing conceptual pictures of interconnected complex systems. They work by capturing and representing cause-and-effect relationships. They are a combination of neuronal networks and fuzzy logic. FCMs consist of various nodes (concepts) connected by directed graphs (edges) with feedback. These connections stand for the causal relationship between the nodes. There is a causal relation between two given concepts whenever a relative variation in one of these concepts causes a relative variation on the other. The result of a causal effect is always a variation in one or more concepts/nodes. Therefore, FCMs show the variation of a concept's value, not the concept's absolute value. Each node has a fuzzy value ranging from $[-1, 1]$. Each edge is associated to a fuzzy weight $w \in [-1, 1]$. A positive weight represents a causal increase; a negative weight indicates a causal decrease.

The capability to capture the characteristics and the behavior of complex systems has led to two general application forms: *Knowledge Mapping* (KM) and *Decision Support Systems* (DSS) [2]. These applications, however, suffer from a major restriction: the representation of systems that use only simple monotonic and symmetric causal relations between concepts. But many real world causal relations are neither symmetric nor monotonic. A fuzzy-rule-based approach provides a far more adequate way to model real-world causal relationships. A *Fuzzy Inference System* (FIS) makes use of fuzzy rules in which real-world coherences are tied together. In practice, these rules are called rule-of-thumb or heuristics. They reflect an expert's action to control or an observation of a system. Identifying and formulating rules that fully describe the behavior of complex systems is the key issue in *Fuzzy Engineering*. But at the same time, it is known to be its natural bottleneck [11].

2.1 Limitations

Until now only a few approaches has utilized the knowledge captured in an FCM to design and build a rule-based FIS. According to Eloff et al. [8], [3] an FCM is an ideal starting point to derive a rule base from. This "practical" approach,

however, leaves the user in the lurch when it comes to typical challenges such as multiple causal relations and non-linear causal behavior.

A more sophisticated approach has been proposed by Carvalho and Tomé: *Rule-Based Fuzzy Cognitive Maps* (RB-FCM) [4], [6], [5], [7]. Such an FCM consists of fuzzy nodes and a rule base which relates nodes to each other. Each concept contains several membership functions which represent the concepts' possible values or the possible change of its values. To adequately model non-linearity and non-symmetric opposition, the authors introduce two novel concepts: *Fuzzy Causal Relation* and *Fuzzy Carry Accumulation*. In comparison to a classic FCM, the major advantages of the RB-FCM are more flexible modeling of causal relations and improved stability regarding the application for scenario simulation purposes. Though, this approach has also some deficits. It is rather complex, its computation is very time-consuming, and it does not explicitly address the issue of how to cope with complex multiple causal relations.

A slightly different model has been proposed by Khan and Khor [2], [1]. In contrast to the RB-FCM in which the nodes' states are seen as additive and cumulative so that the state values can be 'carried over' when they exceed a maximum, the authors suggest that each concept has a maximum and a minimum limit. This limit is expressed in the form of a weight vector such that the total of the causality is within the interval $[0, 1]$. By proposing the aggregation operator $A : (c_1, \dots, c_i) = \sum_{i=1}^n w_i d_i$, their approach explicitly addresses the issue of causalities in the multiple input case. But, at the same time, they neglect to address the issue of how to derive a rule base from an existing expert FCM.

2.2 Aim of This Paper

The aim of this paper is to introduce a novel framework to derive a rule-based FIS from an expert FCM: *Knowledge-to-Fuzzy* (K2F). Complex analyzes and time-consuming computation shall be prevented. In contrast to all preceding approaches, this paper makes a suggestion how to calculate a crisp value out of an FCM. For this purpose, a first application in the field of valuation of investments in high-technology ventures is briefly highlighted.

3 K2F – The Novel Framework

3.1 The Interpretation of Nodes and Edges

The major feature of the novel framework is that it does not employ a central rule base that represents the causal relations between the nodes. By contrast, it assumes that each causal link can be represented by a standardized rule base. A causal link of the causal strength w_{ij} between two nodes C_i and C_j depicted by an arrow in an FCM is represented by a rule-based FIS FIS_{ij} (Figure 1). Hence, each causal link in an FCM can be translated into a corresponding single-input-single-output FIS. By applying this procedure, every highly-individual FCM can be translated into a network of FISs.

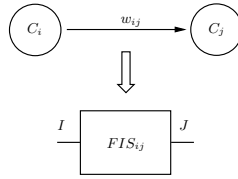


Fig. 1. Representation of a Single Causal Link by an FIS

3.2 Multiple Causal Inputs

To cope with multiple causal inputs, a method is proposed that is based on the aggregation operator A proposed by Khan and Khor [2], [1]. Initially, all incoming edges are transformed to individual FISs according to the causal weight associated to them. Afterwards, the results of these FISs are combined using the aggregation operation $A = 1/n \sum_{i=1}^n c_i$. Thereby, the incoming signals can be weighted according to their (assumed) importance (Figure 2). Usually, the signals are all treated equally. The outgoing signal represents the combined causal effect of the preceding cause nodes on the following effect node.

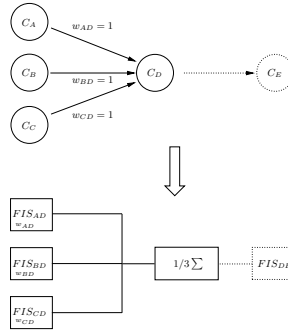


Fig. 2. Interpretation of Multiple Causal Inputs

3.3 Rule-Based Causality

Each of the causalities is represented by a specific rule-based FIS. The type of causality (positive or negative) as well as its strength is expressed by a specific rule base. The rule base is made for a single-input-single-output FIS. This decreases the calculation power needed to calculate a whole network. Moreover, it reduces the risk of rule explosion.

Variable Determination. The input and output parameters of the FIS correspond to the level of activation of the actual node. They are represented by I , whereas J denotes the output parameter level-of-activation of the effected node. Both parameters are defined on the basic set $X_{I,J} = [-1; 1] \in R$. An input signal $I = 1.0$ denotes the highest activation of the actual concept. It

can correspond to the highest growth of the underlying concept. In contrast to this, an input value $I = -1.0$ expresses the highest activation of the node in the negative sense, it corresponds to the largest decline of the actual concept. The parameters I and J can take 21 parameter values each. Every step at 0.1 is represented by a parameter value. The output signal J equals the value of the concept node C_j and is thus potentially the input signal I of a following FIS. Each parameter value is mapped onto a fuzzy set. The corresponding membership function is bell-shaped or *Gaussian*. This form of membership functions allows for a more realistic modeling of the underlying concept values and their corresponding membership degrees.

Definition of Rules. Next, the causal linkage between the two parameters has to be defined. To model (non-linear and non-symmetric) causality a specified rule base is designed. Each causality, and hence each causal strength is represented by a specific rule base. The *IF*-part of the rule consists of only one input I . This signal equals the fuzzy number of the concept node value C_i . Due to the discretization in steps of 0.1, the rule base is limited to a reasonable number of rules. Correspondingly, the *THEN*-part of the rule consists of only one defuzzified output signal J . To determine the individual rule base, the general understanding of the causal relationship between input and output parameter has to be highlighted. A positive causal relationship between C_i and C_j means that if C_i increases (less/ much) then C_j also increases (less/ much). A positive causal relation that has the causal strength $w_{ji} < 1$ leads, thus, to a weakened increase of C_j in case C_i increases. This weakening effect grows the lower the causal strength is. For example, a concept value of $C_i = 1.0$ may lead only to a concept value $C_j = 0.2$. The same holds true for a decline when there is a positive causal relation. An exemplary rule bases representing positive causal strength $w = 1.0$ and the corresponding output profile is shown in Figure 3. If there is even less than a strong decline ($C_i = -0.2$) then there may be no

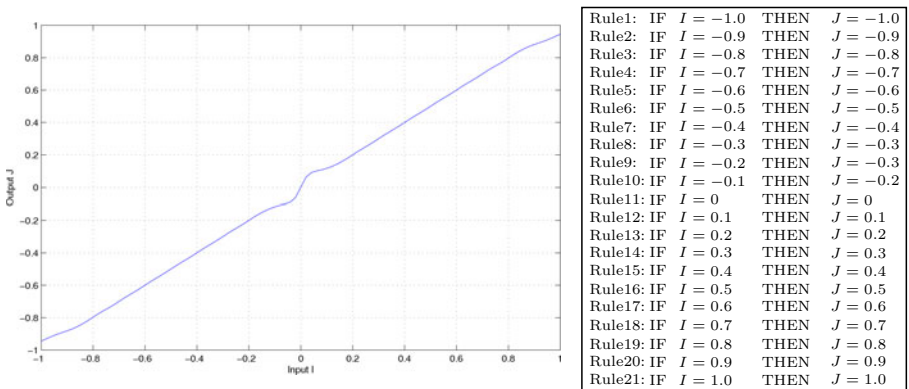


Fig. 3. Rule Base and Output Profile for Causality $w_{ij} = 1.0$

change in the effect concept ($C_j = 0$) at all, if there is a weak negative causal relationship. Analogously to this example specific rule bases are implemented for all possible causal weights $w, w \in [-1, 1]$. The methodological framework proposed here employs a total of 11 specific rule bases.

4 Example of Application

The aforementioned methodology enables the transformation of an arbitrary FCM into a network of FIS. The mere transformation is, however, not of particular practical utility. The major advantage of this methodology is the capability to obtain a real (crisp) value. This feature goes far beyond the conventional qualitative answer to *What-If*-scenarios. For this reason, a specific component is needed: a 'final' FIS. Its purpose is to take all directly preceding causal links and to calculate a final output value. The input value is obtained by aggregating all incoming causal links (inputs) by using the aggregation operator (X). Its value ranges from +1 (highest increase) to -1 (highest decrease) (Figure 4). The input value is mapped onto the corresponding output value by a specific

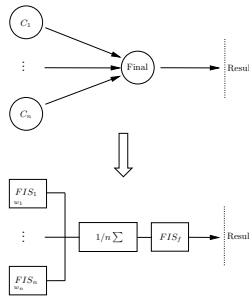


Fig. 4. Aggregation and final FIS - Output of a Crisp Value

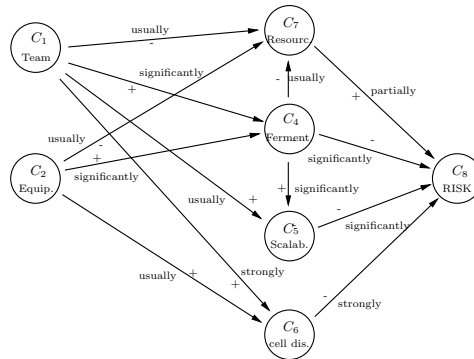


Fig. 5. Exemplary FCM

rule base. These rules and the values of the output parameter naturally depend on the context of the system to be modeled and analyzed.

An specific biotechnology-venture FCM was obtained by expert interviews (Figure 5). It pictures the different, interrelated factors that impact the risk and thus the discount rate used in common venture valuation methods. Using the *K2F*-method this FCM was then transformed into a network of FISs. First experiments have tested the ability of the new framework. Using the numerical simulation software *Simulink*, including the *Fuzzy Logic Toolbox*, this FIS network exhibited stable behavior and yielded a crisp final result. The simulation output matches the empirically proved coherences between individual characteristics of an high-technology venture and an adequate discount rate [14]. The resulting output profile is pictured in Figure 5.

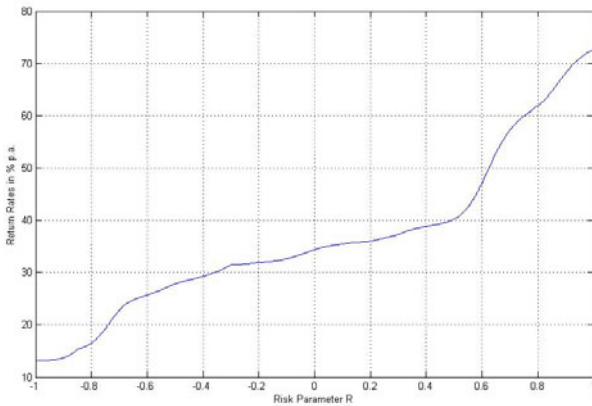


Fig. 6. Simulation Results - Discount Rate r

5 Conclusions and Outlook

This first attempt has revealed the framework's suitability to model complex systems and to eventually calculate a crisp return value. This could be applied in real-world for example for assessing a risk-adequate return rate for venture capital investments on the basis of the venture's individual characteristics. Besides these promising first results, some issues have to be addressed. First of all, the herein proposed rule bases representing causality could possibly be optimized regarding the number and the characteristic of the rules. Possible results may be a different output profile and, hence, a different modeling of causality. Furthermore, the integration of all preceding causal links into one auxiliary input value for the finale FIS could be discussed. This holds also true for the final crisp value. Its meaningfulness largely depends on the definition of what this value stands for. Results of further experiments to validate this new approach will be reported on in future.

References

1. Khan, M.S., Khor, S.W.: Framework for Fuzzy Rule-Based Cognitive Maps. LNCS, pp. 454–463. Springer, Heidelberg (2004)
2. Khan, M.S., Chong, A., Quaddus, M.: Fuzzy Cognitive Maps and Intelligent Decision Support - A Review. *Journal of Systems Research and Information Science* 26, 257–268 (2004)
3. Eloff, J.H.P., Smith, E.: Using Cognitive Modelling for enhanced Risk Assessment in a Health Care Institution. *IEEE Intelligent Systems and their Applications* 15(2), 69–75 (2000)
4. Carvalho, J.P., Tomé, J.A.: Rule-based Fuzzy Cognitive Maps and Fuzzy Cognitive Maps - A Comparative Study. In: *Proceedings of the 18th International Conference of the North American Fuzzy Information Processing Society, NAFIPS 1999*, New York USA (1999)
5. Carvalho, J.P., Tomé, J.A.: Expressing Time in Qualitative System Dynamics. In: *Proceedings of the 2001 FUZZ-IEEE*, Melbourne, Australia (2001)
6. Carvalho, J.P., Tomé, J.A.: Rule-based Fuzzy Cognitive Maps - Qualitative Systems Dynamics. In: *Proceedings of the 19th International Conference of the North American Fuzzy Information Processing Society, NAFIPS 2000*, Atlanta USA (2000)
7. Carvalho, J.P.: Mapas Cognitivos Baseados em Regras Difusas: Modelacao e Simulacao da Dinamica de Sistemas Qualitativos. In: *Instituto Superior Tecnico, Universidade Tecnica de Lisboa, Portuga, Lisbon, Portugal* (2002)
8. Eloff, J.H.P., Smith, E.: Transaction-based Risk Analysis - Using Cognitive Fuzzy Techniques. In: *IFI TC-11 8th Annual Working Conference on Information Security Management & Small Systems Security*, pp. 141–156 (2001)
9. Kosko, B.: Fuzzy Systems as Universal Approximators. *IEEE Transactions on Computers* 15, 1329–1333 (1994)
10. Kosko, B.: Fuzzy Cognitive Maps. *International Journal of Man-Machine Studies* 24, 65–75 (1986)
11. Kosko, B.: *Fuzzy Engineering*. Prentice-Hall, Saddle River (1997)
12. Axelrod, R.: *Framework for a General Theory of Cognition and Choice*. University of Berkely Press, Berkely (1972)
13. Axelrod, R.: *Structure of Decision*. Princeton University Press, Princeton (1976)
14. Achleitner, A.-K., Nathusius, E.: *Venture Valuation and Venture Capital Financing (German)*. *Wirtschaftswissenschaftliches Studium* (3), 134–139 (2004)

On Prediction Generation in Efficient MPC Algorithms Based on Fuzzy Hammerstein Models

Piotr M. Marusak

Institute of Control and Computation Engineering, Warsaw University of Technology,
ul. Nowowiejska 15/19, 00-665 Warszawa, Poland
P.Marusak@ia.pw.edu.pl

Abstract. In the paper a novel method of prediction generation, based on fuzzy Hammerstein models, is proposed. Using this method one obtains the prediction described by analytical formulas. The prediction has such a form that the MPC (Model Predictive Control) algorithm utilizing it can be formulated as a numerically efficient quadratic optimization problem. At the same time, the algorithm offers practically the same performance as the MPC algorithm in which a nonlinear, non-convex optimization problem must be solved at each iteration. It is demonstrated in the control system of the distillation column – a nonlinear control plant with significant time delay.

Keywords: fuzzy control, fuzzy systems, predictive control, nonlinear control, constrained control.

1 Introduction

In the MPC algorithms a model of the control plant is used to predict behavior of the control system. The control signals are generated using this prediction. Thanks to such an approach the MPC algorithms can be successfully used in control systems of processes with difficult dynamics (e.g. with large delay) and constraints, see e.g. [2,5,10,12].

The standard MPC algorithms use linear control plant models. In the case of a nonlinear control plant, however, such an approach may be inefficient especially if the control system should work in different operating points. Operation of the control system may be then improved using the MPC algorithm based on a nonlinear model. Usage of a nonlinear process model leads, however, to necessity of solving a nonlinear (often non-convex) optimization problem at each iteration of the algorithm. Such a problem is usually hard to solve (numerical problems may occur) and time needed to find the solution is hard to predict. Therefore, usually MPC algorithms with linear approximation of the control plant model, obtained at each iteration, are used [6,7,8,12].

Hammerstein models can be used to model many processes, like for example distillation columns or chemical reactors, see e.g. [4]. These models consist of

the nonlinear static part followed by the linear dynamic part. The Hammerstein models with fuzzy static part are considered in the paper as the fuzzy models offer many advantages [9], like e.g. relative easiness of model identification and relatively simple obtaining of linear approximation. It was also assumed that the dynamic part of the Hammerstein models under consideration has the form of the step responses.

A method of prediction generation using an approximation of the fuzzy Hammerstein model of the process is proposed in the paper. The approach exploits structure of the model. The prediction is obtained using the original fuzzy Hammerstein model and its linear approximation. It is done in such a way that a numerically efficient MPC algorithm, formulated as the standard quadratic programming problem, is obtained. The algorithm offers almost the same performance as the algorithm with nonlinear optimization and outperforms the standard MPC algorithm based on a linear model.

The next section contains description of MPC algorithms. In Sect. 3 the nonlinear prediction generation utilizing the fuzzy Hammerstein model is proposed. Example results illustrating performance offered by the MPC algorithm using the proposed method are presented in Sect. 4. The paper is summarized in the last section.

2 Model Predictive Control Algorithms

The Model Predictive Control (MPC) algorithms generate control signals predicting future behavior of the control plant many sampling instants ahead. The prediction is obtained using a process model. The values of manipulated variables are calculated in such a way that the prediction fulfills assumed criteria. Usually, these criteria are formulated as demand to minimize a performance function subject to the constraints of manipulated and output variables [2,5,10,12]:

$$\min_{\Delta \mathbf{u}} \left\{ J_{\text{MPC}} = \sum_{i=1}^p (\bar{y}_k - y_{k+i|k})^2 + \sum_{i=0}^{s-1} \lambda (\Delta u_{k+i|k})^2 \right\} \quad (1)$$

subject to:

$$\Delta \mathbf{u}_{\min} \leq \Delta \mathbf{u} \leq \Delta \mathbf{u}_{\max} \quad , \quad (2)$$

$$\mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \quad , \quad (3)$$

$$\mathbf{y}_{\min} \leq \mathbf{y} \leq \mathbf{y}_{\max} \quad , \quad (4)$$

where \bar{y}_k is a set-point value, $y_{k+i|k}$ is a value of the output for the $(k+i)^{\text{th}}$ sampling instant, predicted at the k^{th} sampling instant, $\Delta u_{k+i|k}$ are future changes in manipulated variable, $\lambda \geq 0$ is a weighting coefficient, p and s denote prediction and control horizons, respectively; $\mathbf{y} = [y_{k+1|k}, \dots, y_{k+p|k}]$, $\Delta \mathbf{u} = [\Delta u_{k+1|k}, \dots, \Delta u_{k+s-1|k}]$, $\mathbf{u} = [u_{k+1|k}, \dots, u_{k+s-1|k}]$, $\Delta \mathbf{u}_{\min}$, $\Delta \mathbf{u}_{\max}$,

\mathbf{u}_{\min} , \mathbf{u}_{\max} , \mathbf{y}_{\min} , \mathbf{y}_{\max} are vectors of lower and upper limits of changes and values of the control signal and of the values of the output variable, respectively. As a solution to the optimization problem (14) the optimal vector of changes in the manipulated variable is obtained. From this vector, the $\Delta u_{k|k}$ element is applied in the control system and the algorithm passes to the next iteration.

The predicted values of the output variable $y_{k+i|k}$ are derived using the dynamic control plant model. If this model is nonlinear then the optimization problem (14) is, in general, hard to solve, non-convex nonlinear optimization problem. Examples of such algorithms are described e.g. in [13].

2.1 MPC Algorithms Based on Linear Models

If the linear model is used in the MPC algorithm, the optimization problem (14) is a standard quadratic programming problem [2,5,10,12], because the superposition principle can be applied. Then, the vector of predicted output values \mathbf{y} is described by the following formula:

$$\mathbf{y} = \tilde{\mathbf{y}} + \mathbf{A} \cdot \Delta \mathbf{u} \quad (5)$$

where $\tilde{\mathbf{y}} = [\tilde{y}_{k+1|k}, \dots, \tilde{y}_{k+p|k}]$ is a free response of the plant which contains future values of the output variable calculated assuming that the control signal does not change in the prediction horizon; $\mathbf{A} \cdot \Delta \mathbf{u}$ is the forced response which depends only on future changes of the control signal (decision variables);

$$\mathbf{A} = \begin{bmatrix} a_1 & 0 & \dots & 0 & 0 \\ a_2 & a_1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_p & a_{p-1} & \dots & a_{p-s+2} & a_{p-s+1} \end{bmatrix} \quad (6)$$

is a matrix, called the dynamic matrix, composed of coefficients of the control plant step response a_i .

Let us introduce the vector $\bar{\mathbf{y}} = [\bar{y}_k, \dots, \bar{y}_k]$ of length p . The performance function from (1) rewritten in the matrix-vector form is as follows:

$$J_{\text{MPC}} = (\bar{\mathbf{y}} - \mathbf{y})^T \cdot (\bar{\mathbf{y}} - \mathbf{y}) + \Delta \mathbf{u}^T \cdot \mathbf{\Lambda} \cdot \Delta \mathbf{u} \quad (7)$$

where $\mathbf{\Lambda} = \lambda \cdot \mathbf{I}$ is the $s \times s$ matrix.

After application of the prediction (5) to the performance function (7) one obtains:

$$J_{\text{LMPC}} = (\bar{\mathbf{y}} - \tilde{\mathbf{y}} - \mathbf{A} \cdot \Delta \mathbf{u})^T \cdot (\bar{\mathbf{y}} - \tilde{\mathbf{y}} - \mathbf{A} \cdot \Delta \mathbf{u}) + \Delta \mathbf{u}^T \cdot \mathbf{\Lambda} \cdot \Delta \mathbf{u} \quad (8)$$

which depends quadratically on decision variables $\Delta \mathbf{u}$. The prediction (5) is also applied to the constraints (4). Thus, all constraints depend linearly on decision variables. As a result a standard linear-quadratic optimization problem is obtained.

3 MPC Algorithm Based on Fuzzy Hammerstein Models

It is assumed that the process model is of the Hammerstein structure (i.e. the nonlinear static block precedes the linear dynamic block) with fuzzy Takagi–Sugeno static part

$$z_k = f(u_k) = \sum_{j=1}^l w_j(u_k) \cdot z_k^j = \sum_{j=1}^l w_j(u_k) \cdot (b_j \cdot u_k + c_j) , \quad (9)$$

where z_k is the output of the static block, $w_j(u_k)$ are weights obtained using fuzzy reasoning, z_k^j are outputs of local models in the fuzzy static model, l is the number of fuzzy rules in the model, b_j and c_j are parameters of the local models in the fuzzy static part of the model. It is also assumed that the dynamic part of the model has the form of the step response:

$$\hat{y}_k = \sum_{n=1}^{p_d-1} a_n \cdot \Delta z_{k-n} + a_{p_d} \cdot z_{k-p_d} , \quad (10)$$

where \hat{y}_k is the output of the fuzzy Hammerstein model, a_i are coefficients of the step response, p_d is the horizon of the process dynamics (equal to the number of sampling instants after which the step response can be considered as settled).

The very basic idea of the proposed approach is to use the (nonlinear) model (10) to obtain the free response for the whole prediction horizon. In a case of any nonlinear model one can use the iterative approach to generate the free response, see e.g. [7]. However, if the Hammerstein model is used the analytical formulas describing the free response may be obtained in a straightforward way, similar to the case when a linear model is used for prediction.

The output of the model (10) in the i^{th} sampling instant is described by the following formula:

$$\hat{y}_{k+i} = \sum_{n=1}^i a_n \cdot \Delta z_{k-n+i} + \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta z_{k-n+i} + a_{p_d} \cdot z_{k-p_d+i} . \quad (11)$$

In (11) the first component depends on future action and the next ones on past control actions. The free response can be then calculated taking into consideration the estimated disturbances (containing also influence of modeling errors):

$$d_k = y_k - \hat{y}_k . \quad (12)$$

The final formula describing the elements of the free response is, thus, as follows:

$$\tilde{y}_{k+i|k} = \sum_{n=i+1}^{p_d-1} a_n \cdot \Delta z_{k-n+i} + a_{p_d} \cdot z_{k-p_d+i} + d_k , \quad (13)$$

where d_k is the DMC-type disturbance model (it is assumed the same for all instants in the prediction horizon).

After calculating the free response in the way described above, the dynamic matrix, needed to predict the influence of the future control changes (generated by the algorithm) is derived. It is done using at each algorithm iteration a linear approximation of the fuzzy Hammerstein model (10):

$$\hat{y}_k = dz_k \cdot \left(\sum_{n=1}^{p_d-1} a_n \cdot \Delta u_{k-n} + a_{p_d} \cdot u_{k-p_d} \right), \quad (14)$$

where dz_k is a slope of the static characteristic near the z_k . It can be calculated numerically using the formula

$$dz_k = \sum_{j=1}^l (w_j(u_k + du) \cdot (b_j \cdot (u_k + du) + c_j) - w_j(u_k) \cdot (b_j \cdot u_k + c_j)) / du, \quad (15)$$

where du is a small number.

Thus, the dynamic matrix will be described by the following formula:

$$\mathbf{A}_k = dz_k \cdot \begin{bmatrix} a_1 & 0 & \dots & 0 & 0 \\ a_2 & a_1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_p & a_{p-1} & \dots & a_{p-s+2} & a_{p-s+1} \end{bmatrix}. \quad (16)$$

Then, the free response (13) and the dynamic matrix (16) are used to obtain the prediction:

$$\mathbf{y} = \tilde{\mathbf{y}} + \mathbf{A}_k \cdot \Delta \mathbf{u}. \quad (17)$$

After application of prediction (17) to the performance function from (1) one obtains:

$$J_{\text{FMPC}} = (\bar{\mathbf{y}} - \tilde{\mathbf{y}} - \mathbf{A}_k \cdot \Delta \mathbf{u})^T \cdot (\bar{\mathbf{y}} - \tilde{\mathbf{y}} - \mathbf{A}_k \cdot \Delta \mathbf{u}) + \Delta \mathbf{u}^T \cdot \mathbf{A} \cdot \Delta \mathbf{u}. \quad (18)$$

The prediction (17) is also used in constraints (4). Then, the linear-quadratic optimization problem with the performance function (18) and constraints (2)–(4) is solved at each iteration in order to derive the control signal.

4 Simulation Experiments

4.1 Control Plant

The control plant under consideration is an ethylene distillation column DA-303 from petrochemical plant in Plock. It is a highly nonlinear plant with a large time delay. The Hammerstein model of the plant is shown in Fig. 1 the steady-state characteristic of the plant is shown in Fig. 2. The output of the plant y is the impurity of the product. From the economic efficiency point of view it should be as high as it is allowed but the limit cannot be exceeded, otherwise the product is lost. Therefore, responses without overshoot are preferred. The manipulated

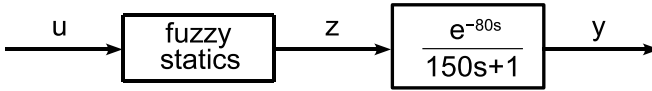


Fig. 1. Hammerstein model of the distillation column

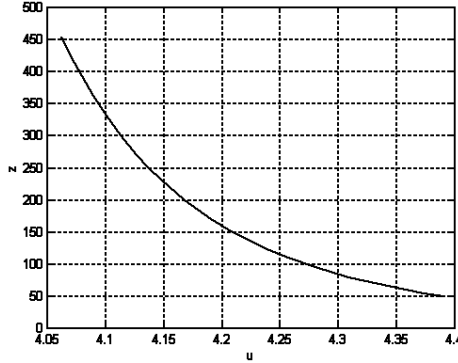


Fig. 2. Steady-state characteristic of the plant

variable u is the reflux. The higher it is the purer product is obtained. During experiments it was assumed that the reflux is constrained $4.05 \leq u \leq 4.4$.

The static part of the Hammerstein model in the form of the Takagi–Sugeno model was used, with three local models of the form: $z_k^j = b_j \cdot u_k + c_j$, where $b_1 = -2222.4$, $b_2 = -1083.2$, $b_3 = -534.4$, $c_1 = 9486$, $c_2 = 4709.3$, $c_3 = 2408.7$. The assumed membership functions are shown in Fig. 3.

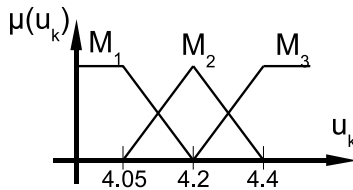


Fig. 3. Membership functions of the static part of the Hammerstein model

4.2 Simulation Experiments

Three MPC algorithms were designed for the considered control plant: an NMPC one (with nonlinear optimization), an LMPC one (with a linear model) and an FMPC one (with prediction method proposed in the paper, exploiting the fuzzy Hammerstein model). The sampling period was assumed equal to $T_s = 20$ min; tuning parameters of all three algorithms were as follows: prediction horizon $p = 44$, control horizon $s = 20$, weighting coefficient $\lambda = 8e6$.

During the experiments performance of control systems with NMPC, LMPC and FMPC algorithms was compared. The example responses obtained after the changes of set-point value are shown in Fig. 4. The responses obtained in the control system with the FMPC algorithm (solid lines in Fig. 4) are very close to those obtained with the NMPC algorithm with full nonlinear optimization (dotted lines in Fig. 4). It should be, however, stressed that the FMPC algorithm is based on the reliable quadratic programming routine.

The responses obtained using the FMPC and NMPC algorithms have very small overshoot. Moreover, the character of these responses is the same for different set-points. The standard LMPC algorithm (dashed lines in Fig. 4) operates almost as good as its counterparts for the set-point change to $\bar{y}_1 = 200$ ppm. However, it works unacceptably bad in the case of the set-point change to $\bar{y}_2 = 300$ ppm. It is the result of significant nonlinearity of the control plant.

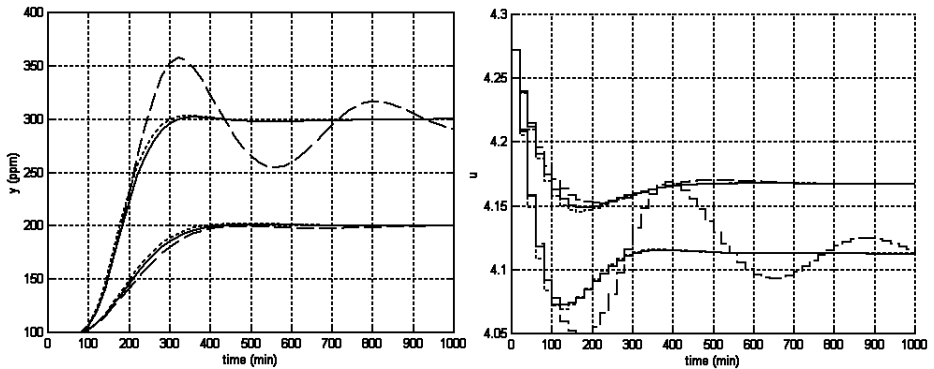


Fig. 4. Responses of the control systems to the change of the set-point values to $\bar{y}_1 = 200$ ppm and $\bar{y}_2 = 300$ ppm; FMPC – solid lines, NMPC – dotted lines and LMPC – dashed lines

5 Summary

The method of nonlinear prediction derivation based on the fuzzy Hammerstein model was proposed in the paper. It uses the nonlinear model to derive the free response of the control plant and its linear approximation to calculate the influence of future control action. Thanks to such an approach the MPC algorithm, using the proposed method of prediction, is formulated as the linear-quadratic optimization problem. Despite that, it outperforms its counterparts based on linear process models and offers practically the same control performance as the algorithms with nonlinear optimization.

Acknowledgment. This work was supported by the Polish national budget funds for science 2009–2011 as a research project.

References

1. Babuska, R., te Braake, H.A.B., van Can, H.J.L., Krijgsman, A.J., Verbruggen, H.B.: Comparison of intelligent control schemes for real-time pressure control. *Control Engineering Practice* 4, 1585–1592 (1996)
2. Camacho, E.F., Bordons, C.: *Model Predictive Control*. Springer, Heidelberg (1999)
3. Fink, A., Fischer, M., Nelles, O., Isermann, R.: Supervision of nonlinear adaptive controllers based on fuzzy models. *Control Engineering Practice* 8, 1093–1105 (2000)
4. Janczak, A.: *Identification of nonlinear systems using neural networks and polynomial models: a block-oriented approach*. Springer, Heidelberg (2005)
5. Maciejowski, J.M.: *Predictive control with constraints*. Prentice Hall, Harlow (2002)
6. Marusak, P.: Advantages of an easy to design fuzzy predictive algorithm in control systems of nonlinear chemical reactors. *Applied Soft Computing* 9, 1111–1125 (2009)
7. Marusak, P.: Efficient model predictive control algorithm with fuzzy approximations of nonlinear models. In: Kolehmainen, M., Toivanen, P., Beliczynski, B. (eds.) *ICANNGA 2009*. LNCS, vol. 5495, pp. 448–457. Springer, Heidelberg (2009)
8. Morari, M., Lee, J.H.: *Model predictive control: past, present and future*. *Computers and Chemical Engineering* 23, 667–682 (1999)
9. Piegat, A.: *Fuzzy Modeling and Control*. Physica-Verlag, Berlin (2001)
10. Rossiter, J.A.: *Model-Based Predictive Control*. CRC Press, Boca Raton (2003)
11. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its application to modeling and control. *IEEE Trans. Systems, Man and Cybernetics* 15, 116–132 (1985)
12. Tatjewski, P.: *Advanced Control of Industrial Processes; Structures and Algorithms*. Springer, London (2007)

Fuzzy Number as Input for Approximate Reasoning and Applied to Optimal Control Problem^{*}

Takashi Mitsuishi¹ and Yasunari Shidama²

¹ University of Marketing and Distribution Sciences, Kobe 651-2188, Japan

Takashi_MITSUISHI@red.ums.ac.jp

² Shinshu University, Nagano 380-8553, Japan

Abstract. The authors present a mathematical framework for studying a fuzzy logic control, which is constructed by IF-THEN type fuzzy rules through a kind of product-sum-gravity method. Moreover, fuzzy numbers are used instead of definite values (crisp numbers) as premise variables in IF-THEN fuzzy rule.

Keywords: Fuzzy logic control, approximate reasoning, L-R fuzzy number, functional space.

1 Introduction

In 1965 Zadeh introduced the notion of fuzziness [1] [2] and then Mamdani has applied fuzzy logic to the field of control theory [3]. After that, fuzzy control has been increased and studied widely, since it able to realize numerically the control represented by human language and sensitivity. In practical use, fuzzy membership functions, which represent input and output states in optimal control system, are decided on the basis of the experience of experts in each peculiar plant.

The optimization of fuzzy control discussed in this paper is different from conventional method such as classical control and modern control. We consider fuzzy optimal control problems as problems of finding the minimum (maximum) value of the performance function with feedback law constructed by approximate reasoning [4]. In this study, we analyzed IF-THEN type fuzzy rule that premise variables are fuzzy number not crisp, and proved compactness of the set of membership functions in L^∞ space with the condition that the approximate reasoning is continuous. To guarantee the convergence of optimal solution, the compactness of the set of membership functions in L^∞ space is proved. And assuming approximate reasoning to be a functional on the set of membership functions, its continuity is obtained. Then, it is shown that the system has an optimal feedback control by essential use of compactness of sets of fuzzy membership functions. The tuple of membership functions, in other words IF-THEN

^{*} The paper was supported in part by Grant-in-Aid for Young Scientists (B) #19700225 from Japan Society for the Promotion of Science (JSPS).

rules, which minimize the integral cost function of fuzzy logic control using the fuzzy numbers like the linguistic inputs exists in L^∞ space. Although in our previous work [5], the set of membership functions of fuzzy sets in the premise part of IF-THEN rules has Lipschitz condition. It is able to be removed in this study.

2 Nonlinear Feedback System

In this study we assume that the feedback part in this system is calculated by approximate reasoning presented in the next section. Using an idea and framework mentioned in the following section 4 and 5, the existence of optimal control based on fuzzy rules will be designed.

\mathbb{R}^n denotes the n -dimensional Euclidean space with the usual norm $\|\cdot\|$. Let $f(v_1, v_2) : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$ be a (nonlinear) vector valued function which is Lipschitz continuous. In addition, assume that there exists a constant $M_f > 0$ such that $\|f(v_1, v_2)\| \leq M_f (\|v_1\| + |v_2| + 1)$ for all $(v_1, v_2) \in \mathbb{R}^n \times \mathbb{R}$. Consider a system given by the following state equation: $\dot{x}(t) = f(x(t), u(t))$, where $x(t)$ is the state and the control input $u(t)$ of the system is given by the state feedback $u(t) = \rho(x(t))$. For a sufficiently large $r > 0$, $B_r := \{x \in \mathbb{R}^n : \|x\| \leq r\}$ denotes a bounded set containing all possible initial states x_0 of the system. Let T be a sufficiently large final time. Then, we have

Proposition 1. [6]. *Let $\rho : \mathbb{R}^n \rightarrow \mathbb{R}$ be a Lipschitz continuous function and $x_0 \in B_r$. Then, the state equation*

$$\dot{x}(t) = f(x(t), \rho(x(t))) \tag{1}$$

has a unique solution $x(t, x_0, \rho)$ on $[0, T]$ with the initial condition $x(0) = x_0$ such that the mapping

$$(t, x_0) \in [0, T] \times B_r \mapsto x(t, x_0, \rho)$$

is continuous.

For any $r_2 > 0$, denote by Φ the set of Lipschitz continuous functions $\rho : \mathbb{R}^n \rightarrow \mathbb{R}$ satisfying

$$\sup_{u \in \mathbb{R}^n} |\rho(u)| \leq r_2.$$

Then, the following 1) and 2) hold.

1) For any $t \in [0, T]$, $x_0 \in B_r$ and $\rho \in \Phi$, $\|x(t, x_0, \rho)\| \leq r_1$, where

$$r_1 \equiv e^{M_f T} r + (e^{M_f T} - 1)(r_2 + 1). \tag{2}$$

2) Let $\rho_1, \rho_2 \in \Phi$. Then, for any $t \in [0, T]$ and $x_0 \in B_r$,

$$\|x(t, x_0, \rho_1) - x(t, x_0, \rho_2)\| \leq \frac{e^{L_f(1+L_{\rho_1})t} - 1}{1 + L_{\rho_1}} \sup_{u \in [-r_1, r_1]^n} |\rho_1(u) - \rho_2(u)|, \tag{3}$$

where L_f and L_{ρ_1} are the Lipschitz constants of f and ρ_1 .

3 Fuzzy Logic Control Using Fuzzy Numbers for Premise Variables

In this section we briefly explain the fuzzy control method which are widely used in practical applications for the convenience of the reader.

3.1 IF-THEN Rules

Assume the feedback law ρ consists of the following m IF-THEN type fuzzy control rules are given [7].

$$\text{IF } X_1(x_1) \text{ is } A_{i1} \text{ and } \dots \text{ and } X_n(x_n) \text{ is } A_{in} \text{ THEN } y \text{ is } B_i \quad (i = 1, \dots, m). \quad (4)$$

Here, $X_1(x_1), \dots, X_n(x_n)$ are fuzzy numbers which are used instead of the definite values as the premise variables. In this paper, we use L-R fuzzy numbers characterized and defined by the following membership function.

$$\mu_{X_j}(x_j, z_j) = \begin{cases} \frac{1-|x_j-z_j|}{e} & \text{if } z_j \in [x_j - e, x_j + e], \\ 0 & \text{otherwise.} \end{cases}$$

x_j ($j = 1, \dots, n$) and $e > 0$ are the center and the width of L-R fuzzy number respectively. The solutions of the nonlinear state equation are applied to x_j ($j = 1, \dots, n$). In practical applications if e is small enough, the fuzzy number can be considered approximately to be a crisp number.

Let $\mu_{A_{ij}}$ ($i = 1, \dots, m; j = 1, \dots, n$) be fuzzy membership functions defined on a certain closed interval of the fuzzy set A_{ij} . Let μ_{B_i} ($i = 1, \dots, m; j = 1, \dots, n$) be fuzzy membership functions defined on the other closed interval of the fuzzy set B_i . For simplicity, we write ‘‘IF’’ and ‘‘THEN’’ parts in the rules by the following notation:

$$\mathcal{A}_i = (\mu_{A_{i1}}, \dots, \mu_{A_{in}}) \quad (i = 1, \dots, m), \quad \mathcal{A} = (\mathcal{A}_1, \dots, \mathcal{A}_m), \quad \mathcal{B} = (\mu_{B_1}, \dots, \mu_{B_m}).$$

Then, the IF-THEN type fuzzy control rules (4) is called a fuzzy controller, and is denoted by $(\mathcal{A}, \mathcal{B})$. In the rules, the pair of fuzzy number

$$X(x) = (X_1(x_1), X_2(x_2), \dots, X_n(x_n))$$

is called an input information given to the fuzzy controller $(\mathcal{A}, \mathcal{B})$, and y is called an control variable.

3.2 Approximate Reasoning

Tanaka proposed that Mamdani method is applied as the approximate reasoning in the case that premise variables are fuzzy numbers [7]. Since Mamdani method uses minimum and maximum operations, the value of the agreement degree and the center of gravity might not change smoothly. Mizumoto proposed the product-sum-gravity method by replacing minimum with product and maximum with summation [9]. In addition, Mamdani method is not necessarily continuous

with respect to the weak topology on L^∞ . In this study, when an input information $X(x)$ is given to the fuzzy controller $(\mathcal{A}, \mathcal{B})$, then one can obtain the amount of operation $\rho_{\mathcal{A}\mathcal{B}}(x)$ from the controller through the following calculation:

Step 1: The degree of each rule is calculated by

$$\alpha_{A_i}(x) = \prod_{j=1}^n \max_{z_j} (\mu_{X_j}(x_j, z_j) \wedge \mu_{A_{ij}}(z_j)) \quad (i = 1, \dots, m).$$

Step 2: The inference result of each rule is calculated by

$$\beta_{A_i B_i}(x, y) = \alpha_{A_i}(x) \mu_{B_i}(y) \quad (i = 1, \dots, m).$$

Step 3: The inference result of all rules is calculated by

$$\gamma_{\mathcal{A}\mathcal{B}}(x, y) = \sum_{i=1}^m \beta_{B_i A_i}(x, y).$$

Step 4: Difuzzification stage. The center of gravity of the inference result is calculated by

$$\rho_{\mathcal{A}\mathcal{B}}(x) = \frac{\int y \gamma_{\mathcal{A}\mathcal{B}}(x, y) dy}{\int \gamma_{\mathcal{A}\mathcal{B}}(x, y) dy}.$$

4 Compactness of a Set of Membership Functions

In this section, we introduce two sets of fuzzy membership functions and study their topological properties. Then we can show that a set of admissible fuzzy controllers is compact and metrizable with respect to an appropriate topology on fuzzy membership functions.

In the following fix $r > 0$, a sufficiently large $r_2 > 0$ and a final time T of the control (1). Put r_1 be the positive constant determined by (2). Denote by $L^\infty[-r_1, r_1]$ the Banach space of all Lebesgue measurable, essentially bounded real functions on $[-r_1, r_1]$. We consider the following two sets of fuzzy membership functions.

$$G_1 = \{\mu \in L^\infty[-r_1, r_1] : 0 \leq \mu(x) \leq 1 \text{ a.e. } x \in [-r_1, r_1]\}$$

and

$$G_2 = \{\mu \in L^\infty[-r_2, r_2] : 0 \leq \mu(y) \leq 1 \text{ a.e. } y \in [-r_2, r_2]\}.$$

We assume that the fuzzy membership function $\mu_{A_{ij}}$ in “IF” parts of the rules (4) belongs to the set G_1 . On the other hand, we also assume that the fuzzy membership function μ_{B_i} in “THEN” parts of the rules (4) belongs to the set G_2 . In the following, we endow the space G_1 and G_1 with the weak topology on $L^\infty[-r_1, r_1]$ and $L^\infty[-r_2, r_2]$, respectively. Then G_1 and G_2 are compact metrizable for the weak topology [8]. Put

$$\mathcal{F} = (G_1^n \times G_2)^m,$$

where G^m denotes the m times Cartesian product of G . Then, every element $(\mathcal{A}, \mathcal{B})$ of \mathcal{F} is a fuzzy controller given by the IF-THEN type fuzzy control rules (4). By the Tychonoff theorem, we can have following proposition.

Proposition 2. \mathcal{F} is compact and metrizable with respect to the product topology on $(G_1^n \times G_2)^m$.

In the defuzzification stage of product-sum-gravity method, the amount of operation is obtained through the gravity calculation

$$\rho_{\mathcal{A}\mathcal{B}}(x) = \frac{\int_{-r_2}^{r_2} y\gamma_{\mathcal{A}\mathcal{B}}(x, y)dy}{\int_{-r_2}^{r_2} \gamma_{\mathcal{A}\mathcal{B}}(x, y)dy}, \quad x = (x_1, \dots, x_n) \in [-r_1, r_1]^n.$$

To avoid making the denominator of the expression above equal to 0, for any $\delta > 0$, we consider the set

$$\mathcal{F}_\delta = \left\{ (\mathcal{A}, \mathcal{B}) \in \mathcal{F} : \int_{-r_2}^{r_2} \gamma_{\mathcal{A}\mathcal{B}}(x, y)dy \geq \delta \text{ for all } x \in [-r_1, r_1]^n \right\}, \quad (5)$$

which is a slight modification of \mathcal{F} . If δ is taken small enough, it is possible to consider $\mathcal{F} = \mathcal{F}_\delta$ for practical applications. We say that an element $(\mathcal{A}, \mathcal{B})$ of \mathcal{F}_δ is an admissible fuzzy controller. Then, we have the following

Proposition 3. The set \mathcal{F}_δ of all admissible fuzzy controllers is compact and metrizable with respect to the product topology on $(G_1^n \times G_2)^m$.

Proof. Assume that a sequence $\{(\mathcal{A}^k, \mathcal{B}^k)\}$ in \mathcal{F}_δ converges to $(\mathcal{A}, \mathcal{B}) \in \mathcal{F}$. Fix $x \in [-r_1, r_1]^n$. Then, it is easy to show that

$$\int_{-r_2}^{r_2} \gamma_{\mathcal{A}\mathcal{B}}(x, y)dy = \lim_{k \rightarrow \infty} \int_{-r_2}^{r_2} \gamma_{\mathcal{A}^k \mathcal{B}^k}(x, y)dy \geq \delta,$$

and this implies $(\mathcal{A}, \mathcal{B}) \in \mathcal{F}_\delta$. Therefore, \mathcal{F}_δ is a closed subset of \mathcal{F} , and hence it is compact metrizable.

5 Unique Solution of State Equation

In this paper, for any pair $(\mathcal{A}, \mathcal{B}) \in \mathcal{F}_\delta$, we define the feedback function

$$\rho_{\mathcal{A}\mathcal{B}}(x) = \rho_{\mathcal{A}\mathcal{B}}(x_1, \dots, x_n) : [-r_1, r_1]^n \rightarrow \mathbb{R}$$

on the basis of the rules by the approximate reasoning which is product-sum-gravity method in section 3. To apply the proposition 1, the following proposition is needed for the existence of unique solution of the state equation (1).

Proposition 4. Let $(\mathcal{A}, \mathcal{B}) \in \mathcal{F}_\delta$. Then, the following 1) and 2) hold.

- 1) $\rho_{\mathcal{A}\mathcal{B}}$ is Lipschitz continuous on $[-r_1, r_1]^n$.
- 2) $|\rho_{\mathcal{A}\mathcal{B}}(x)| \leq r_2$ for all $x \in [-r_1, r_1]^n$.

Proof. 1) For any $x, x' \in [-r_1, r_1]^n$ and any $i = 1, \dots, m$, we have

$$|\alpha_{\mathcal{A}_i}(x) - \alpha_{\mathcal{A}_i}(x')| \leq \frac{\sqrt{n}}{2e} \|x - x'\| \tag{6}$$

Hence the mapping $\alpha_{\mathcal{A}_i}$ is Lipschitz continuous on $[-r_1, r_1]^n$.

Noting that $|\mu_{B_i}(y)| \leq 1$, we have

$$|\beta_{B_i \mathcal{A}_i}(x, y) - \beta_{B_i \mathcal{A}_i}(x', y)| \leq \frac{\sqrt{n}}{2e} \|x - x'\| \tag{7}$$

and

$$|\gamma_{\mathcal{A}\mathcal{B}}(x, y) - \gamma_{\mathcal{A}\mathcal{B}}(x', y)| \leq \frac{m\sqrt{n}}{2e} \|x - x'\|. \tag{8}$$

Put

$$g(x) = \int_{-r_2}^{r_2} y \gamma_{\mathcal{A}\mathcal{B}}(x, y) dy, \quad h(x) = \int_{-r_2}^{r_2} \gamma_{\mathcal{A}\mathcal{B}}(x, y) dy.$$

In the same way, noting that $|g(x)| \leq r_2^2$ and $\delta \leq |h(x)| \leq 2r_2$ for all $x \in [-r_1, r_1]^n$, it follows from (6), (7) and (8) that

$$\begin{aligned} |\rho_{\mathcal{A}\mathcal{B}}(x) - \rho_{\mathcal{A}\mathcal{B}}(x')| &\leq \frac{|g(x) - g(x')||h(x')| + |h(x) - h(x')||g(x')|}{\delta^2} \\ &\leq \frac{2r_2^3 m \sqrt{n}}{e \delta^2} \|x - x'\|, \end{aligned}$$

and the Lipschitz continuity of $\rho_{\mathcal{A}\mathcal{B}}$ is proved. The proof of 2) is omitted.

Let $(\mathcal{A}, \mathcal{B})$ be a fuzzy controller given by the IF-THEN type fuzzy control rules (4). We say that the system (1) is a fuzzy feedback system if the control function $u(t)$ is given by the state feedback $u(t) = \rho_{\mathcal{A}\mathcal{B}}(x(t))$, where $\rho_{\mathcal{A}\mathcal{B}}(x(t))$ is the amount of operation from the fuzzy controller $(\mathcal{A}, \mathcal{B})$ for an input information $x(t)$.

It is easily seen that every bounded Lipschitz function $\rho : [-r_1, r_1]^n \rightarrow \mathbb{R}$ can be extended to a bounded Lipschitz function $\tilde{\rho}$ on \mathbb{R}^n without increasing its Lipschitz constant and bound. In fact, define $\tilde{\rho} : \mathbb{R}^n \rightarrow \mathbb{R}$ by

$$\tilde{\rho}(x) = \tilde{\rho}(x_1, \dots, x_n) = \begin{cases} \rho(x_1, \dots, x_n), & \text{if } x \in [-r_1, r_1]^n \\ \rho(\varepsilon(x_1)r_1, \dots, \varepsilon(x_n)r_1), & \text{if } x \notin [-r_1, r_1]^n, \end{cases}$$

where

$$\varepsilon(u) = \begin{cases} 1, & \text{if } u > r_1 \\ -1, & \text{if } u < -r_1. \end{cases}$$

Let $(\mathcal{A}, \mathcal{B}) \in \mathcal{F}_\delta$. Then it follows from proposition 3 and the fact above that the extension $\tilde{\rho}_{\mathcal{A}\mathcal{B}}$ of $\rho_{\mathcal{A}\mathcal{B}}$ is Lipschitz continuous on \mathbb{R}^n with the same Lipschitz constant of $\rho_{\mathcal{A}\mathcal{B}}$ and satisfies $\sup_{u \in \mathbb{R}^n} |\tilde{\rho}_{\mathcal{A}\mathcal{B}}(u)| \leq r_2$. Therefore, by proposition 1 the state equation (1) for the feedback law $\tilde{\rho}_{\mathcal{A}\mathcal{B}}$ has a unique solution $x(t, x_0, \tilde{\rho}_{\mathcal{A}\mathcal{B}})$ with the initial condition $x(0) = x_0$ [10]. Though the extension $\tilde{\rho}_{\mathcal{A}\mathcal{B}}$ of $\rho_{\mathcal{A}\mathcal{B}}$ is not unique in general, the solution $x(t, x_0, \tilde{\rho}_{\mathcal{A}\mathcal{B}})$ is uniquely determined by $\rho_{\mathcal{A}\mathcal{B}}$ using the inequality (3) of 2) of proposition 1. Consequently, in the following the extension $\tilde{\rho}_{\mathcal{A}\mathcal{B}}$ is written as $\rho_{\mathcal{A}\mathcal{B}}$ without confusion.

6 Application to Optimal Control Problem

The performance index of this fuzzy feedback control system is evaluated with the following integral performance function:

$$J = \int_{B_r} \int_0^T w(x(t, \zeta, \rho_{\mathcal{A}\mathcal{B}}), \rho_{\mathcal{A}\mathcal{B}}(x(t, \zeta, \rho_{\mathcal{A}\mathcal{B}}))) dt d\zeta, \quad (9)$$

where $w : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$ is a positive continuous function. The following theorem guarantees the existence of a fuzzy controller $(\mathcal{A}, \mathcal{B})$ (IF-THEN type fuzzy control rules) which minimizes and maximize the previous performance function (9).

Theorem 1. *The mapping*

$$(\mathcal{A}, \mathcal{B}) \mapsto \int_{B_r} \int_0^T w(x(t, \zeta, \rho_{\mathcal{A}\mathcal{B}}), \rho_{\mathcal{A}\mathcal{B}}(x(t, \zeta, \rho_{\mathcal{A}\mathcal{B}}))) dt d\zeta$$

has a minimum (maximum) value on the compact space \mathcal{F}_δ defined by (5).

Proof. Since compactness of \mathcal{F}_δ is already derived by proposition 2, it is sufficient to prove that the performance function is continuous on \mathcal{F}_δ . Routine calculation gives the estimate

$$\begin{aligned} & \sup_{x \in [-r_1, r_1]^n} |\rho_{\mathcal{A}^k \mathcal{B}^k}(x) - \rho_{\mathcal{A}\mathcal{B}}(x)| \\ & \leq \frac{r_2 m}{\delta^2} \left\{ 2 \sum_{i=1}^m \left| \int_{-r_2}^{r_2} y \mu_{B_i^k}(y) dy - \int_{-r_2}^{r_2} y \mu_{B_i}(y) dy \right| \right. \\ & \left. + r_2 \sum_{i=1}^m \left| \int_{-r_2}^{r_2} \mu_{B_i^k}(y) dy - \int_{-r_2}^{r_2} \mu_{B_i}(y) dy \right| \right\} + \frac{4r_2^3 m}{\delta^2} \sum_{i=1}^m \|\alpha_{\mathcal{A}_i^k} - \alpha_{\mathcal{A}_i}\|_\infty. \end{aligned}$$

Assume that $(\mathcal{A}^k, \mathcal{B}^k) \rightarrow (\mathcal{A}, \mathcal{B})$ in \mathcal{F}_δ and fix $(t, \zeta) \in [0, T] \times B_r$. Then it follows from the estimate above that

$$\lim_{k \rightarrow \infty} \sup_{x \in [-r_1, r_1]^n} |\rho_{\mathcal{A}^k \mathcal{B}^k}(x) - \rho_{\mathcal{A}\mathcal{B}}(x)| = 0. \quad (10)$$

Hence, by 2) of proposition 1, we have

$$\lim_{k \rightarrow \infty} \|x(t, \zeta, \rho_{\mathcal{A}^k \mathcal{B}^k}) - x(t, \zeta, \rho_{\mathcal{A}\mathcal{B}})\| = 0. \quad (11)$$

Further, it follows from (10), (11) and 1) of proposition 1 that

$$\lim_{k \rightarrow \infty} \rho_{\mathcal{A}^k \mathcal{B}^k}(x(t, \zeta, \rho_{\mathcal{A}^k \mathcal{B}^k})) = \rho_{\mathcal{A}\mathcal{B}}(x(t, \zeta, \rho_{\mathcal{A}\mathcal{B}})). \quad (12)$$

Noting that $w : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$ is positive and continuous, it follows from (11), (12) and the Lebesgue's dominated convergence theorem that the mapping is continuous on the compact metric space \mathcal{F}_δ . Thus it has a minimum (maximum) value on \mathcal{F}_δ , and the proof is complete.

7 Conclusion

Fuzzy logic control which the feedback law is constructed by fuzzy approximate reasoning in the nonlinear feedback control was studied. In this paper fuzzy numbers are used as premise variable in IF-THEN fuzzy production rules. Lipschitz conditions are able to be removed from all the membership functions of fuzzy sets which construct IF-THEN rules by using L-R fuzzy numbers. Lipschitz continuity of the product-sum-gravity method on the state variables was proved. Then the existence of unique solution of the state equation was obtained. On the other, since the product-sum-gravity method on the set of membership functions in L-infinity space is continuous as functional, the existence of tuple of membership functions which minimize the cost function of the feedback control was proved.

This means that the IF-THEN rules which give optimal feedback law to nonlinear feedback control exist. It is recognized that in various applications it could be a useful tool in analyzing the convergence of fuzzy control rules modified recursively.

References

1. Zadeh, L.A.: Fuzzy Sets. *Information and Control* 8, 338–353 (1965)
2. Zadeh, L.A.: Fuzzy algorithms. *Information and Control* 12, 94–102 (1968)
3. Mamdani, E.H.: Application of fuzzy algorithms for control of simple dynamic plant. In: *Proc. IEE*, vol. 121(12), pp. 1585–1588 (1974)
4. Shidama, Y., Yang, Y., Eguchi, M., Yamaura, H.: The compactness of a set of membership functions and its application to fuzzy optimal control. *The Japan Society for Industrial and Applied Mathematics* 6(1), 1–13 (1996)
5. Mitsuishi, T., Shidama, Y.: Continuity of Product-Sum-Gravity Method on L^2 Space Using Fuzzy Number for Premise Variable. In: *Proc. of 2007 9th International Symposium on Signal Processing and Its Applications*, CD-ROM (2007)
6. Mitsuishi, T., Kawabe, J., Wasaki, K., Shidama, Y.: Optimization of Fuzzy Feedback Control Determined by Product-Sum-Gravity Method. *Journal of Nonlinear and Convex Analysis* 1(2), 201–211 (2000)
7. Tanaka, K.: *Advanced Fuzzy Control*. Kyoritsu Syuppan, Tokyo (1994)
8. Mitsuishi, T., Endou, N., Shidama, Y.: Continuity of Nakamori Fuzzy Model and Its Application to Optimal Feedback Control. In: *Proc. IEEE International Conference on Systems, Man and Cybernetics*, pp. 577–581 (2005)
9. Mizumoto, M.: Improvement of fuzzy control (IV) - Case by product-sum-gravity method. In: *Proc. 6th Fuzzy System Symposium*, pp. 9–13 (1990)
10. Miller, R.K., Michel, A.N.: *Ordinary Differential Equations*. Academic Press, New York (1982)
11. Riesz, F., Sz.-Nagy, B.: *Functional Analysis*. Dover Publications, New York (1990)
12. Dunford, N., Schwartz, J.T.: *Linear Operators Part I: General Theory*. John Wiley & Sons, New York (1988)

Fuzzy Functional Dependencies in Multiargument Relationships

Krzysztof Myszkorowski

Institute of Information Technology, Technical University of Łódź,
Wólczajska 215, 93-005 Łódź, Poland

kamysz@ics.p.lodz.pl

<http://edu.ics.p.lodz.pl>

Abstract. A multiargument relationship may be formally presented using the relational notation: $R(X_1, X_2, \dots, X_n)$, where R is the name of the relationship, and attributes X_i denote keys of entity sets which participate in it. The dependencies between all n attributes describe the integrity constraints and must not be infringed. They constitute a restriction for relationships of fewer attributes. In the paper an analysis of fuzzy functional dependencies between attributes of R is presented. Attribute vales are given by means of possibility distributions.

Keywords: Fuzzy relational database model, fuzzy functional dependencies, possibility distribution, implication of fuzzy data dependencies, n -ary relationships.

1 Introduction

Conventional database systems are designed with the assumption of precision of information collected in them. The problem becomes more complex if our knowledge of the fragment of reality to be modeled is imperfect [1]. In such cases one has to apply tools for describing uncertain or imprecise information [2,3,4]. One of them is the fuzzy set theory [5,6]. So far, a great deal of effort has been devoted to the development of fuzzy data models [7,8,9,10,11]. Numerous works discuss how uncertainty existing in databases should be handled. Some authors proposed incorporating fuzzy logic into data modeling techniques. There are two major approaches concerning fuzzy data representation, namely, the similarity-based approach [12] and the possibility-based approach [13]. In the former domains of attributes are associated with similarity relations and attribute values can be ordinary subsets of their domains. In the latter attribute values are expressed by means of possibility distributions [14].

The aim of the paper is to analyze fuzzy multiargument relationships. Apart from relationships between all the attributes one has to consider relationships of fewer attributes which are embedded in them. This issue for ternary relationships in conventional databases was investigated in [15,16]. For various types of ternary relationships the authors formulated the rules to which cardinalities of binary relationships between pairs of sets are subjected. The analysis can be

also carried out using the theory of functional dependencies (FDs) which reflect integrity constraints and should be studied during the design process. In fuzzy databases the notion of functional dependency has to be modified. Hence different approaches concerning fuzzy functional dependencies (FFDs) have been described in professional literature. A number of different definitions emerged (see for example [13,17,18,19,20,21]). In further considerations it will be applied the definition proposed by Chen [22].

The paper is organized as follows. Functional dependencies, which may exist between attributes of n -ary relationships, are described in the next section. In section 3 some concepts related to fuzzy relational databases are presented. Section 4 formulates the rules to which fuzzy functional dependencies in multiargument relationships must be subordinated.

2 Functional Dependencies as Integrity Constraints for Multiargument Relationships

A multiargument relationship R may be formally presented using the relational notation: $R(X_1, X_2, \dots, X_n)$, where R is a relation scheme, and attributes X_i denote keys of entity sets which participate in it. The n number is called the relationship degree. The relationship's cardinality may be presented as $M_1: M_2: \dots : M_n$ where M_i denotes the number of X_i values that can occur for each value of $\{X_1, X_2, \dots, X_{i-1}, X_{i+1}, \dots, X_n\}$.

For ternary relationships ($n = 3$), analysis of four possible cases is necessary: 1:1:1, $M:1:1$, $M:N:1$, $M:N:P$. The article [16] formulates the rules determining the possibility of the occurrence of binary relationships within a ternary relationship. Pursuant to these, cardinalities of imposed binary relationships cannot be lower than the cardinality of the ternary relationship. Therefore, for the first case, the binary relationships of any cardinalities can be imposed. In the fourth case the binary relationships may be only of the many-to-many type.

The above analysis may be also carried out using the theory of functional dependencies. The FDs describing a relationship $R(X_1, X_2, \dots, X_n)$ may be presented as follows:

$$U - \{X_i\} \rightarrow X_i, \quad i = 1, 2, \dots, n \quad , \tag{1}$$

where $U = \{X_1, X_2, \dots, X_n\}$. These dependencies describe the integrity constraints and must not be infringed. They constitute a restriction for $(n-1)$ -ary relationships. Their quantity is equal to the quantity of number one in relationship's cardinality. For a relationship of cardinality 1:1: ... :1 this number is equal to n . If $M_i > 1$ for every i , the dependencies (I) do not occur at all.

The FDs describing the relationships between $(n-1)$ attributes "embedded" in the n -ary relationship may be presented as follows:

$$U - \{X_i, X_j\} \rightarrow X_i, \quad i \neq j, \quad i = 1, 2, \dots, n \quad . \tag{2}$$

Let us denote by F the set of FDs determined by (I). The $(n-1)$ -ary relationship is admissible, if it does not cause any infringement of integrity constraints. This

may occur when a consequence of its introduction will be the occurrence of a new FD in a form determined by (II), which does not belong to F . The attributes, which occur only on the left side of dependencies (II) determine values of other attributes. Thus they belong to every candidate key of R [23]. Imposing the $(n-1)$ -ary relationship with such attributes on the right side of (2) would infringe the data integrity because $U - \{X_i, X_j\} \rightarrow X_i \Rightarrow U - \{X_i\} \rightarrow X_i \notin F$.

Corollary 1. *In the n -ary $R(X_1, X_2, \dots, X_n)$ relationship with F set of functional dependencies between n attributes, determined by formula (II), there may exist a functional dependency $U - \{X_i, X_j\} \rightarrow X_i$ where $i \neq j, i, j = 1, 2, \dots, n$, if attribute X_i does not belong at least to one candidate key of R .*

Example 1. Let us consider a ternary relationship $R(X, Y, Z)$ of $M:1:1$ cardinality. The set F contains two FDs: $XY \rightarrow Z$ and $XZ \rightarrow Y$. Imposition of $Y \rightarrow X$ (or $Z \rightarrow X$) implies $YZ \rightarrow X \notin F$. This means infringement of integrity constraints. Therefore, the dependency $Y \rightarrow X$ (also $Z \rightarrow X$) is inadmissible.

3 Fuzzy Functional Dependencies

In conventional databases the functional dependency $X \rightarrow Y$ between attributes X and Y of the relation scheme R means that every value of attribute X corresponds exactly to one value of attribute Y : $X \rightarrow Y \Leftrightarrow \forall_{t_1, t_2 \in r} t_1(X) = t_2(X) \Rightarrow t_1(Y) = t_2(Y)$, where r is a relation of R and t_1, t_2 denote tuples of r . This corresponds to the assumption that the equality of attributes values may be evaluated formally, using a two-valued logic. In fuzzy databases it is possible to evaluate the degree of the closeness of compared values. The existence of such a dependency means that close values of attribute X correspond to close values of attribute Y .

Let us assume that attribute values are given by means of normal possibility distributions:

$$t(X) = \{\pi_X(x)/x : x \in D_X\}, \quad \sup_{x \in D_x} \pi_X(x) = 1, \tag{3}$$

where D_X is a domain of the attribute X and $\pi_X(x)$ is a possibility degree of $t(X) = x$. The closeness measure of possibility distributions π_1 and π_2 is defined by the formula [22]:

$$=_c (\pi_1, \pi_2) = \sup_x \min(\pi_1(x), \pi_2(x)) \quad . \tag{4}$$

Let $R(X_1, X_2, \dots, X_n)$ be a relation scheme. A relation r is a subset of cartesian product $\Pi(D_1) \times \Pi(D_2) \times \dots \times \Pi(D_n)$, where $\Pi(D_i)$ is a set of possibility distributions of X_i .

Definition 1. [22] *Let $R(U)$ be a relation scheme where $U = \{X_1, X_2, \dots, X_n\}$. Let X and Y be subsets of U : $X, Y \subseteq U$. Y is functionally dependent on X in θ degree, denoted by $X \rightarrow_\theta Y$, if and only if for every relation r of R the following condition is met:*

$$\min_{t_1, t_2 \in r} I(t_1(X) =_c t_2(X), t_1(Y) =_c t_2(Y)) \geq \theta, \tag{5}$$

where $\theta \in [0,1]$, $=_c : [0,1] \times [0,1] \rightarrow [0,1]$ is a closeness measure (4) and $I : [0,1] \times [0,1] \rightarrow [0,1]$ is a fuzzy implicator (Gödel implication):

$$I(a, b) = 1 \quad \text{for } a \leq b, \quad I(a, b) = b \quad \text{for } a > b \quad . \quad (6)$$

In conventional relational databases there exists a sound and complete set of inference rules known as Armstrong’s axioms. These rules can be used to derive new FDs from given ones. For FFDs Armstrong’s axioms have been extended accordingly [22]:

- A1: if $Y \subseteq X$, then $X \rightarrow_\theta Y$ for all θ
- A2: if $X \rightarrow_\theta Y$, then $XZ \rightarrow_\theta YZ$
- A3: if $X \rightarrow_\alpha Y$ and $Y \rightarrow_\beta Z$, then $X \rightarrow_\lambda Z$, $\lambda = \min(\alpha, \beta)$

From A1, A2 and A3 the following inference rules can be derived:

- D1: if $X \rightarrow_\alpha Y$ and $X \rightarrow_\beta Z$, then $X \rightarrow_\lambda YZ$, $\lambda = \min(\alpha, \beta)$
- D2: if $X \rightarrow_\alpha Y$ and $WY \rightarrow_\beta Z$, then $XW \rightarrow_\lambda Z$, $\lambda = \min(\alpha, \beta)$
- D3: if $X \rightarrow_\alpha Y$ and $Z \subseteq Y$, then $X \rightarrow_\alpha Z$
- D4: if $X \rightarrow_\alpha Y$, then $X \rightarrow_\beta Y$ for $\alpha \leq \beta$

Inclusion of fuzzy functional dependencies requires extension of the notion of relation key. Let Γ denotes a set of FFDs for the relation scheme $R(U)$, $U = \{X_1, X_2, \dots, X_n\}$. Its closure (the set of FFDs that are logically implied by Γ) shall be denoted by Γ^+ . Subset K of the set of attributes U ($K \subseteq U$) is a θ -key of R , if dependency $K \rightarrow_\theta U$ belongs to Γ^+ and there is no set $K' \subset K$, such that $K' \rightarrow_\theta U \in \Gamma^+$ (U is fully functionally dependent on K in θ degree). Attributes belonging to a θ -key are called θ -prime-attributes.

Example 2. Consider the relationship $PES(P, E, S)$ between the post held (P), education (E) and salary (S) with the following FFDs: $PE \rightarrow_{0.8} S$ and $SE \rightarrow_{0.6} P$. The scheme PES has two candidate keys: PE - 0.8-key and SE - 0.6-key.

If every θ -nonprime-attribute is fully functionally dependent on a θ -key in α degree, where $\alpha > 0$, then the scheme $R(U)$ is in the θ -fuzzy second normal form (θ -F2NF). Higher normal forms impose on the attributes of R more restrictions. The scheme $R(U)$ is in the third θ -fuzzy normal form (θ -F3NF), if for every FFD $X \rightarrow_\phi Y$, where $X, Y \subseteq U$, $Y \not\subseteq X$, X contains the θ -key of R or Y is a θ -prime-attribute. The θ -fuzzy third normal form eliminates the possible occurrence of transitive dependencies between attributes. Eliminating the possibility that attribute Y in dependency $X \rightarrow_\phi Y$ is θ -prime leads to the definition of θ -fuzzy Boyce-Codd normal form (θ -FBCNF). If a relation scheme is in (θ -FBCNF) then the left side of any FFD contains a θ -key.

Example 3. Scheme PES from the previous example is in 0.6-F3NF (also in 0.6-FBCNF). Let us augment it by attribute A determining age and assume that its values are connected with values of attribute P . Let us assume that the relationship between the post and age is expressed by dependency $P \rightarrow_\phi A$. In result of such modification the scheme is not in 0.6-F3NF. The left side of $P \rightarrow_\phi A$ does not contain a θ -key.

4 Multiargument Relationships in Fuzzy Databases

Let us consider the relationship $R(X_1, X_2, \dots, X_n)$ with the following FFDs:

$$U - \{X_i\} \rightarrow_{\alpha_i} X_i, \quad \text{where } \alpha_i > 0, \quad i = 1, 2, \dots, n \quad . \quad (7)$$

The scheme R has n θ -keys in the form $U - \{X_i\}$. Let us define the following fuzzy sets of attributes:

$$\mathcal{L} = \{(1 - \alpha_1)/X_1, (1 - \alpha_2)/X_2, \dots, (1 - \alpha_n)/X_n\}, \quad (8)$$

$$\mathcal{B} = \{\alpha_1/X_1, \alpha_2/X_2, \dots, \alpha_n/X_n\}. \quad (9)$$

Membership grades of attributes depend on the level of suitable FFDs. If X_i does not occur on the right side of any dependency (7) its degrees of membership in \mathcal{L} and \mathcal{B} are equal to 1 and 0, respectively.

Levels γ_i of functional dependencies:

$$U - \{X_i, X_j\} \rightarrow_{\gamma_i} X_i, \quad \text{where } i \neq j, \quad i, j = 1, 2, \dots, n \quad (10)$$

determining $(n-1)$ -ary relationships cannot exceed relevant values of α_i because of the disturbance of integrity constraints determined by dependencies (7). A consequence of existence of any dependency (10) is one of the existing dependencies (7) with a changed level. For due to the extended Armstrong's rules (A2 and D3) we have: $U - \{X_i, X_j\} \rightarrow_{\gamma_i} X_i \Rightarrow U - \{X_i\} \rightarrow_{\gamma_i} X_i$. This dependency is not contradictory to assumptions if $\gamma_i \leq \alpha_i$. This results from rule D4: if $\gamma_i \leq \alpha_i$, then $U - \{X_i\} \rightarrow_{\alpha_i} X_i \Rightarrow U - \{X_i\} \rightarrow_{\gamma_i} X_i$. Otherwise, the obtained dependency is inadmissible. Its consequence is a change of membership degrees of attributes in sets \mathcal{L} and \mathcal{B} .

Corollary 2. *In the fuzzy n -ary relationship $R(X_1, X_2, \dots, X_n)$ with fuzzy functional dependencies in the form determined by (7), there may exist $(n-1)$ -ary relationships determined by fuzzy functional dependencies (10), in which $\gamma_i \leq \alpha_i$.*

Notice that if the number of FFDs (7) is lower than n , there are attributes fully belonging to \mathcal{L} . They cannot occur on the right side of any fuzzy functional dependency (10).

Scheme $R(X_1, X_2, \dots, X_n)$ with fuzzy functional dependencies (7) occurs in θ -FBCNF, where $\theta = \min_i(\alpha_i)$. After having introduced fuzzy functional dependencies (10) the conditions required by θ -FBCNF may be disturbed. Let us denote by m the number of FFDs (7) occurring in relationship $R(X_1, X_2, \dots, X_n)$ and assume that $m > 1$. These dependencies correspond to the following sets \mathcal{L} and \mathcal{B} :

$$\mathcal{L} = \{(1 - \alpha_1)/X_1, \dots, (1 - \alpha_m)/X_m, 1/X_{m+1}, \dots, 1/X_n\}, \quad (11)$$

$$\mathcal{B} = \{1/X_1, \dots, 1/X_m\} \quad . \quad (12)$$

Attributes X_{m+1}, \dots, X_n fully belong to \mathcal{L} . Let us examine the consequences of imposing the fuzzy functional dependency $U - \{X_i, X_j\} \rightarrow_{\gamma_i} X_i$, where $i \neq j$, i

$= 1, 2, \dots, m, j = 1, 2, \dots, n$ and $\gamma_i \leq \alpha_i$. If $X_j \in \mathcal{B}$, i.e. $j \leq m$, one can remove X_i from dependency $U - \{X_j\} \rightarrow_{\alpha_j} X_j$. Basing on rule D2 we obtain:

$$U - \{X_i, X_j\} \rightarrow_{\gamma_i} X_i \wedge U - \{X_j\} \rightarrow_{\alpha_j} X_j \Rightarrow U - \{X_i, X_j\} \rightarrow_{\lambda_{i,j}} X_j \quad , \quad (13)$$

where $\lambda_{i,j} = \min(\gamma_i, \alpha_j)$. A new key arises, at the level not higher than $\theta' = \max_i(\alpha_i)$. This is formed by attributes occurring on the left side of the introduced dependency. The conditions of the definition of θ -FBCNF have not been disturbed. If in the introduced dependency $X_j \in \mathcal{L}$, i.e. $j > m$, no new key will be formed. Scheme $R(X_1, X_2, \dots, X_n)$ will not occur in θ -FBCNF. The left side of the dependency $U - \{X_i, X_j\} \rightarrow_{\gamma_i} X_i$ does not contain the key. However, it will remain in θ -F3NF, because X_i is θ -prime.

With one fuzzy functional dependency (7) ($m = 1$) there is only one candidate key. Having introduced a new dependency, the scheme $R(X_1, X_2, \dots, X_n)$ will not occur in θ -F3NF, because the left side does not contain the θ -key, and X_i is not θ -prime. Furthermore, a partial dependency of attribute X_i on the θ -key will arise, which means a disturbance in the conditions defining the θ -fuzzy second normal form.

Let us consider the ternary relationship $R(X, Y, Z)$ with the following FFDs:

$$XY \rightarrow_{\alpha} Z, \quad XZ \rightarrow_{\beta} Y, \quad YZ \rightarrow_{\gamma} X, \quad \text{where } \alpha > 0, \beta > 0, \gamma > 0. \quad (14)$$

Let us assume that $\alpha > \beta > \gamma$. Scheme $R(X, Y, Z)$ has three candidate keys at the $\theta \leq \gamma$ level, two at the $\theta \leq \beta$ level and one at the $\theta \leq \alpha$ level. The sets \mathcal{L} and \mathcal{B} are as follows:

$$\mathcal{L} = \{(1 - \alpha)/X, (1 - \beta)/Y, (1 - \gamma)/Z\} \quad , \quad (15)$$

$$\mathcal{B} = \{\alpha/X, \beta/Y, \gamma/Z\} \quad . \quad (16)$$

The possibility to impose fuzzy binary relationships is limited by values of α , β and γ . The values determining membership degrees of attributes in sets \mathcal{L} and \mathcal{B} cannot be changed due to the introduced new functional dependency. Its level cannot be arbitrary. Let us check the possible existence of the dependency $Y \rightarrow_{\phi} X$. The rules A2 and D3 yield the dependency $YZ \rightarrow_{\phi} X$. Integrity constraints will not be disturbed if $\phi \leq \gamma$, because then $YZ \rightarrow_{\gamma} X \Rightarrow YZ \rightarrow_{\phi} X$ (rule D4). Let us notice that the dependency $Y \rightarrow_{\phi} X$, where $\phi \leq \alpha$, implies dependency $Y \rightarrow_{\phi} Z$. For we have: $Y \rightarrow_{\phi} X \wedge XY \rightarrow_{\alpha} Z \Rightarrow Y \rightarrow_{\phi} Z$.

Example 4. Let us return to scheme *PES* from Example 2. The sets \mathcal{L} and \mathcal{B} are as follows: $\mathcal{L} = \{0.4/P, 1/E, 0.2/S\}$ and $\mathcal{B} = \{0.6/P, 0.8/S\}$. Let us assume that between attributes P and S there should be dependency $E \rightarrow_{\phi} S$, where $\phi = 0.5$. Integrity conditions are not disturbed. Sets \mathcal{L} and \mathcal{B} are not changed. By using rule D2 we get $E \rightarrow_{0.5} P$. Attribute E becomes a θ -key with $\phi = 0.5$. Scheme *PES* is in 0.5-FBCNF. If, however, $\phi = 0.9$, the dependency $PE \rightarrow_{0.9} S$ is forced, which is contradictory to the assumption. Changed are the sets \mathcal{L} and \mathcal{B} : $\mathcal{L} = \{0.4/P, 1/E, 0.1/S\}$ and $\mathcal{B} = \{0.6/P, 0.9/S\}$. The similar situation would occur when dependency $E \rightarrow_{\phi} P$ were imposed.

5 Conclusions

The subject of the paper is an analysis of multiargument relationships $R(X_1, X_2, \dots, X_n)$ in fuzzy databases by means of functional dependencies. Apart from dependencies (7) between all n attributes there may be also dependencies describing relationships of fewer attributes (10). However, there is no complete arbitrariness. The $(n-1)$ -ary relationships embedded in n -ary relationship must not be contrary to it. Fuzzy functional dependencies (10) are admissible if their levels does not exceed the levels of relevant dependencies determined by formula (7). The dependencies between $(n-1)$ attributes may disturb fuzzy normal forms of scheme R .

References

1. Motro, A.: Imprecision and uncertainty in database systems. In: Bosc, P., Kacprzyk, J. (eds.) *Studies in Fuzziness: Fuzziness in Database Management Systems*, pp. 3–22. Physica Verlag, Heidelberg (1995)
2. Dey, D., Sarkar, S.: Generalized Normal Forms for Probabilistic Relational Data. *IEEE Transactions on Knowledge and Data Engineering* 14(3), 485–497 (2002)
3. Motro, A.: Accommodating Imprecision in Database Systems: Issues and Solutions. *SIGMOD RECORD* 19(4), 69–74 (1990)
4. Petry, F.: *Fuzzy Databases: Principles and Applications*. Kluwer Academic Publishers, Boston (1996)
5. Fedrizzi, M., Kacprzyk, J.: A brief introduction to fuzzy sets. In: Bosc, P., Kacprzyk, J. (eds.) *Studies in Fuzziness: Fuzziness in Database Management Systems*, pp. 59–67. Physica Verlag, Heidelberg (1995)
6. Zadeh, L.A.: Fuzzy sets. *Information and Control* 8, 338–353 (1965)
7. Galindo, J., Urrutia, A., Piattini, M.: *Fuzzy Databases. Modeling, Design and Implementation*. Idea Group Publishing, London (2005)
8. van Gyseghem, N., de Caluwe, R.: Imprecision and uncertainty in UFO database model. *Journal of the American Society for Information Science* 49(3), 236–252 (1998)
9. Bordogna, G., Pasi, G., Lucarella, D.: A Fuzzy Object-Oriented Data Model for Managing Vague and Uncertain Information. *International Journal of Intelligent Systems* 14, 623–651 (1999)
10. Ma, Z.M., Zhang, W.J., Ma, W.Y.: Extending object-oriented databases for fuzzy information modeling. *Information Systems* 29, 421–435 (2004)
11. Yazici, A., George, R.: *Fuzzy Database Modeling*. Physica-Verlag, Heidelberg (1999)
12. Buckles, B.P., Petry, F.E.: A Fuzzy Representation of Data for Relational Database. *Fuzzy Sets and Systems* 7, 213–226 (1982)
13. Prade, H., Testemale, C.: Generalizing Database Relational Algebra for the Treatment of Incomplete or Uncertain Information and Vague Queries. *Information Science* 34, 115–143 (1984)
14. Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems* 1, 3–28 (1978)
15. Dullea, J., Song, I., Lamprou, I.: An Analysis of Structural Validity in Entity-Relationship Modeling. *Data and Knowledge Engineering* 47, 167–205 (2001)

16. Jones, T., Song, I.: Analysis of Binary/Ternary Relationships Cardinality Combinations in Entity-Relationship Modeling. *Data and Knowledge Engineering* 19, 39–64 (1996)
17. Raju, K.V.S.V.N., Majumdar, A.K.: Fuzzy Functional Dependencies and Loss-less Join Decomposition of Fuzzy Relational Database Systems. *ACM Trans. On Database Systems* 13, 120–166 (1988)
18. Sheno, S., Melton, A., Fan, L.T.: Functional dependencies and normal forms in the fuzzy relational database model. *Information Sciences* 60, 1–28 (1992)
19. Cubero, J.C., Vila, M.A.: A new definitions of fuzzy functional dependency in fuzzy relational databases. *International Journal for Intelligent Systems* 9, 441–448 (1994)
20. Sozat, M.I., Yazici, A.: A Complete Approximation for Fuzzy Functional and Multi-valued Dependencies in Fuzzy Database Relations. *Fuzzy Sets and Systems* 117(2), 161–181 (2001)
21. Tyagi, B.K., Sharfuddin, A., Dutta, R.N., Devendra, K.T.: A complete axiomatization of fuzzy functional dependencies using fuzzy function. *Fuzzy Sets and Systems* 151, 363–379 (2005)
22. Chen, G.: *Fuzzy Logic in Data Modeling - semantics, constraints and database design*. Kluwer Academic Publishers, Boston (1998)
23. Saiedian, H., Spencer, K.: An Efficient Algorithm to Compute the Candidate Key of a Relational Database Schema. *The Computer Journal* 39, 124–132 (1996)

Methods of Evaluating Degrees of Truth for Linguistic Summaries of Data: A Comparative Analysis

Adam Niewiadomski and Oskar Korczak

Institute of Information Technology,
Technical University of Łódź, Poland
aniewiadomski@ics.p.lodz.pl
<http://edu.ics.p.lodz.pl>

Abstract. The paper, like some of our previous publications, focuses on linguistic summaries of databases in the sense of Yager [1], with further modifications by Kacprzyk and Yager [2]. In particular, we are interested in some alternative methods of evaluating **degrees of truth, DTs**, for linguistic summaries. The considered methods, for instance, Sugeno integral, GD method, or MVCP, are alternatives for „traditional” DTs, by Zadeh [3]. Our original contribution is an analysis of these methods and their interpretation in terms of linguistic summarization of databases. Especially, different DTs for a summary are evaluated and the computational cost is assessed. Based on that, sensitivity of the methods to different parameters of linguistic summaries, e.g. increasing number of qualifiers or the first/the second form, single/composed summarizers, relative/absolute/coherent quantifiers, are examined.

Keywords: Linguistically quantified statements, linguistic summaries of data, degrees of truth, Sugeno integral, GD method, MVCP method.

1 Motivation and Preliminaries

The Calculus of Linguistically Quantified Statements based on fuzzy sets, was proposed by Zadeh in 1983 [3]. It is commonly known that *existential*, \exists , and *general (universal)*, \forall , quantifiers extend the classic (two-valued) calculus of predicates. Similarly, linguistic statements can be quantified by *linguistic quantifiers*, e.g. LESS THAN HALF. There are two forms of linguistically quantified statements in the sense of Zadeh. *The first form:*

$$Q \text{ objects are } S_1 \tag{1}$$

e.g. MANY *cars are expensive* (also referred as Q^I), and *the second form*

$$Q \text{ objects being } S_2 \text{ are } S_1 \tag{2}$$

e.g. MANY *well-equipped cars are expensive* (referred as Q^{II}) [3]. In terms of fuzzy logic, Q is a linguistic quantifier – a linguistic expression for quantity (*a quantity*

in agreement), modeled by a normal¹ and convex² fuzzy set in $\mathcal{X}_Q \subseteq \mathbb{R}^+ \cup \{0\}$. S_1 and S_2 are labels represented by fuzzy sets in finite universes of discourse \mathcal{X}_{S_1} , \mathcal{X}_{S_2} . Hence, in the examples above, one may assign: MANY=Q, expensive= S_1 , and well-equipped= S_2 .

The **degree of truth**, DT, of (III) is evaluated via Σ counts, cf. [4]:

$$T(Q \text{ objects are } S_1) = \mu_Q \left(\frac{\Sigma\text{count}(S_1)}{M} \right) \tag{3}$$

where $M = \Sigma\text{count}(\mathcal{X}_{S_1})$ for relative Q, and $M = 1$ for absolute. And of (2):

$$T(Q \text{ objects being } S_2 \text{ are } S_1) = \mu_Q \left(\frac{\Sigma\text{count}(S_1 \cap S_2)}{\Sigma\text{count}(S_2)} \right) \tag{4}$$

In case $\mathcal{X}_{S_1} \neq \mathcal{X}_{S_2}$, we consider $S_1 \cap S_2$ as the intersection of cylindric extensions of S_1 and S_2 to $\mathcal{X}_{S_1} \times \mathcal{X}_{S_2}$.

Coherent family of relative fuzzy quantifiers is the family of normal and convex fuzzy sets $\{Q_1, \dots, Q_n\}$ in $[0, 1]$ which fulfil the following conditions:

$$1) Q_1 = \forall \quad 2) Q_n = \exists \quad 3) \forall_{i=1, \dots, n-1} \forall_{x \in [0,1]} \mu_{Q_i}(x) \leq \mu_{Q_{i+1}}(x) \tag{5}$$

It must be noticed that all the quantifiers in a coherent family are represented by non-decreasing membership functions. Besides, the traditional quantifiers \forall and \exists are represented by the fuzzy sets: $\{1.0/1\}$ and $\int_{x \in (0,1]} 1.0/x$, respectively.

Linguistic Summaries of Databases. Let $\mathcal{Y} = \{y_1, \dots, y_m\}$ be a set of objects, e.g. cars. Let V_1, \dots, V_n be attributes manifested by the objects, e.g. age. Let $\mathcal{X}_1, \dots, \mathcal{X}_n$ be the domains of V_1, \dots, V_n , respectively. We denote a value of V_j for object y_i as $V_j(y_i)$, $i \leq m$, $j \leq n$, e.g. let $V_j = \text{Age}$, $y_i = \text{'Honda'}$, $V_j(y_i) = 15$, where $15 \in \mathcal{X}_j$. Hence, the database \mathcal{D} , collecting information on y 's, is $\mathcal{D} = \{d_1, \dots, d_m\}$, and $d_i = \langle V_1(y_i), \dots, V_n(y_i) \rangle \in \mathcal{X}_1 \times \dots \times \mathcal{X}_n$, $i = 1, \dots, m$, is the tuple describing y_i .

The general form of the linguistic summary considered by Yager, is

$$Q \text{ } P \text{ are/have } S [T] \tag{6}$$

where Q is a linguistic quantifier, P is the subject of the summary, S is the so-called *summarizer* represented by a fuzzy set, and $T \in [0, 1]$ is a *degree of truth*. In particular, we use the so-called *compound summarizers* S represented by a few fuzzy sets modeling properties S_1, \dots, S_n related to V_1, \dots, V_n , respectively. Thus, its membership function $\mu_S: \mathcal{X}_1 \times \dots \times \mathcal{X}_n \rightarrow [0, 1]$ is given as $\mu_S(d_i) = \mu_{S_1}(d_i) t \dots t \mu_{S_n}(d_i)$, $i = 1, 2, \dots, m$, where t is a t-norm which represents the connective AND. Thus,

$$T(Q \text{ } P \text{ are/have } S_1 \text{ AND } \dots \text{ AND } S_n) = \mu_Q \left(\frac{\sum_{i=1}^m \mu_S(d_i)}{M} \right) \tag{7}$$

¹ A fuzzy set A in \mathcal{X} is normal iff $\sup_{x \in \mathcal{X}} \mu_A(x) = 1$.

² A in $\mathcal{X} \subseteq \mathbb{R}$ is convex iff $\forall_{\alpha \in (0,1]} \forall_{r,s \in A_\alpha} \forall_{\lambda \in [0,1]} \lambda r + (1 - \lambda)s \in A_\alpha - \alpha\text{-cut of } A$.

where $M = m$ for a relative Q , or $M = 1$ for an absolute Q . T is a real number from the interval $[0, 1]$, and it is interpreted as the degree of truth.

Another form of a summary – *summary with a qualifier*, is based on (2):

$$Q \text{ } P \text{ being/having } W \text{ are/have } S [T] \tag{8}$$

where W is *qualifier* – a property of the objects, represented by fuzzy sets W_{g_1}, \dots, W_{g_x} , $x \leq n$, $x \in \mathbb{N}$, $g_1, \dots, g_x \in \{1, \dots, n\}$, in $\mathcal{X}_{g_1}, \dots, \mathcal{X}_{g_x}$, respectively, so $\mu_W(d_i) = \mu_{W_{g_1}}(d_i) t \dots t \mu_{W_{g_x}}(d_i)$, $i = 1, \dots, m$. The DT of (8) is evaluated as:

$$T(Q \text{ } P \text{ being/having } W \text{ are/have } S) = \mu_Q \left(\frac{\sum_{i=1}^m (\mu_S(d_i) t \mu_W(d_i))}{\sum_{i=1}^m \mu_W(d_i)} \right) \tag{9}$$

In (9), only the relative quantification is possible, because the sum of membership degrees to $S \cap W$ is related to the sum of membership degrees to W .

2 Alternative Degrees of Truth of Linguistic Summaries

Apart from formulae for DTs given by Zadeh, i.e. (3) and (4), there exist a few other methods, described by the authors of (5). Some of them have already been applied to linguistic summaries, e.g. Sugeno integral, see (6).

Our original contribution is to analyze the usefulness of these methods from the point of view of linguistic summaries. Degrees of truth of linguistically quantified statements in the first and the second form are now interpreted in terms of linguistic summarization, see Eqs. (6)–(9), and presented in the two following subsections. Then, in Sec. 3, experiments are presented: the comparison of the discussed methods is done and their computational costs are estimated.

2.1 Degrees of Truth for Summaries in the First Form

Yager’s Method Based on OWA Operator/Choquet Integral based on Ordered Weighted Averaging Operators (7), works only for coherent (i.e. non-decreasing) quantifiers, see (5). This DT is denoted $T_{Y_{Q_I}}$ and evaluated as:

$$T_{Y_{Q_I}} = \sum_{i=1}^m w_i \cdot b_i \tag{10}$$

where b_i is the i -th largest value of μ_S , and w_i can be found from: $w_i = \mu_Q \left(\frac{i}{m} \right) - \mu_Q \left(\frac{i-1}{m} \right)$ where $i = 1, 2, \dots, m$, and $\mu_Q(0) = 0$. This method is equivalent to the Choquet-integral-based method, for a specific case of a fuzzy measure, cf. (5).

The Method Based on Sugeno Integral is another method which works only for non-decreasing quantifiers (5). The degree of truth is denoted $T_{S_{Q_I}}$:

$$T_{S_{Q_I}} = \max_{1 \leq i \leq m} \min \left(\mu_Q \left(\frac{i}{m} \right), b_i \right) \tag{11}$$

where b_i – as in (10).

Cardinality-Based Methods G. The family G of methods based on the cardinality approach takes into account absolute quantifiers [5]:

$$T_{G_{Q_I}} = (E(S, 0) \ t \ \mu_Q(0)) \ s \ (E(S, 1) \ t \ \mu_Q(1)) \ s \dots \ s \ (E(S, m) \ t \ \mu_Q(m)) \quad (12)$$

as far as relative:

$$T_{G_{Q_I}} = \left(E(S, 0) \ t \ \mu_Q\left(\frac{0}{m}\right) \right) \ s \dots \ s \ \left(E(S, m) \ t \ \mu_Q\left(\frac{m}{m}\right) \right) \quad (13)$$

in which $E(S, i)$, $i = 1, \dots, m$, denotes possibility that exactly i elements from \mathcal{D} belong to S [3]: $E(S, i) = L(S, i) \ t \ (1 - L(S, i + 1))$ and $L(S, i)$ denotes possibility that at least i elements from \mathcal{D} belong to S [4]:

$L(S, i) = \mu_S(d_{p_1}) \ s \dots \ s \ (\mu_S(d_{p_1}) \ t \dots \ t \ \mu_S(d_{p_i}))$, $1 \leq i \leq m$, or 1 for $i = 0$, or 0 for $i > m$ [5]. I_i is a set of indices of tuples: $I_i = \{(p_1, \dots, p_i) : p_1 < \dots < p_i, p_q \in \{1, \dots, m\}, q \in \{1, \dots, i\}\}$ where t is a t -norm, and s is a t -conorm.

Method GD is one of G methods [5] and it works for absolute quantifiers:

$$T_{GD_{Q_I}} = \sum_{i=0}^m ED(S, i) \cdot \mu_Q(i), \quad (14)$$

as far as for relative: $T_{GD_{Q_I}} = \sum_{i=0}^m ED(S, i) \cdot \mu_Q\left(\frac{i}{m}\right)$, where $ED(S, i) = b_i - b_{i+1}$ with $b_0 = 1$, $b_{m+1} = 0$.

2.2 Degrees of Truth for Summaries in the Second Form

Yager’s Method Based on OWA Operator is the first of presented DTs for linguistic summaries in the form of (2), i.e. Q^{II} . It is applied for non-decreasing quantifiers, see (5), cf. [5]. We denote this DT as $T_{Y_{Q^{II}}}$ and evaluate as:

$$T_{Y_{Q^{II}}} = \sum_{i=1}^m w_i \cdot c_i, \quad (15)$$

where c_i is the i -th largest value of the membership function to the fuzzy set $W^c \vee S$, where $\mu_{W^c}(x) = 1 - \mu_W(x)$ (the complement of a fuzzy set). The weights w_i are evaluated as $w_i = \mu_Q(H_i) - \mu_Q(H_{i-1})$, $i = 1, \dots, m$, $H_0 = 0$, $H_i = \frac{\sum_{j=1}^i e_j}{\sum_{k=1}^m e_k}$ where e_k is the k -th smallest membership degree to W .

Method of Vila, Cubero, Medina, Pons (MVCP) uses the so-called *orness* measure, defined by Yager for non-decreasing coherent quantifiers [7]:

$$orness(Q) = \sum_{i=1}^m \left(\frac{m-i}{m-1} \right) \cdot \left(\mu_Q\left(\frac{i}{m}\right) - \mu_Q\left(\frac{i-1}{m}\right) \right) \in [0, 1] \quad (16)$$

³ In particular, S and \mathcal{D} can be given in different domains, so one should consider belonging of d_i , $i = 1, \dots, m$ to the cylindrical extension of S to $\mathcal{X}_1 \times \dots \times \mathcal{X}_n$.

⁴ See Footnote [3].

The *orness* measure says how close to the $\exists \int_{x \in (0,1)} 1.0/x$ quantifier the analysed Q is placed, so $orness(\exists) = 1$, $orness(\forall) = 0$. $T_{V_{Q^{II}}}$ is evaluated as:

$$T_{V_{Q^{II}}} = orness(Q) \cdot \max_{d \in \mathcal{D}} \{ \mu_W(d) \wedge \mu_S(d) \} + (1 - orness(Q)) \cdot \min_{d \in \mathcal{D}} \{ \mu_S(d) \vee (1 - \mu_W(d)) \} \quad (17)$$

Generalization of the GD Method for linguistic summaries in the second form, is now presented [5]:

$$T_{GD_{Q^{II}}} = \sum_{c \in CR(\frac{S}{W})} ER\left(\frac{S}{W}, c\right) \cdot \mu_Q(c) \quad (18)$$

Besides, $ER\left(\frac{S}{W}, c\right) = \sum_{c=C(\frac{S}{W}, \alpha_i)} (\alpha_i - \alpha_{i+1})$, $\forall_{c \in CR(\frac{S}{W})}$, in which $C\left(\frac{S}{W}, \alpha_i\right) = \frac{|(S \cap W)_{\alpha_i}|}{|W_{\alpha_i}|}$ and

$$CR\left(\frac{S}{W}\right) = \left\{ \frac{|(S \cap W)_{\alpha}|}{|W_{\alpha}|} \right\}, \alpha \in M\left(\frac{S}{W}\right) \quad (19)$$

Moreover, $M\left(\frac{S}{W}\right)$ denotes: $M\left(\frac{S}{W}\right) = M(S \cap W) \cup M(W)$, and $M(W) = \{ \alpha \in [0, 1] : \exists_{d_i \in \mathcal{D}} \mu_W(d_i) = \alpha \}$.

ZS – Possibilistic Method of evaluating degrees of truth for summaries in the second form is now shown [5]:

$$T_{ZS_{Q^{II}}} = \max_{c \in CR(\frac{S}{W})} \min\left(ES\left(\frac{S}{W}, c\right), \mu_Q(c)\right), \quad (20)$$

where $CR\left(\frac{S}{W}\right)$ is given by (19). And relative cardinality ES :

$$ES\left(\frac{S}{W}, c\right) = \max\left\{ \alpha \in M\left(\frac{S}{W}\right) : c = \frac{|(S \cap W)_{\alpha}|}{|W_{\alpha}|} \right\}, \forall_{c \in CR(\frac{S}{W})} \quad (21)$$

where $M\left(\frac{S}{W}\right)$ is given in the previous paragraph.

3 Experiments and Results

The experiments answer whether the analysed methods of evaluating degrees of truth are able to reflect different combinations of quantifiers Q with increasing/decreasing numbers of summarizers S and/or qualifiers W . Table 1 enumerates methods and related quantifiers which are examined. The database chosen to this experiment consists of 480 tuples describing measured parameters of a climate [8]. Sample attributes and their linguistic values are Season and Winter, Temperature and Icily, or Sea Level Pressure and ModerateToHigh.

For each experiment, a summary with fixed S and W is created. Next, each summary is combined with a quantifier of related type, and degrees of truth via the methods described in Sec. 2, are evaluated. The results are collected in Tables 2 ÷ 5. Table 6 presents computational cost for the examined methods.

Table 1. Possible quantifiers for linguistic summaries of type Q^I and Q^{II}

Method Q^I	Absolute	Relative	Coherent
Zadeh	YES	YES	YES
Yager (OWA)/Choquet	no	no	YES
Sugeno	no	no	YES
G	YES	YES	YES
GD	YES	YES	YES
Method Q^{II}	Absolute	Relative	Coherent
Zadeh	no	YES	YES
Yager (OWA)	no	no	YES
MVCP	no	no	YES
GD	YES	YES	YES
ZS	YES	YES	YES

Table 2. Degrees of truth in Experiment 1: Linguistic summary Q^I $S_1 \rightarrow$ Time = 60's, $S_2 \rightarrow$ Season = Winter, $S_3 \rightarrow$ Temperature = Icily

Method	Quantifier – Relative				Quantifier – Coherent			
	Few	Couple	Moderate	Many	All	Most	At least half	Exists
Yager (OWA)	-	-	-	-	0	0.05	0.101	1
Choquet	-	-	-	-	0	0.05	0.101	1
Sugeno	-	-	-	-	0	0.002	0.004	1
G	0.389	0	0	0	0	0.065	0.125	0.5
GD	0.35	0	0	0	0	0.052	0.105	1
Zadeh	0.35	0	0	0	0	0.053	0.105	1

Table 3. Degrees of truth in Experiment 2: Linguistic summary Q^I , $S_1 \rightarrow$ Time = 60's $S_2 \rightarrow$ Season = Winter, $S_3 \rightarrow$ Temperature = Icily $S_4 \rightarrow$ Sea Level Pressure = ModerateToHigh $S_5 \rightarrow$ Humidity = QuiteDamp

Method	Quantifier – Relative				Quantifier – Coherent			
	Few	Couple	Moderate	Many	All	Most	At least half	Exists
Yager (OWA)	-	-	-	-	0	0.008	0.016	0.72
Choquet	-	-	-	-	0	0.008	0.016	0.72
Sugeno	-	-	-	-	0	0.002	0.004	0.72
G	0.148	0	0	0	0	0.029	0.058	0.556
GD	0.061	0	0	0	0	0.009	0.018	0.72
Zadeh	0.061	0	0	0	0	0.009	0.018	1

Table 4. Degrees of truth in Experiment 3: Linguistic summary $Q^{II} W \rightarrow$ Sea Level Pressure = ModerateToHigh $S_1 \rightarrow$ Time = 60's, $S_2 \rightarrow$ Season = Winter, $S_3 \rightarrow$ Temperature = Icily

Method	Quantifier – Relative				Quantifier – Coherent			
	Few	Couple	Moderate	Many	All	Most	At least half	Exists
Yager (OWA)	-	-	-	-	0	0.222	0.416	1
MVCP	-	-	-	-	0	0.498	0.748	1
GD	0.078	0	0.38	0	0.04	0.252	0.464	0.52
ZS	0.083	0	0.51	0	0.133	0.5	0.51	0.51
Zadeh	0.291	0	0	0	0	0.044	0.087	1

Table 5. Degrees of truth in Experiment 4: Linguistic summary $Q^{II}: W_1 \rightarrow$ Sea Level Pressure = ModerateToHigh, $W_2 \rightarrow$ Relative Humidity = QuiteDamp, $S_1 \rightarrow$ Time = 60's $S_2 \rightarrow$ Season = Winter, $S_3 \rightarrow$ Temperature = Icily

Method	Quantifier – Relative				Quantifier – Coherent			
	Few	Couple	Moderate	Many	All	Most	At least half	Exists
Yager (OWA)	-	-	-	-	0	0.383	0.603	1
MVCP	-	-	-	-	0	0.358	0.539	0.72
GD	0	0	0	0	0.02	0.02	0.02	0.02
ZS	0	0	0	0	0	0	0	0
Zadeh	0.261	0	0	0	0	0.039	0.078	1

Table 6. Approximated pessimistic computational cost of the examined methods

Method Q^I	Computational cost	Method Q^{II}	Computational cost
Yager (OWA)	$O(n \log n)$	Yager (OWA)	$O(n \log n)$
G	$O(n \log n)$	ZS	$O(n \log n)$
GD	$O(n \log n)$	GD	$O(n \log n)$
Choquet	$O(n \log n)$	MVCP	O(n)
Sugeno	$O(n \log n)$		
Zadeh Q^I	$O(n)$	Zadeh Q^{II}	$O(n)$

4 Final Observations

Experiments 1 – 2 (for Q^I):

1. The changing number of summarizers in the G, GD, Choquet and Yager (OWA) methods affect their results, similarly to the Zadeh method. The more summarizers S_1, S_2, \dots , the smaller DT, which is related to properties

of the t -norm operator used. However, they are slower than the Zadeh original method.

2. Each of the methods applied with $Q_{\exists} = \exists = \int_{x \in (0,1]} 1.0/x$ shows whether there exist records corresponding to a summary in a summarized database.
3. Yager's method based on OWA operator applied with quantifiers $Q_{\forall} = \forall = \{1.0/1\}$ and for $Q_{\exists} = \exists$, is able to indicate the lowest and the largest membership degree to based on summarizer S , in a linguistic summary in the form of Q^I , respectively. The lowest degree is b_m and the largest is b_0 , where: $m = |X|$.

Experiments 3 – 4 (for Q^{II}):

1. The changing number of summarizers in MVCP, GD, and ZS methods does not affect the degree of truth.
2. MVCP, GD, ZS are affected by numbers of qualifiers: the more qualifiers in a summary, the smaller its DT.
3. MVCP seems to be the most sensitive method. It means that each extension of the set of qualifiers affects the DT.
4. **MVCP is the fastest method alternative for Zadeh's Q^{II} , its computational cost equals $O(n)$.**
5. Increasing/decreasing number of summarizers or qualifiers in Yager's OWA method affects the result: the more summarizers, the smaller DT.

References

1. Yager, R.R.: A new approach to the summarization of data. *Information Sciences* 28, 69–86 (1982)
2. Kacprzyk, J., Yager, R.R.: Linguistic summaries of data using fuzzy logic. *International Journal of General Systems* 30, 133–154 (2001)
3. Zadeh, L.A.: A computational approach to fuzzy quantifiers in natural languages. *Computers and Maths with Applications* 9, 149–184 (1983)
4. De Luca, A., Termini, S.: A definition of the non-probabilistic entropy in the setting of fuzzy sets theory. *Information and Control* 20, 301–312 (1972)
5. Delgado, M., Sanchez, D., Vila, M.A.: Fuzzy cardinality based evaluation of quantified sentences. *International Journal of Approximate Reasoning* 23, 23–66 (2000)
6. Kacprzyk, J., Wilbik, A., Zadrozny, S.: Linguistic summaries of time series via a quantifier based aggregation using the Sugeno integral. In: 2006 IEEE International Conference on Fuzzy Systems, pp. 713–719 (2006)
7. Yager, R.R.: On ordered weighted averaging operators in multicriteria decision making. *IEEE Transactions on Systems, Man and Cybernetics* 18, 183–190 (1988)
8. http://nsidc.org/data/docs/noaa/g02141_esdimmet/

On Non-singleton Fuzzification with DCOG Defuzzification

Robert K. Nowicki and Janusz T. Starczewski*

¹ Academy of Management (SWSPiZ), Institute of Information Technology,
ul. Sienkiewicza 9, 90-113 Łódź, Poland

² Department of Computer Engineering, Czestochowa University of Technology,
Al. Armii Krajowej 36, 42-200 Czestochowa, Poland
{robert.nowicki,janusz.starczewski}@kik.pcz.pl

Abstract. The paper discusses the non-singleton fuzzification. We prove that for logical-type fuzzy systems with DCOG defuzzification, the non-singleton fuzzification can be implemented in a singleton fuzzification architecture.

1 Introduction

The fuzzy systems are popular techniques in many tasks like approximation, prediction, control and classification. The fuzzy system is composed of a few blocks which realize separate operations (fuzzification, reasoning, aggregation and defuzzification) and a block with knowledge database in form of fuzzy rules. In literature [1], [9], [16] fuzzy systems have been presented with various defuzzification methods, reasoning methods, form of rules and types of fuzzification.

The published results of researches specify the scope of usefulness of individual solutions, e.g. Mamdani reasoning is more appropriate to approximation and modeling tasks, and logical reasoning is more appropriate to classification tasks [14], [15]. The full implementation of a fuzzy system could be simplified by using one of defuzzification methods, which allows the fuzzy system to be assembled from elementary functional elements linked by real value connections. The non-singleton fuzzification usually leads to implementation problems. In this paper, we show that fuzzy systems with implication reasoning and employing DCOG defuzzification can be modeled without any change of the singleton system architecture.

2 Logical-Type Fuzzy Systems

2.1 Fuzzification

Fuzzification is the process mapping from real input space $\mathbf{X} \subset \mathbb{R}^n$ to fuzzy sets defined in \mathbf{X} . The singleton fuzzification is the most frequently used method of

* This work was partly supported by Polish Ministry of Science and Higher Education (Habilitation Projects N N516 372234 2008–2011 and N N514 414134 2008–2011).

fuzzification. This one maps real input values $\bar{\mathbf{x}} = [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n]$ into fuzzy set $A' \subseteq \mathbf{X}$ with following membership function

$$\mu_{A'}(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} = \bar{\mathbf{x}} \\ 0 & \text{if } \mathbf{x} \neq \bar{\mathbf{x}} \end{cases} \tag{1}$$

The alternative to the singleton fuzzification is a non-singleton fuzzification which maps $\bar{\mathbf{x}}$ into any other fuzzy set with membership function which takes value 1 when $\mathbf{x} = \bar{\mathbf{x}}$, e.g. Gaussian defined by

$$\mu_{A'}(\mathbf{x}) = \exp \left[- \left(\frac{x - \bar{x}}{\sigma} \right)^2 \right], \tag{2}$$

where σ describes width of the set. The motivation to choose the non-singleton fuzzification is dictated by taking into consideration the imprecision of the measurements or using a fuzzy system in the case of noisy data [6], [11], [13].

2.2 Rule Base and Implications

In fuzzy systems, the following form of rules is applied:

$$R^r: \mathbf{IF} \ x_1 \text{ is } A_1^r \text{ AND } x_2 \text{ is } A_2^r \text{ AND } \dots \tag{3}$$

$$\dots \text{ AND } x_n \text{ is } A_n^r \ \mathbf{THEN} \ y \text{ is } B^r$$

where $A^r = A_1^r \times \dots \times A_n^r$ is an antecedent fuzzy set, and B^r is a consequent fuzzy set, both used in the r -th rule.

As it was mentioned in Section 1, there are two approaches to design fuzzy inference systems. In the case of the Mamdani approach, function I takes the form

$$I(a, b) = T\{a, b\} \tag{4}$$

where T is any t-norm.

However in this paper, we deal with extensions of two-valued implications, called genuine fuzzy implications. They are used in the logical approach [1], [9], [12]. We can enumerate some groups of genuine fuzzy implications [5]:

- S-implications

$$I(a, b) = S\{N\{a\}, b\}. \tag{5}$$

The Lukasiewicz, Reichenbach, Kleene-Dienes, Fodor and Dubois-Prade implications are known examples of S-implications.

- R-implications

$$I(a, b) = \sup_{z \in [0, 1]} \{z | T\{a, z\} \leq b\}. \tag{6}$$

The Rescher, Goguen and Gödel implications are examples of R-implications.

- QL-implications

$$I(a, b) = S\{N\{a\}, T\{a, b\}\}. \tag{7}$$

The Zadeh implication is an example of QL-implications.

In formulas (5)-(7) $a, b \in [0, 1]$, T is any t-norm, S is any t-conorm, N is any fuzzy negation. In the paper, we use negations N being involutive, i.e. $N(N(a)) = a \forall a \in [0, 1]$, such that T and S are dual with respect to N with both T and S being continuous.

The result of a fuzzy reasoning process is the conclusion in the form

$$y \text{ is } B', \tag{8}$$

where B' is aggregated from conclusions B'^r obtained as results of a fuzzy reasoning process for separated rules R^r , $r = 1, \dots, N$. Namely, $B'^r = A' \circ (A^r \mapsto B^r)$ are the fuzzy sets with the membership function defined using *sup* – T composition, i.e.

$$\mu_{B'^r}(y) = \sup_{\mathbf{x} \in \mathbf{X}} T \{ \mu_{A'}(\mathbf{x}), I(\mu_{A^r}(\mathbf{x}), \mu_{B^r}(y)) \}. \tag{9}$$

The aggregation method depends strictly on a reasoning method. In the case of using S-implication, R-implication and QL-implication, i.e. in logical approach, we have $B' = \bigcap_{r=1}^R B'^r$, hence

$$\mu_{B'}(y) = \bigwedge_{r=1}^N \mu_{B'^r}(y). \tag{10}$$

Only in the case of the singleton fuzzification (see (11)), equation (9) comes down to the following form

$$\mu_{B'^r}(y) = I(\mu_{A^r}(\bar{\mathbf{x}}), \mu_{B^r}(y)). \tag{11}$$

2.3 Defuzzification

In literature various neuro-fuzzy systems have been proposed [3, 4, 7-8, 10-15]. One of the most important elements (besides implication) determining the architecture of such systems is defuzzification. In the sequel we shortly review and discuss defuzzification methods used in designing neuro-fuzzy systems. By T and S we denote t-norm and t-conorm, respectively.

The basic method of defuzzification is the *center of gravity defuzzification* (COG) called also the *center of area defuzzification* (COA) defined by

$$\bar{y} = \frac{\int_{y \in \mathbf{Y}} y \cdot \mu_{B'}(y) dy}{\int_{y \in \mathbf{Y}} \mu_{B'}(y) dy}, \tag{12}$$

where B' , is an aggregated conclusion of reasoning for all rules and is calculated using equation (10). The discretization only in centers of consequents leads to the *center average defuzzification* (CA) called also the *height defuzzification* defined by

$$\bar{y} = \frac{\sum_{r=1}^N \bar{y}^r \cdot \mu_{B'^r}(\bar{y}^r)}{\sum_{r=1}^N \mu_{B'^r}(\bar{y}^r)}, \tag{13}$$

where

$$\mu_{B^r}(\bar{y}) = T(\mu_{A^r}(\bar{x}), \mu_{B^r}(\bar{y})). \tag{14}$$

However, this defuzzification in the case of logical approach yields $h(B_j^r) = 1$. This drawback can be removed if we apply the *discrete center of gravity defuzzification* (DCOG) [9], [12] defined by

$$\bar{y} = \frac{\sum_{r=1}^N \bar{y}^r \cdot \mu_{B^r}(\bar{y}^r)}{\sum_{r=1}^N \mu_{B^r}(\bar{y}^r)}, \tag{15}$$

where B^r is calculated as in the COG method using equation (10)

3 Singleton Interpretation of Fuzzification

As it has been mentioned in Section 2.1, the fuzzy systems are usually created using the singleton fuzzification. The realization of formula (11) is much simpler than realization of formula (9). In this section, it will be shown that we can use the same structure for modeling both singleton and non-singleton defuzzification in the case of DCOG defuzzification. The differences in the implementation are limited to the shape of antecedent fuzzy set membership functions. Theorems 1.2 determine membership functions of fuzzy sets A^r used in the neuro-fuzzy systems with the singleton fuzzification. These membership functions serve to model a fuzzy system with the non-singleton fuzzification by function \underline{A}^r , and with antecedent fuzzy sets \underline{A}^r .

The composition of a fuzzy premise with a fuzzy antecedent set in the case of Mamdani systems (reasoning with t-norms) was derived as the well known sup-T composition by Mouzouris and Mendel [6] (see also [11] and [13]). Figure 1 shows the example of such composition. Particularly, Figure 1b) presents Gaussian antecedent fuzzy sets of the non-singleton fuzzy system (dashed line), and antecedent fuzzy sets of singleton neuro-fuzzy system (solid line) when the fuzzification is proceeded using the fuzzy set shown in Figure 1a).

In this paper, we study fuzzification in logical-type fuzzy systems.

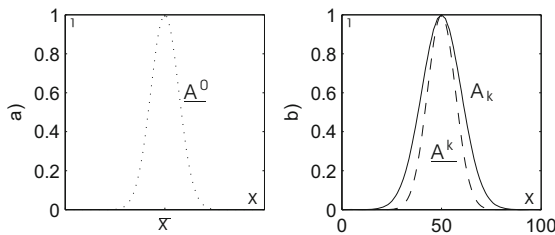


Fig. 1. Modification of antecedent fuzzy set in Mamdani-type fuzzy system

Theorem 1. For any neuro-fuzzy system based on *S*-implication and the DCOG (15) defuzzification method, the non-singleton fuzzification can be realized by using the antecedent fuzzy sets A^r , defined as follows

$$\mu_{A^r}(\bar{\mathbf{x}}) = N \left\{ \sup_{\mathbf{x} \in \mathbf{X}} T \left\{ \mu_{\underline{A}'}(\mathbf{x}), N \left\{ \mu_{\underline{A}'}(\mathbf{x}) \right\} \right\} \right\} \quad (16)$$

where r is a number of rules, \underline{A}^r is the antecedent fuzzy set from original fuzzy system, \underline{A}' is the result of input value fuzzification process, N stands for any involutive negation [2].

Proof. We can treat the two systems as equivalent in the following situation. If the input value $\bar{\mathbf{x}}$ given for both systems are the same then the output value of singleton \bar{y} and the output value of nonsingleton \bar{y} architecture are equal. Using CA (13) or DCOG (15) defuzzification and any type of aggregation (10), this condition will be fulfilled when

$$\mu_{B^r}(\bar{y}^r) = \mu_{\underline{B}^r}(\bar{y}^r) \quad (17)$$

or

$$\begin{cases} \mu_{B'^r}(\bar{y}^r) = \mu_{\underline{B}'^r}(\bar{y}^r) \\ \mu_{B'^j}(\bar{y}^r) = \mu_{\underline{B}'^j}(\bar{y}^r) \quad \forall j \neq r \end{cases} \quad (18)$$

where B'^r and B'^j are the results of the reasoning in the singleton neuro-fuzzy architecture and \underline{B}'^r and \underline{B}'^j are the results of the reasoning in the modeled nonsingleton neuro-fuzzy architecture. When we use formula (11) for the singleton architecture and formula (9) for the nonsingleton architecture we obtain the following

$$\begin{cases} I(\mu_{A^r}(\bar{\mathbf{x}}), \mu_{B^r}(\bar{y}^r)) = \sup_{\mathbf{x} \in \mathbf{X}} T \left\{ \mu_{\underline{A}'}(\mathbf{x}), I(\mu_{\underline{A}'}(\mathbf{x}), \mu_{B^r}(\bar{y}^r)) \right\} \\ I(\mu_{A^j}(\bar{\mathbf{x}}), \mu_{B^j}(\bar{y}^r)) = \sup_{\mathbf{x} \in \mathbf{X}} T \left\{ \mu_{\underline{A}'}(\mathbf{x}), I(\mu_{\underline{A}'}(\mathbf{x}), \mu_{B^j}(\bar{y}^r)) \right\} \end{cases} \quad (19)$$

or, using the condition about normal and nonoverlapping consequent fuzzy sets B^r , we obtain

$$\begin{cases} I(\mu_{A^r}(\bar{\mathbf{x}}), 1) = \sup_{\mathbf{x} \in \mathbf{X}} T \left\{ \mu_{\underline{A}'}(\mathbf{x}), I(\mu_{\underline{A}'}(\mathbf{x}), 1) \right\} \\ I(\mu_{A^j}(\bar{\mathbf{x}}), 0) = \sup_{\mathbf{x} \in \mathbf{X}} T \left\{ \mu_{\underline{A}'}(\mathbf{x}), I(\mu_{\underline{A}'}(\mathbf{x}), 0) \right\} \end{cases} \quad (20)$$

In the case of the logical approach, the first subequation of equations (20) are the identity if only set A' is normal, and the second subequation leads to equation (16) if only negation N is involutive.

Figure 2 shows the example for the application of Theorem 1. Particularly, Figure 2b) presents Gaussian antecedent fuzzy sets of the non-singleton fuzzy system (dashed line), and antecedent fuzzy sets of singleton neuro-fuzzy system (solid line) when the fuzzification is proceeded using the fuzzy set shown in Figure 2a).

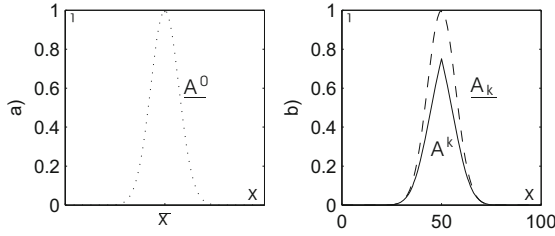


Fig. 2. Modification of antecedent fuzzy set in fuzzy system with S-implication

Theorem 2. For any neuro-fuzzy system based on Q-implication and the DCOG (15) defuzzification method, the non-singleton fuzzification can be realized by using the antecedent fuzzy sets A^r , defined as follows

$$\mu_{A^r}(\bar{\mathbf{x}}) = N \left\{ \sup_{\mathbf{x} \in \mathbf{X}} T \{ \mu_{\underline{A}^r}(\mathbf{x}), N \{ \mu_{\underline{A}^r}(\mathbf{x}) \} \} \right\} \quad (21)$$

where $r = 1, \dots, N$ is an index for rules and antecedent fuzzy sets, N is a total number of rules, \underline{A}^r is the antecedent fuzzy set from original fuzzy system, \underline{A}^r is the result of input value fuzzification process. T and S are cotinuous dual triangular norms with respect to operation N . Negation N is continuous and involutive.

Proof. The consideration leading in Proof 3 up to equations (20) are independent of a reasoning method. When we replace I in equations (20) by Q-implication, applying equation (7), the first subequation of equations (20) are the identity if T, S are dual with respect to N , all three functions are continuous and set A^r is normal. The second equation in (20) leads to equation (21) if only negation N is involutive.

4 Final Remarks

Not always the singleton fuzzification may be a reasonably good choice in the design of fuzzy systems. If the designer takes into consideration the imprecision of the measurements or the system operates on noisy input data then the non-singleton fuzzification comes as a much better choice. The non-singleton fuzzification leads to the hardly implemented formula (9) whereas the singleton fuzzification leads to the much simpler formula (11). In the paper, it has been shown and proven that the non-singleton fuzzification in the case of logical-type fuzzy systems (for S- and QL-implications) with DCGOG defuzzification can be realized in the same architecture as the singleton fuzzification. The only change required is the modification of the membership functions of antecedent fuzzy sets. The new membership functions are determined once before the fuzzy system performs.

References

1. Czogała, E., Łęski, J.: Fuzzy and Neuro-Fuzzy Intelligent Systems. Physica-Verlag, Springer, Heidelberg (2000)
2. Klement, E.P., Mesiar, R., Pap, E.: Triangular Norms. Kluwer Academic Publishers, Dordrecht (2000)
3. Lee, K.M., Kwang, D.H.: A fuzzy neural network model for fuzzy inference and rule tuning. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 2(3), 265–277 (1994)
4. Lin, C.T., Lee, G.C.S.: Neural-network-based fuzzy logic control and decision system. *IEEE Trans. Comput.* 40(12), 1320–1336 (1991)
5. Mas, M., Monserrat, M., Torrens, J., Trillas, E.: A survey on fuzzy implication functions. *IEEE Trans. Fuzzy Systems* 15(6), 1107–1121 (2007)
6. Mouzouris, G.C., Mendel, J.M.: Nonsingleton fuzzy logic systems: theory and application. *IEEE Trans. Fuzzy Systems* 5(1), 56–71 (1997)
7. Nauck, D., Klawonn, F., Kruse, R.: Foundations of Neuro-Fuzzy Systems. Wiley, Chichester (1997)
8. Nowicki, R.: Rough sets in the neuro-fuzzy architectures based on non-monotonic fuzzy implications. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) *ICAISC 2004. LNCS (LNAI)*, vol. 3070, pp. 518–525. Springer, Heidelberg (2004)
9. Rutkowska, D., Nowicki, R.: Implication - based neuro-fuzzy architectures. *International Journal of Applied Mathematics and Computer Science* 10(4), 675–701 (2000)
10. Rutkowska, D., Nowicki, R.: New neuro-fuzzy architectures. In: Proc. Int. Conf. on Artificial and Computational Intelligence for Decision, Control and Automation in Engineering and Industrial Applications, AcIDcA 2000, Monastir, Tunisia, March 2000, pp. 82–87 (2000)
11. Rutkowska, D., Nowicki, R., Rutkowski, L.: Singleton and non-singleton fuzzy systems with nonparametric defuzzification. In: *Computational Intelligence and Application*, pp. 292–301. Springer, Heidelberg (1999)
12. Rutkowska, D., Nowicki, R., Rutkowski, L.: Neuro-fuzzy architectures with various implication operators. In: *The State of the Art in Computational Intelligence - Proc. Intern. Symposium on Computational Intelligence (ISCI 2000)*, Kosice, pp. 214–219 (2000)
13. Rutkowska, D., Rutkowski, L., Nowicki, R.: On processing of noisy data by fuzzy inference neural networks. In: *Proceedings of the IASTED International Conference, Signal and Image Processing*, Nassau, Bahamas, October 1999, pp. 314–318 (1999)
14. Rutkowski, L., Cpalka, K.: Flexible neuro-fuzzy systems. *IEEE Trans. Neural Networks* 14(3), 554–574 (2003)
15. Rutkowski, L., Cpalka, K.: Designing and learning of adjustable quasi-triangular norms with applications to neuro-fuzzy systems. *IEEE Trans. Fuzzy Systems* 13(1), 140–151 (2005)
16. Yager, R.R., Filev, D.P.: *Essentials of Fuzzy Modeling and Control*. John Wiley and Sons, Chichester (1994)

Does an Optimal Form of an Expert Fuzzy Model Exist?

Andrzej Piegat and Marcin Olchowy

Department of Methods of Artificial Intelligence and Applied Mathematics
Faculty of Computer Science and Information Systems
Westpomeranian University of Technology
apiegat@wi.zut.edu.pl, molchowy@wi.zut.edu.pl

Abstract. In expert fuzzy models various membership functions, various operators for AND and OR operations, for implication in single rules and aggregation of activated conclusions of single rules in one resultant conclusion, and various defuzzification methods can be chosen by a modeler. Is there, in the universal case, possible to answer the question, which set of the above elements of the fuzzy model is optimal? The authors try to give the answer to this difficult, but very interesting question in this paper.

Keywords: fuzzy modeling, fuzzy set theory, fuzzy logic.

1 Introduction

The expert fuzzy models of the $y=f(x_1, \dots, x_n)$ dependences which exist in plants and systems are described in many excellent books and papers. A few of them, issued recently are: [1],[4],[5],[7]-[10]. These publications manifest a high development, acceptance and stabilization of fuzzy logic in the scientific community. The most often applied type of the fuzzy models seems to be the Mamdani - Assilan model, which is described in [6]. The characteristic feature of this type of model is the form of rules (1) used here in the version for the 3D-space,

$$\text{IF } (x_1 \text{ is } A_{1i}) \text{ AND } (x_2 \text{ is } B_{2j}) \text{ THEN } (y \text{ is } C_k) \quad (1)$$

where x_1, x_2 are the input variables, y is the output, $i=1, \dots, n$, $j=1, \dots, m$, $k=1, \dots, l$, A_{1i} , B_{2j} , C_k are the fuzzy sets, which represent the linguistic values such as: very small, small, medium, large, very large, close to 20, close to 50, close to 100, etc. An example of the rule, received from a car expert (for the fourth gear of a certain car), is shown in (2).

$$\begin{aligned} &\text{IF (the accelerator pedal is pressed to approximately 0.5 of its range of} \\ &\quad \text{movement)} \\ &\quad \text{AND (the way inclination is close to } 0^\circ) \\ &\quad \text{THEN (the car speed is close to 90 km/h)} \end{aligned} \quad (2)$$

In the expert rule there are present the fuzzy sets such as: “about 0.5”, “close to 0° ”, “close to 90km/h”. The definitions of these sets have to be extracted from the plant expert by the modeler. All the linguistic rules create the rule base, which is the basis

for the calculations of the output for given values of the plant inputs. However, to be able to use the rule base, the modeler has to make many measures quoted below [8].

1. Choose the form of the membership functions (triangle-, trapezoidal-, Gauss-, polynomial functions, etc)
2. Choose the operator type for the AND and OR operation for calculation of the premise truth in the rules (e.g. MIN, PRODUCT, Einstein PRODUCT, etc. in case of the AND-operation).
3. Choose the implication operator for calculation of the activated conclusions of the single rules (e.g. Mamdani-, Łukasiewicz-, Zadeh-implication, etc).
4. Choose the operator for aggregation of the activated conclusions of the single rules in one resultant conclusion (e.g. MAXIMUM, BOUNDED SUM, EINSTEIN SUM, etc).
5. Choose the method for defuzzification of the resultant conclusion for all the rules (e.g. the center - of - gravity method, the first - of - maxima method, the middle - of -maxima method, the height method, etc).

Many options can be chosen in each of 5 computational steps of the fuzzy model. Thus, there exists a very large number of combinations of the elements of the fuzzy model: of the membership functions, AND-, OR-, implication-, aggregation-operators and defuzzification methods, which can be chosen by the modeler. Each combination gives the different results of calculation. If the fuzzy model is constructed on the basis of the measurement samples of the real system, then the search method of the genetic algorithms, proposed e.g. in [7] can be applied to determine the optimal elements of the fuzzy model. However, in many cases no measurements are for disposal and this method cannot be used. Then, the model can be constructed only from the linguistic rules received from the system expert. The expert can also give us the definitions of the linguistic values which occur in the rules. However, he is not able to give us the elements (1-5) of the fuzzy model used by his mind. Therefore, the choice of these elements has to be made by the modeler himself. But which set of the fuzzy model elements is optimal? And, if it is optimal then in which meaning? Which criterion of the optimum of the solution should be used for the evaluation of the model if there are no measurements for evaluation of the quality of the model? According to the authors' knowledge, the above problems have not been solved yet. But the expert fuzzy models and controllers are being constructed and applied. Which way, currently, the modelers solve the task of choice of the model elements in practice? They make it in two ways given below.

1. The partly arbitral choice. There are chosen such sets of the fuzzy model elements, which are used by others scientists (the imitation method).
2. The fully arbitral choice of the fuzzy model elements which depends on the modeler only.

The arbitrariness in the construction of the fuzzy models is the important reason of attacks on fuzzy logic and disrespect of this theory as the "unripe" theory by a number of scientists. Example of such an opinion is the Hand's opinion expressed in [3]: "fuzzy logic ... together with possibility theory and rough set theory remains rather a controversial theory: it does not have the solid bases, a widespread application and acceptance of probability". The Hand's opinion is also very controversial. There exist

many publications, which show a direct connection of fuzzy logic and its results with the strongly based classical mathematics. The example of one of the last of such books is [4]. One of the possibilities to refute the Hand's objections is to show that the arbitrariness in the choice of the elements of the fuzzy models can be eliminated, which is the subject of this paper. However, because of the volume limitation of the paper to 8 pages, an exhaustive explanation of this problem is not possible. The expanded explanation will be published in the Marcin Olchowy's doctoral thesis, which is currently prepared.

2 Expert Rule as an Approximate Point of Knowledge (APK) in the 2D-Space

The input-output dependences in the 2D-space are of the $y=f(x)$ form. Let us assume that there is the expert knowledge in the form of 2 rules (3) for disposal.

$$\begin{aligned} \text{R1. IF } (x \text{ is close to } x_a) \text{ THEN } (y \text{ is close to } y_a) \\ \text{R2. IF } (x \text{ is close to } x_b) \text{ THEN } (y \text{ is close to } y_b) \end{aligned} \quad (3)$$

The example of such an expert knowledge can be the knowledge of the dependence between the pressing degree x of the car acceleration pedal and the car speed y (the gear III, the horizontal way).

$$\begin{aligned} \text{R1. IF } (x \text{ is close to } 0) \text{ THEN } (y \text{ is close to } 25 \text{ km/h}) \\ \text{R2. IF } (x \text{ is close to } 1) \text{ THEN } (y \text{ is close to } 90 \text{ km/h}) \end{aligned} \quad (4)$$

The expert knowledge is most often referred to the border points of the dependence $y=f(x)$, of its maxima, minima and other special points of the dependence. The expert rules (3) are shown in a graphical way on Fig.1.

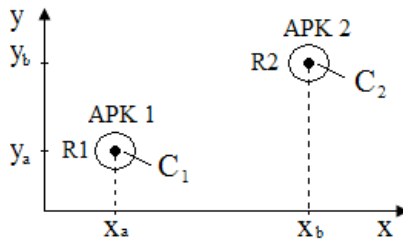


Fig. 1. The fuzzy rules R1 and R2 (3) as the approximate points of the expert knowledge (APK). The C1, C2 – the centers of the points.

It should be noticed (very important!) that the supports of the fuzzy sets “close to x_a ”, “close to x_b ” are usually referred not to the entire interval $[x_a, x_b]$, but only to the near neighborhood of the C1 and C2 centers of the APKs, Fig.2.

The definitions of the expert membership functions usually do not have a large range (the support). E.g., no car driver understands the speed of 25 km/h as a speed “close to 90 km/h”. Usually, just the speed of 90 ± 10 km/h is understood as such a speed. Also the shape and parameters of the expert membership functions cannot be

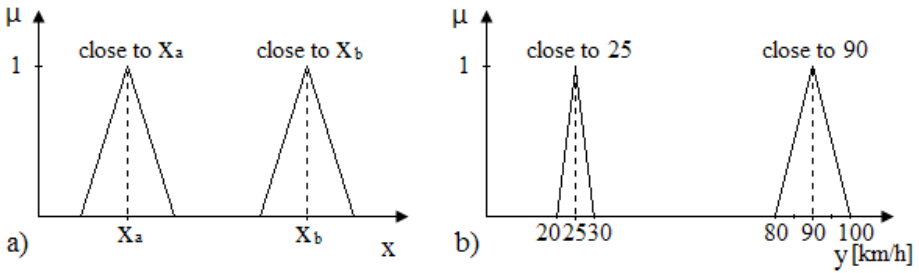


Fig. 2. The illustration of the limited range of expert linguistic values, a) the general case, b) the case of the car speed

extracted accurately by the modeler from the expert. If the expert possesses only the knowledge concerning a small vicinity of the APK-centers then two important questions arise.

1. In which way does the expert make inferences about the function $y=f(x)$ -values between the knowledge points?
2. In which way should the fuzzy model calculate the $y=f(x)$ -values between the two knowledge points?
 (How should the fuzzy model execute the interpolation between the knowledge points)?

Let us notice that if the expert knowledge is referred to only a small vicinity of two APKs then the interval between the points causes the “information gap” [12]. The expert doesn’t know which are the values of the $y=f(x)$ function between APK1 and APK2. In the interval of the information gap no statements can be formulated, because there are no measurements (no knowledge) in the gap. Therefore, also no statements can be proved in the experimental way. However, the information gaps can be filled up with the credible hypotheses, as the theory of information gaps suggests [12]. The human intelligence universally uses the method of credible hypotheses to solve many everyday problems, which result from the information gaps, especially the problems referred to the future events. Thus, to execute the interpolation between the two knowledge points, possibly good interpolation hypotheses should be assumed.

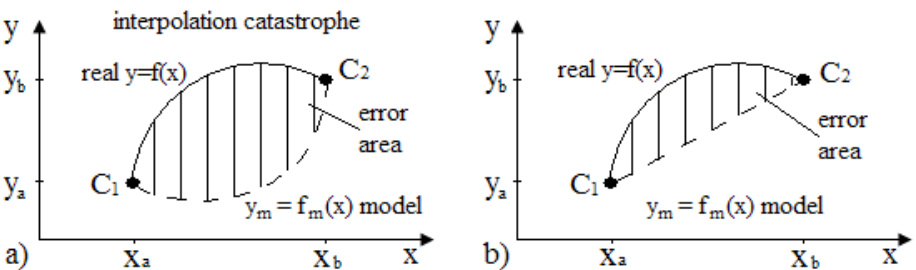


Fig. 3. The illustration of the possibility of the interpolation catastrophe by the inappropriate fuzzy model $y_m=f_m(x)$, (Fig.3a) and the possibility to decrease the maximal possible error by the type of the fuzzy model (Fig.3b) chosen in the appropriate way

However, what does the “possibly good hypothesis” definition mean? Which criterion of evaluation of the hypothesis should be used in this case? Because in the information gap no measurements are for disposal, therefore the criterion of absolute or square model error cannot be used. The only rational criterion seems: the criterion of minimization of the possible maximal interpolation error, which prevents against so called “interpolation catastrophes”. Let us consider the problem presented in Fig.3.

Choice of the fuzzy model elements determines the type of the interpolation between the APKs. If the model elements were chosen in an inappropriate way, then the model can execute a convex interpolation, when the real characteristic of the $y=f(x)$ plant is of a concave type (Fig.3a). It causes the interpolation catastrophe because of the model error, e.g. the absolute model error (5) will be very large.

$$\text{area of the absolute error} = \int_{x_a}^{x_b} |y - y_m| dx \tag{5}$$

The most safe interpolation between the two knowledge points seems to be the linear interpolation, Fig.3b., which prevents the very large interpolation errors. The following conclusion can be drawn from the above consideration: the elements of the $y_m=f_m(x)$ fuzzy models in the 2D-space should be selected so that the model will execute the $y_m=a_0+a_1x$ linear interpolation between the expert knowledge points, as it is shown in Fig.4.

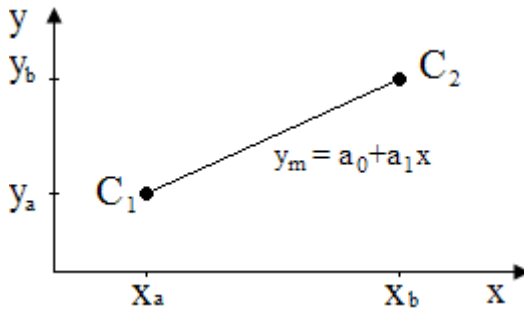


Fig. 4. The optimal, safe interpolation between the two points of the expert knowledge (the linear interpolation) in the 2D-space

Let us notice that the linear interpolation shown in Fig.4 possesses not only the feature of linearity. It is also the shortest and the simplest interpolation between the two points and it satisfies the Occam’s razor postulate of simplicity of the model [11]. To achieve the effect of the linear interpolation between the two APKs, the following, given below, elements of the fuzzy model should be chosen.

1. The triangle membership functions of the fuzzy sets of the x input that satisfy the condition of the unity partition [8] with the supports, which cover the entire $[x_a, x_b]$ interval. These supports usually differ from the small supports of the fuzzy sets given by the experts. The singleton fuzzy sets should be chosen for the y output.
2. The PRODUCT or MIN operators should be chosen for the implication in single rules.

$$\mu_{A \rightarrow B}(x, y) = \mu_A(x)\mu_B(y)$$

$$\mu_{A \rightarrow B}(x, y) = \text{MIN}[\mu_A(x)\mu_B(y)]$$

3. The SUM operator should be chosen for aggregation of the activated singleton-conclusions of all the rules.

$$\mu_{\text{Res } B_1 \cup B_2}(y) = \mu_{B_1}(y) + \mu_{B_2}(y)$$

4. The defuzzification should be executed with the method of the center of gravity of the singletons.

3 Expert Rules and the Safe, Fuzzy Interpolation in the 3D-Space

Let us assume that 4 expert knowledge points (6) are for disposal in the 3D-space.

- R1. IF (x_1 is close to x_{1a}) AND (x_2 is close to x_{2a}) THEN (y is close to y_{aa})
 - R2. IF (x_1 is close to x_{1b}) AND (x_2 is close to x_{2a}) THEN (y is close to y_{ba})
 - R3. IF (x_1 is close to x_{1a}) AND (x_2 is close to x_{2b}) THEN (y is close to y_{ab})
 - R4. IF (x_1 is close to x_{1b}) AND (x_2 is close to x_{2b}) THEN (y is close to y_{bb})
- (6)

The expert knowledge is usually referred to only a small vicinity of the centers C_i of APKs and it does not cover the full universe of discourse of the problem. Therefore, the space between the knowledge points makes the information gap, Fig.5.

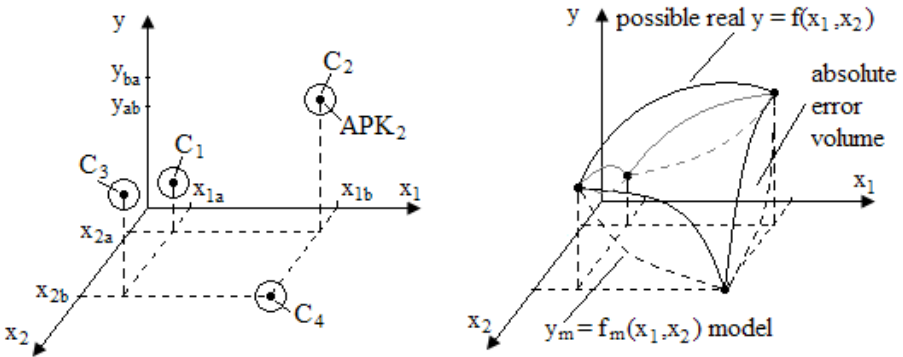


Fig. 5. Example of 4 expert knowledge points with the C_i centers and with the inner space which makes the information gap (Fig.3a), the surface of the real, but unknown $y=f(x_1, x_2)$ dependence and the surface of an exemplary model $y_m=f_m(x_1, x_2)$, which results from the inappropriate choice of the model elements

The real dependence $y=f(x_1, x_2)$, which exists in the modeled system is, apart from the APKs, not known to the expert. The inappropriate choice of the fuzzy model elements results in the $y_m=f_m(x_1, x_2)$ model surface, which differs dramatically from the real model surface and can be a reason of the very large (absolute) model errors (of the interpolation catastrophe) given by (7).

$$\text{volume of the absolute error} = \int_{x_{1a}}^{x_{1b}} \int_{x_{2a}}^{x_{2b}} |y - y_m| dx_1 dx_2 \tag{7}$$

To prevent the very large errors, there should be selected the appropriate type of interpolation. Unfortunately, in case of the 3D-space, the $y_m=a_0+a_1x_1+a_2x_2$ linear interpolation cannot be applied as in the 2D-space because of the 4 knowledge points. In the universal case, no flat surface exists (the linear function), which goes through 4 points. The analyses of the question suggests using such an interpolation model, which has the minimal area of the model surface and creates the linear interpolation between all pairs of border knowledge points. The model- surface area S can be calculated with the formula (8) taken from [2].

$$S = \int_{x_{1a}}^{x_{1b}} \int_{x_{2a}}^{x_{2b}} \sqrt{1 + \left(\frac{\partial f_m}{\partial x_1}\right)^2 + \left(\frac{\partial f_m}{\partial x_2}\right)^2} dx_1 dx_2 \tag{8}$$

Because there exists an infinite number of the $y_m=f_m(x_1,x_2)$ different models which can interpolate the information gap between 4 points of the expert knowledge, thus it is probably impossible to determine in an analytical way which of them is optimal and minimizes the S area of the interpolation surface given by the formula (8). One of the authors, Andrzej Piegat, has formulated a hypotheses that the optimal interpolation is the interpolation of the polynomial type, given by (9) with the number of degrees of freedom equal to the number of the knowledge points, i.e. 4 in the 3D-space.

$$y_m = a_0 + a_1x_1 + a_2x_2 + a_3x_1x_2 \tag{9}$$

Remark: each cut of this interpolation surface in 3D for $x_1=\text{const}=e$ or for $x_2=\text{const}=f$ gives the linear interpolation (9) in the 2D-space, which can be observed in Fig.5 (see the border lines of the model).

$$y_m=(a_0+a_1e)+(a_2+a_3e)x_2 \quad \text{or} \quad y_m=(a_0+a_2f)+(a_1+a_3f)x_1 \tag{10}$$

Correspondingly, in the 4D-space the optimal interpolation is given by (11): 8 degrees of a_i freedom correspond in this polynomial to 8 knowledge points APKs in the 4D-space.

$$y_m=a_0+a_1x_1+a_2x_2+a_3x_3+a_4x_1x_2+a_5x_1x_3+a_6x_2x_3+a_7x_1x_2x_3 \tag{11}$$

Fig.6 shows in which way the nonlinear interpolation (9) prevents the interpolation catastrophes and minimizes the maximal, possible model error.

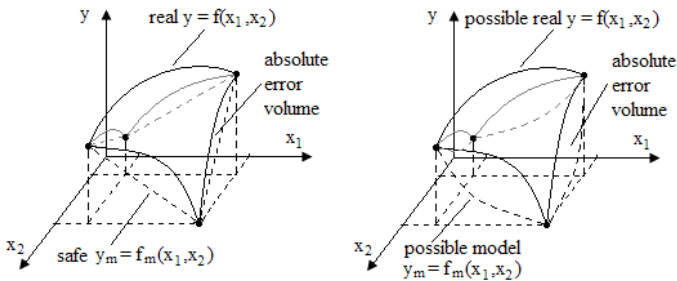


Fig. 6. Illustration of the minimization effect of the maximal, possible model error by the polynomial interpolation (9), Fig.5a, and illustration of the possible interpolation catastrophe in case, when a different, arbitrarily chosen interpolation type will be used, Fig.5b

If the fuzzy model $y_m=f_m(x_1, x_2)$, is to be able to create an optimal and safe interpolation surface (8) in the meaning of interpolation catastrophe, the following given below elements of the model should be assumed.

1. The triangle membership functions for the fuzzy sets of the inputs x_1, x_2 (“close to x_{1a} ,” “close to x_{1b} ,” “close to x_{2a} ,” “close to x_{2b} ”) with the supports which cover the entire $[x_{1a}, x_{1b}]$ and $[x_{2a}, x_{2b}]$ intervals. These supports are usually different from the supports given by the experts which typically refer only to a small vicinity of the $x_{1a}, x_{1b}, x_{2a}, x_{2b}$ points.
2. The singleton membership functions for the y output, placed in the C_i centers of the expert knowledge points.
3. The PRODUCT-operators for the AND operation in the rules premises (if the rule premise contains the OR operation, then it can be easily transformed in the corresponding number of rules with the premises which contain only the AND-operation).
4. The PRODUCT- or MIN-operator for the $A \rightarrow B$ implication in the single rules.

$$\mu_{A \rightarrow B}(x_1, x_2, y) = \mu_A(x_1, x_2)\mu_B(y)$$

$$\mu_{A \rightarrow B}(x_1, x_2, y) = \text{MIN}[\mu_A(x_1, x_2), \mu_B(y)]$$

5. The SUM-operator for aggregation of the activated conclusions of all rules in one resultant conclusion.

$$\mu_{\text{Res } B_1 \cup B_2}(y) = \mu_{B_1}(y) + \mu_{B_2}(y)$$

6. The center-of-gravity method for defuzzification of the resultant singleton-conclusion of the rule base.

As an analytic proving the hypothesis that the elements of the fuzzy model shall be chosen, so that the fuzzy model $y_m = f_m(x_1, x_2)$ could execute the polynomial, nonlinear interpolation (9) with the minimal surface area and the linear borders, is probably not possible the authors executed quite a lot of the computer experiments with the different fuzzy models, which manifested that the polynomial interpolation (9) really gives the required, minimal surface area. However, because of limitation of the paper volume, results of only two experiments of the fuzzy modeling in the space 3D will be presented. The task of the fuzzy model was to generate the interpolation surface between centers of 4 expert knowledge points $\{x_1, x_2, y\}$: $\{0,0,0\}$, $\{100,0,500\}$, $\{0,100,500\}$, $\{100,100,0\}$. Fig.7 shows the interpolation surfaces generated by 2 different models and gives the numerical areas of their surfaces.

Fig.7a. shows a nonlinear surface which consists of straight lines (each line is a cut of the surface according to formula (10)).

The authors executed many computer experiments, which confirmed that the interpolative fuzzy model possessing elements consistent with the recommendations 1-6 has the minimal surface area and the linear borders and that the other fuzzy models do not have these properties. It means that it is the optimal one from the point of view of the interpolation catastrophe. Therefore, this model can be referred to as the safe model.

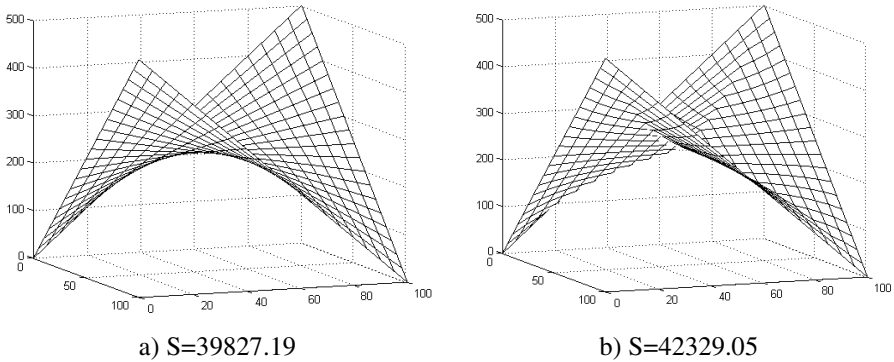


Fig. 7. The interpolation surface of the fuzzy model (area $S=39827.19$), which possesses the elements recommended by the points 1-6, (Fig7a) and the interpolation surface of the fuzzy model ($S=42329.05$) with the elements inconsistent with the recommendations 1-6, where the operator PRODUCT for AND-operation in premises was replaced by the MIN-operator (Fig.7b)

4 Conclusions

The investigations executed by the authors with reference to the question, which set of the fuzzy model elements is the optimal have shown that they determined this set. This is the set of such elements, which makes the fuzzy model to generate the interpolation surface between the expert knowledge points which have the minimal area and the linear borders. The optimal interpolation surface is given by the polynomial with the number of degrees of freedom equal to the number of the expert knowledge points. In case of the 3D-space the optimal polynomial is given by the formula (9), for the 4D-space by the formula (11). These polynomials determine the elements of the optimal fuzzy model, which was given by the recommendations 1-6 in Chapter 4. The optimal fuzzy model thanks to its minimal interpolation surface can be referred to as the safe model, because it prevents from possible interpolation catastrophes in the meaning of the large model errors. It satisfies the postulate of simplicity recommended by the Occam's razor-concept. Because of the required volume limitation of the paper the problem was presented only in a very limited way. The author of the theory and concepts in this paper is Andrzej Piegat, whilst the computer experiments were executed by Marcin Olchowy.

References

1. Babuska, R.: Fuzzy modeling for control. Kluwer Academic Publishers, Dordrecht (1998)
2. Bronsztejn, I.N., et al.: Compendium of mathematics. Scientific Publishers PWN, Warsaw (2004) (in Polish)
3. Hand, D., et al.: Principles of data mining. Massachusetts Institute of Technology (2001)
4. Kluska, J.: Analytical methods in fuzzy modeling and control. Springer, Heidelberg (2009)
5. Łęski, J.: Neuro-fuzzy systems. Scientific-Technical Publishers WNT, Warsaw (2008) (in Polish)

6. Mamdani, E.H., Assilan, S.: An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies* 20(2), 1–13 (1975)
7. Pedrycz, W., Gomide, F.: *Fuzzy systems engineering*. John Wiley & Sons, Inc., Hoboken (2007)
8. Piegat, A.: *Fuzzy modeling and control*. Springer, Heidelberg (2001)
9. Ross, T.J.: *Fuzzy logic with engineering applications*. Willey, New York (2004)
10. Rutkowski, L.: *Computational intelligence: methods and techniques*. Springer, Heidelberg (2008)
11. Wikipedia (2009), <http://en.wikipedia.org/wiki/Occam's-razor>
12. Yakov, B.H.: *Info-gap decision theory*, 2nd edn. Elsevier, Amsterdam (2006)

Fuzzy Logic in the Navigational Decision Support Process Onboard a Sea-Going Vessel

Zbigniew Pietrzykowski, Janusz Magaj, Piotr Wolejsza, and Jarosław Chomski

The Institute of Marine Navigation,
Maritime University of Szczecin, Szczecin, Poland
z.pietrzykowski@am.szczecin.pl, j.magaj@am.szczecin.pl,
p.wolejsza@am.szczecin.pl, j.chomski@am.szczecin.pl

Abstract. Ship collisions make up one of the major hazards in marine navigations. The problem of collision prevention in ship encounter situations are characterized. Described with fuzzy logic tools, criteria for the choice of track are discussed. The presented algorithm of ship track optimization in collision situations is based on the method of multi-stage control in fuzzy environment. Finally, the algorithm integrated into the navigational decision support system is presented.

1 Introduction

To select the right track in a collision situation, the navigator has to analyze and evaluate the situation and to determine a new ship movement trajectory, such that other vessels will pass safely. In a ship encounter situation, when a risk of collision exists, the decision to be made comes down to the determination of collision prevention manoeuvre: change (if any) of own ship's course and/or speed, or the trajectory and/or speed, assuring safe passing of the encountered vessel.

The selected solutions should comply with the regulations in force, assure safe performance of the manoeuvre and be rational. This means that criteria of the choice of track used by navigators have to be taken into account. These criteria are often inaccurate, or uncertain. The human being uses linguistic terms, such as 'safe distance' or 'little deviation from course'. If we consider the problem of the choice of track as a multi-stage control problem, and accounting for inaccurate formulation of the relevant criteria, this problem can be formulated as one of the multi-stage control in a fuzzy environment.

Navigational decision support systems are the latest step in the development of shipboard information systems. Apart from information functions, their task is to generate solutions - safe trajectories of ship movement determined in the process of collision prevention.

2 Navigational Decision Support on a Sea-Going Vessel

2.1 Criteria of Navigational Situation Analysis and Assessment

An analysis and assessment of navigator's own ship situation make up one stage in the decision process. These authors have developed inference algorithms for

the interpretation of Collision Regulations (COLREGs). They have also considered principles, implied by the said regulations, which should be used by ships that see each other or in restricted visibility. The process of algorithmization takes account of ships encounter phases and ship's stand-on or give-way status.

The assumed basic criteria for navigational situation assessment are the CPA (Closest Point of Approach) and the time to CPA (TCPA). These two parameters are commonly used in automatic radar plotting aids (ARPA). It is assumed that the navigator will define minimum values of: (safe) closest point of approach (CPA_{limit}) and the time to the closest point of approach ($TCPA_{\text{limit}}$). If the CPA value is lower than the defined CPA_{limit} and the TCPA is shorter than the assumed $TCPA_L$, then a collision preventing manoeuvre has to be performed immediately.

The criterion of ship domain has been additionally introduced for situation assessment. The ship domain is an area around the ship that the navigator wants to keep clear of other vessels and objects [6].

Uncertainties (inaccuracies) in safety assessment can be coped with by using fuzzy logic, which allows to describe the level of safety with linguistic values used by the human: 'safe', 'barely safe', 'dangerous', etc. In this step crisp values, e.g. the measured distance x , are attributed degrees of membership $\mu(x) \in \langle 0, 1 \rangle$. This means that, apart from the membership (1) or lack of membership (0), there may be partial membership. This applies to the fuzzy closest point of approach and the ship fuzzy domain.

The former fuzzy criterion of safety assessment means that the existence of tolerance interval $\langle CPA_{L_{\min}}, CPA_{L_{\max}} \rangle$ ($CPA_{L_{\min}} \leq CPA_L \leq CPA_{L_{\max}}$) is assumed, and any CPA value is attributed a degree of membership to the fuzzy set of CPA_{LF} . This degree is described by the function of membership to this set (μ_{CPALF}).

The ship fuzzy domain is similarly interpreted. The criterion for safe passing of other ships will then be the set 'ship fuzzy domain' D_{SF} , described by the function of membership to this set (μ_{DSF}). Both the fuzzy CPA and the ship fuzzy domain are criteria that take into consideration imprecision or uncertainties characteristic of the human being. A detailed analysis of these criteria is given in [6].

2.2 Criteria of the Choice of Track

The navigator should define a trajectory that, to be followed, requires certain manoeuvres. These have to be effective, complying with the regulations, performed in due time and noticeable to other navigators in the vicinity. These manoeuvres should be feasible and economical in terms of track covered, time and fuel consumption.

An effective manoeuvre is one that results in safe passing of obstructions and navigational dangers. The criteria of safety are given earlier herein: fuzzy closest point of approach (CPA_{LF}) and ship fuzzy domain (D_{SF}).

The manoeuvre done early enough and noticeably is aimed at such ship conduct that the other traffic participants will be able to observe and assess the

effectiveness of manoeuvres and will not force other navigators to take extra action.

The criterion of a noticeable manoeuvre can be formulated as follows: when the course is altered, it is recommended that this alteration is noticed by others. This means that such alteration should be possibly close to that recommended by experienced navigators and in line with directions given in ship handling handbooks. The above criterion can be presented using the fuzzy sets theory as a fuzzy set of noticeable course alterations C_{CWF} , described by the function of membership to this set μ_{CWF} .

In an optimization problem of the choice of track, factors such as lengthened track, time, fuel consumption etc. make up elements of the objective function (quality indicator) and are minimized. The lengthened track can be expressed by the value of deviation from the original trajectory. If we assume that the minimum and maximum acceptable deviations from the original trajectory are known, the lengthened track can be presented in the form of a set of fuzzy deviations from the original trajectory C_{LF} , described by the function of membership to this set (μ_{CLF}). Detailed analysis of the mentioned criteria can be found in [5].

The discussed criteria can be used in problems of ship movement trajectory optimization, in single- or multi-decision processes (single- and multi-stage control).

2.3 Problem of Safe Ship Movement Trajectory Determination

The problem of determining a safe trajectory of a proceeding ship can be considered as a single-stage or multi-stage control [4], [5].

For a given finite space of states $\mathbf{X} = \{x_1, \dots, x_n\}$ and a finite space of control settings $\mathbf{U} = \{u_1, \dots, u_m\}$ the transitions of states in subsequent k control stages are defined by this function:

$$f : \mathbf{X} \times \mathbf{U} \rightarrow \mathbf{X} \tag{1}$$

such that:

$$x_{t_{i+1}} = f(x_{t_i}, u_{t_i}), i = 0, 1, 2, \dots, k - 1 \tag{2}$$

This problem may consist in the determination of optimal trajectory by specifying the waypoints and headings on the sections defined by these points, or by rudder settings or/and speed at specific instants. In both cases the formulation of criteria for the choice of the track is essential. If we bear in mind that the goals and constraints of ship trajectory optimization problem are inaccurately defined in real conditions, the multi-stage decision-making (control) in a fuzzy environment can be an alternative to the classic approach of dynamic optimization [3]. The concept of fuzzy environment [1] is understood as the ordered four $\langle \mathbf{G}, \mathbf{C}, \mathbf{D}, \mathbf{U} \rangle$ (\mathbf{G} – fuzzy goal, \mathbf{C} – fuzzy constraints, \mathbf{D} – fuzzy decision, \mathbf{U} – set of decisions). The fuzzy goal is defined as the fuzzy set $\mathbf{G} \subseteq \mathbf{U}$ described by the function of membership μ_G :

$$\mu_G : \mathbf{X} \times \mathbf{U} \rightarrow [0, 1] \in R \tag{3}$$

and the fuzzy constraint as the fuzzy set $\mathbf{C} \subseteq \mathbf{U}$ with the membership function μ_C :

$$\mu_C : \mathbf{X} \times \mathbf{U} \rightarrow [0, 1] \in R \tag{4}$$

For a given finite space of states $\mathbf{X} = \{x_1, \dots, x_n\}$ and finite control space $\mathbf{U} = \{u_1, \dots, u_m\}$ the transitions of states in subsequent k control stages are defined by the function (2).

When a decision is made in a fuzzy environment, i.e. with a constraint \mathbf{C} and goal \mathbf{G} , described, respectively, by membership functions $\mu_C(x)$ and $\mu_G(x)$ the fuzzy decision \mathbf{D} is determined from this relation:

$$\mu_D(x) = \min_{x \in X} (\mu_G(x), \mu_C(x)) \tag{5}$$

It is assumed that an optimal decision is the one that maximizes the degree of membership to the set of fuzzy decision \mathbf{D} :

$$\mu_D(x^*) = \max_{x \in X} (\mu_D(x)) \tag{6}$$

This also refers to a situation when many constraints and goals exist. Then the fuzzy decision is defined as:

$$\mu_D(x) = \mu_{G_1}(x) * \mu_{G_2}(x) * \dots * \mu_{G_s}(x) * \mu_{C_1}(x) * \mu_{C_2}(x) * \dots * \mu_{C_p}(x) \tag{7}$$

where: p – number of goals, s - number of constraints.

The control process for the space of states \mathbf{X} and space of controls \mathbf{U} consists in selecting controls u with imposed constraints $\mu_C(x)$, while at each subsequent stage the goals $\mu_G(x)$ are imposed on the states x . In the multi-stage process of decision making (control) the process quality indicator for k stages is assumed to be the fuzzy decision:

$$\mathbf{D}(x_{t_0}) = \mathbf{C}^0 * \mathbf{G}^1 * \mathbf{C}^1 * \mathbf{G}^2 * \mathbf{C}^{k-1} * \mathbf{G}^k \tag{8}$$

Described by these membership functions:

$$\mu_D(u_{t_0}, \dots, u_{t_{k-1}} | x_{t_0}) = \mu_{C_0}(u_{t_0}) * \mu_{G_1}(x_{t_1}) * \dots * \mu_{C_{k-1}}(u_{t_{k-1}}) * \mu_{G_k}(x_{t_k}) \tag{9}$$

The states $x_{t_1}, x_{t_2}, \dots, x_{t_k}$ reached are defined by subsequent application of the equation (2). The problem of multi-stage control in a fuzzy environment is formulated as follows:

$$\mu_D(u_{t_0}^*, \dots, u_{t_{k-1}}^* | x_{t_0}) = \max (\mu_D(u_{t_0}, \dots, u_{t_{k-1}} | x_{t_0})) \tag{10}$$

Then the optimal strategy consists of a series of controls u^* :

$$u^* = (u_{t_0}^*, u_{t_1}^*, \dots, u_{t_{k-1}}^*) \tag{11}$$

The problem above can be solved by dynamic programming method, branch and bound method, or using the graph theory.

In the problem of defining a safe ship movement trajectory, the goal is to safely pass other vessels or objects. This goal is described by functions of membership to the fuzzy closest point of approach, or alternatively, ship fuzzy domain. There are two constraints in this optimization problem: a visible course alteration in the form of fuzzy set C_{CWF} , described by the function of membership to this set μ_{CWF} and lengthened tracks in the form of fuzzy set of deviations from the original trajectory C_{LF} described by the function of membership to this set (μ_{CLF}).

3 Navigational Decision Support System on a Sea-Going Ship

3.1 Characteristics of the System

Basic functions of the navigational decision support system include automatic acquisition and distribution of navigational information, analysis of a navigational situation, generating solutions to collision situations and interaction with the navigator.

A prototype of such system has been developed at the Institute of Marine Navigation, Maritime University of Szczecin [7]. The system utilizes knowledge of experienced navigators using artificial intelligence methods and tools, including fuzzy logic. One essential feature of the system is that generated conclusions can be explained. The prototype was tested in laboratory and real conditions, onboard ship.

The system has the following functions:

- acquisition, fusion and integration of navigational data available on a ship,
- visualization of a navigational situation,
- analysis of a navigational situation making use of navigators' criteria,
- signaling of dangerous situations and the present level of navigational safety,
- determination of a manoeuvre and movement trajectory in collision situations,
- visualization of the proposed a manoeuvre / manoeuvres,
- explanation (justification) of the generated solution.

3.2 Decision Support Process

The process of supporting navigator's decisions in the discussed system is shown using a record of subsequent encounter situations. The ECDIS simulator operated at the Maritime University of Szczecin was used for the purpose.

Figure 1 illustrated an initial phase of the navigator's (own) ship encounter with three other ships. The situation and proposed solutions are shown on the screen by means of the graphic user interface (GUI) and a standard electronic navigational chart (ENC).

According to COLREGs (good visibility conditions), the situation involving the indicated (yellow) vessel was classified as ships on crossing courses. The navigator's ship is obliged to give way to the target vessel. This information is

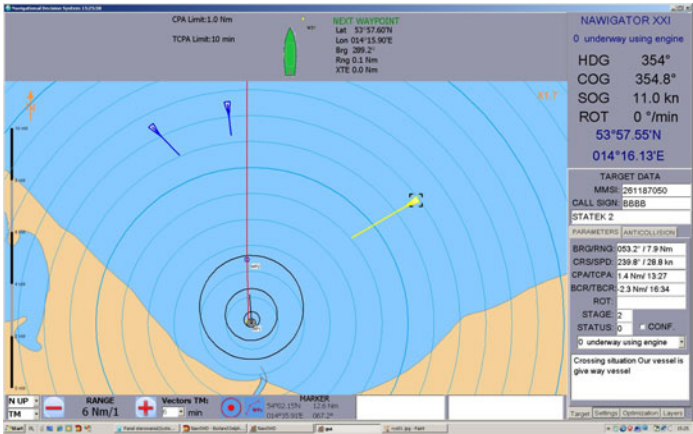


Fig. 1. Navigational decision support system on a sea-going vessel - qualification of a situation

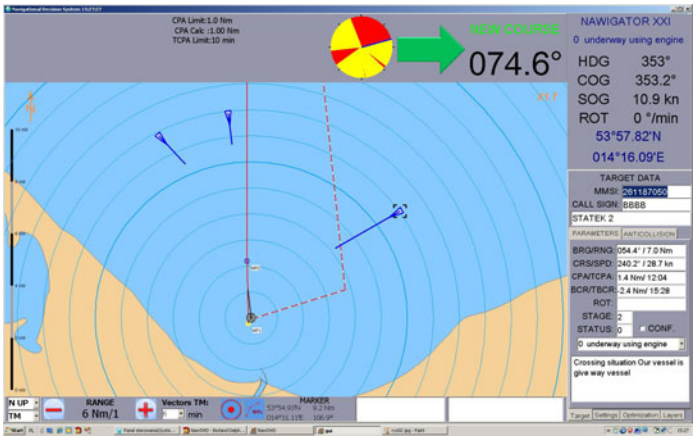


Fig. 2. Recommended course and trajectory of own ship in a collision situation

displayed on the screen in the bottom right-hand screen corner. As the ranges between vessels are still long, the system does not generate any solutions. This is compatible with the regulations as well as principles of good sea practice. In this situation the system is capable of defining an optimal safe trajectory on operator's request (multi-stage control, multi-stage control in the fuzzy environment).

When the navigator chooses the ship domain or ship fuzzy domain as the criterion of safety assessment, the domain is displayed on the GUI screen. The fuzzy domain is defined by the boundaries for three safety levels (degree of membership to the fuzzy set 'dangerous situation') 0.1, 0.5 and 0.9.

Figure 2 shows the same encounter situation as Figure 1, the difference being that the ranges are shorter now. The navigator's ship, with no right of way, is

obliged to take action in order to avoid a dangerous situation - it has to make a collision preventing manoeuvre.

The basic solution to the collision situation generated by the system and displayed to the navigator is course alteration. The scopes of allowable safe course values of own ship are presented in the upper part of the screen (circle, yellow sections). The proposed value of own ship course is also graphically depicted (circle - blue line) and alphanumeric form (new course). The navigator may additionally request that the solution is supplemented with a graphic display of the proposed safe new trajectory (red dashed line).

The trajectory presented to the navigator complies with the binding international collision regulations. Other safe trajectories specified by the system can be displayed on navigator's request.

3.3 Integration of the Multi-stage Control Algorithm in a Fuzzy Environment

The algorithm introduced in section 2.3 for the determination of safe ship movement trajectory by the method of multi-stage control in a fuzzy environment was integrated into the navigational decision support system. The graph method was used to solve the optimization problem [2].

Figure 3 shows the previously discussed situation. Now the optimization results - proposed safe trajectories - are depicted graphically. In addition, parameters of the recommended manoeuvres are displayed in the alphanumeric form in the bottom right-hand corner of the screen.

The safe trajectories were determined by the multi-stage control in a fuzzy environment method for the criterion of fuzzy closest point of approach. As a standard, according to the COLREGs, one trajectory is displayed for a course alteration to starboard. If there is no solution or on operator's request a safe trajectory running to port is shown. The trajectory is described by subsequent waypoints with their positions, times to reach these points and courses enabling movement along the specified trajectory. It practically means that to pass other vessels safely the vessel in question (own vessel) has to perform a sequence of course alterations. The system herein presented has an option of automatic performance of the manoeuvres using the ship's autopilot.

The defined trajectory is supplemented with the times to start a manoeuvre, course alterations (satisfying the requirement of noticeable manoeuvre), CPAs to the other vessels (safety criterion) and deviations from the original trajectory (criterion of minimized track lengthening, or deviation).

For comparison, the additionally determined movement trajectory is marked as well, showing the recommended manoeuvre of course alteration (dashed red line). Optimal trajectories determined by the method of multi-stage control in a fuzzy environment for the criterion of fuzzy closest point of approach are characterized by minimized lengthening of track and associated time increase. Observed by the navigator, the trajectory optimal for the course altered to starboard is in compliance with the COLREGs. The other marked change - course alteration to port - is not recommended, although in a specific situation, particularly to avoid a collision, that trajectory is acceptable.

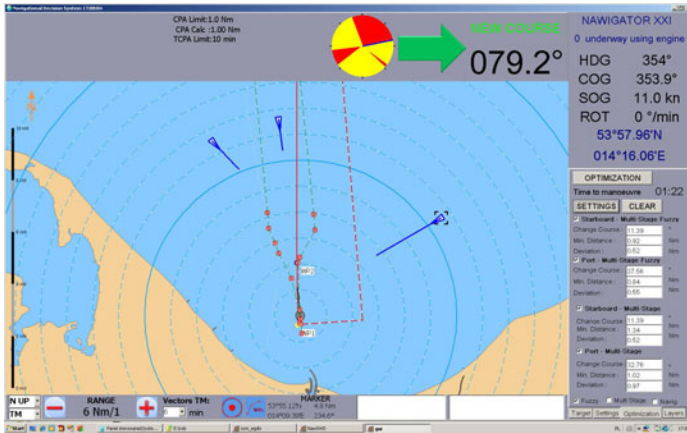


Fig. 3. Safe trajectories of own ship - solutions determined by the method of multi-stage control in a fuzzy environment

4 Summary

Both the analysis and assessment of ship encounter situations and solutions to collision situation generated by the navigational decision support system and suggested to the navigator should satisfy several requirements. These requirements should be in compliance with the regulations in force, guarantee performance of a safe manoeuvre and be rational.

Navigators criteria for analysis and assessment of a navigational situation and the choice of track (deviation) are often inaccurate in the sense of their uncertainties. The criteria presented herein conform with the criteria used by navigators. These criteria are described with the tools of fuzzy logic.

Optimal trajectories determined by the method of multi-stage control in a fuzzy environment, particularly the criterion of fuzzy closest point of approach, meet the requirements of safety, compliance with regulations and rationality. In practice, these are basic requirements that navigators will accept and make use of.

The presented solutions are applicable to navigation in the open sea. Further research will aim at extending the scope of the prototype system applicability to restricted areas as well.

References

1. Bellman, R.E., Zadeh, L.A.: Decision making in a fuzzy environment. *Management Science* 17 (1970)
2. Deo, N.: *The Theory of Graphs and its Application in Technology and Computer Science*. PWN Warszawa (1980) (in Polish)
3. Kacprzyk, J.: *Multi-stage fuzzy control*. WNT Warszawa (2001) (in Polish)

4. Pietrzykowski, Z.: Modelling of Decision Processes in Sea-Going Ship Movement Control. Studies, Maritime University of Szczecin 43 (2004) (in Polish)
5. Pietrzykowski, Z.: Fuzzy Control in Solving Collision Situations at Sea. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., urada, J. (eds.) Computational Intelligence: Methods and Applications, AOW EXIT, Warszawa, pp. 103–111 (2008)
6. Pietrzykowski, Z., Uriasz, J.: The ship domain - a criterion of navigational safety assessment in an open sea area. The Journal of Navigation 62, 93–108 (2009)
7. Pietrzykowski, Z., Magaj, J., Chomski, J.: A navigational decision support system for sea-going ships. Measurement Automation and Monitoring 10, 860–863 (2009)

A Hybrid Approach for Fault Tree Analysis Combining Probabilistic Method with Fuzzy Numbers

Julwan H. Purba¹, Jie Lu¹, Da Ruan², and Guangquan Zhang¹

¹ University of Technology, Sydney (UTS)
P.O. Box 123, Broadway, NSW 2007, Australia
{julwan, jielu, zhangg}@it.uts.edu.au

² Belgian Nuclear Research Centre (SCK•CEN)
Boeretang 200, 2400 Mol, Belgium
druan@sckcen.be

Abstract. Conventional fault tree analysis in safety analysis of complex engineering systems calculates the occurrence probability of the top undesired event using probabilistic failure rates. However, it is often very difficult to obtain those failure rates well in advance due to insufficient data, environment changing or new components. Fuzzy numbers can be applied to estimate failure rates by handling linguistic terms. This study proposes a hybrid approach of Fuzzy Numbers and Fault Tree Analysis to solve the conventional problem and describes its procedures using a case study of emergency core cooling system of a typical nuclear power plant.

Keywords: Fault tree analysis, probabilistic failure rate, fuzzy failure rate, safety analysis.

1 Introduction

Fault Tree Analysis (FTA) is widely used for safety analysis of complex engineering systems such as nuclear power plants (NPPs). Conventional FTA utilizes failure rates of every individual basic event constructing the tree to approximately calculate the occurrence probability of the top undesired event. However, it is often very difficult to exactly estimate the basic event failure rates due to insufficient data, environment changing or new components.

Fuzzy probabilities have been introduced to calculate the failure probability of a typical emergency cooling system using FTA [1]. The occurrence probability of the top undesired event is calculated using fuzzy multiplication rules for the Boolean AND gate and fuzzy complementation rules for the Boolean OR gate. However, this approach is limited only for trapezoidal fuzzy numbers. It also does not consider qualitative analysis and criticality analysis. The α -cut method has been introduced to calculate the failure probability of the reactor protective system (WASH-1400) [2]. All basic events are assumed to have probability distributions prior to designing triangular fuzzy numbers. The point

median value and the error factor are used to represent the lower bound and the upper bound of the triangular fuzzy numbers. The middle value of the triangular fuzzy numbers is represented by the point median value. However, this approach does not consider qualitative analysis prior to estimating the occurrence probability of the top event. A computational system analysis for FTA called FuzzyFTA has been developed and implemented to calculate the failure probability of Auxiliary Feedwater System (AFWS) of Angra-I Westinghouse NPP [3] and containment cooling system (CCS) of a typical four-loops pressurized water reactor [4]. However, this methodology is applicable only for triangular fuzzy numbers.

This paper proposes a Fuzzy Numbers based Fault Tree Analysis (FNFTA) approach. The approach combines probabilistic method with fuzzy numbers to estimate the failure probability of the top event. Fuzzy numbers can be in triangular and/or in trapezoidal form. The paper is organized as follows. The fuzzy numbers, fuzzy possibility scores and failure rates are explained in Section 2. Section 3 discusses the structure of the FNFTA approach and its procedures. In Section 4, a case study for NPPs shows the applicability of the approach. Finally, Section 5 summarizes future research tasks.

2 Fuzzy Numbers, Fuzzy Possibility Scores and Failure Rates

2.1 Fuzzy Numbers

A fuzzy number A is a subset of real line R whose membership function $f_A(x)$ can be continuously mapping from R into a closed interval $[0, w]$ where $0 \leq w \leq 1$ [5]. A normal fuzzy number A is expressed as follows.

$$f_A(x) = \begin{cases} f_A^L(x), & a \leq x \leq b \\ 1, & b \leq x \leq c \\ f_A^R(x), & c \leq x \leq d \\ 0, & otherwise \end{cases}$$

where $f_A^L(x):[a, b] \rightarrow [0, 1]$ and $f_A^R(x):[c, d] \rightarrow [0, 1]$. If both $f_A^L(x)$ and $f_A^R(x)$ are linear, then the fuzzy number A is a trapezoidal fuzzy number and usually denoted by $A = (a, b, c, d)$. When $b = c$, the trapezoidal fuzzy number becomes the triangular fuzzy number.

2.2 Fuzzy Possibility Scores

Fuzzy possibility score (FPS) is a crisp score that represents experts belief of the most likely score that an event may occur [6] [7]. Centroid-based distance method is to convert fuzzy numbers into fuzzy possibility scores [8]. The centroid (x_0, y_0) of the fuzzy number A is calculated as follows.

$$x_0(A) = \frac{\int_a^b (x \cdot f_A^L(x)) dx + \int_b^c x \cdot dx + \int_c^d (x \cdot f_A^R(x)) dx}{\int_a^b f_A^L(x) dx + \int_b^c dx + \int_c^d f_A^R(x) dx} \tag{1}$$

$$y_0(A) = \frac{\int_0^1 (y \cdot g_A^R(y)) dy - \int_0^1 (y \cdot g_A^L(y)) dy}{\int_0^1 g_A^R(y) dy - \int_0^1 g_A^L(y) dy} \tag{2}$$

where $g_A^R(y)$ and $g_A^L(y)$ are the inverse function of $f_A^R(x)$ and $f_A^L(x)$, respectively.

The FPS of fuzzy number A is the Euclidean distance of fuzzy number A , which is calculated from the origin to the centroid of the fuzzy numbers as follows.

$$FPS(A) = \sqrt{(x_0)^2 + (y_0)^2} \tag{3}$$

2.3 Fuzzy Failure Rates

Fuzzy failure rate (FFR) is an error rate which is obtained by dividing the frequency of an error with the total chance that an event may have error [7]. Onisawa proposed a logarithmic function to estimate the nature of human justification in [9]:

$$e = \frac{1}{1 + (K \times \log(1/E_m))^3} \tag{4}$$

where e is analogous to FPS and E_m is the most likely fault rate. K is a constant, which represents the safety criterion based on the lowest lower bound of the error rate and error rates of a routine. Onisawa defined that $K = 2.301$. E_m is represented by FFR, which is estimated in [10]:

$$FFR = \begin{cases} \frac{1}{10^m}, & FPS \neq 0 \\ 0, & FPS = 0 \end{cases} \tag{5}$$

where

$$m = \left[\frac{1 - FPS}{FPS} \right]^{\frac{1}{3}} \times 2.301$$

3 The Structure of the FNFTA Approach

The FNFTA approach consists of four analysis phases, which are system analysis phase, qualitative analysis phase, quantitative analysis phase, and criticality analysis phase (Fig. 1).

Phase 1: System Analysis Phase. System performance is analysed to build a fault tree describing failure modes that may occur to the system during its life time. This output becomes input to the second phase.

Phase 2: Qualitative Analysis Phase. Repeating events are removed from the fault tree by identifying cut sets and minimal cut sets. The output of this phase is a simplified fault tree which is equivalent to the fault tree obtained in the first phase but it is free from repeating events. This output becomes input to the third phase. Figure 2 shows the flowchart of this phase.



Fig. 1. The structure of the FNFTA approach

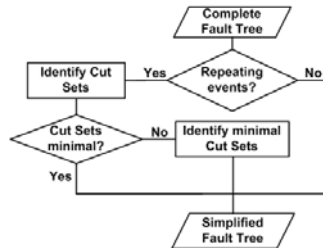


Fig. 2. The flowchart for qualitative analysis phase

Phase 3: Quantitative Analysis Phase. All basic events constructing the simplified fault tree are divided into two different evaluation techniques; probabilistic method and fuzzy numbers. The failure rates of basic events, which are evaluated using probabilistic method, can be obtained from reliable documents. The failure rates of other basic events are estimated using fuzzy numbers, which are designed based on expert justifications. These fuzzy numbers then are converted into FPS and finally into FFR. After all basic events have failure rates, Boolean algebra is used to estimate the occurrence probability of the top undesired event. The occurrence probability of the top event for two independent events for OR gate and AND gate are given in [11].

$$P_T = P_A + P_B - P_A \times P_B \tag{6}$$

$$P_T = P_A \times P_B \tag{7}$$

where P_A and P_B are the failure rates of basic event A and basic event B, respectively. The output of this phase is the occurrence probability of the top undesired event and is an input to the fourth phase. Figure 3 shows the flowchart of this phase.

Phase 4: Criticality Analysis Phase. This phase evaluates how far a basic event contributes to the occurrence of the top undesired event by calculating the criticality index of every basic event. Based on the calculated criticality index, the order of critical components can be justified. Fussell-Vesely (FV) importance

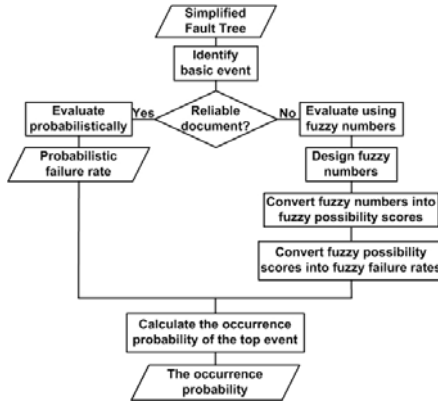


Fig. 3. The flowchart for quantitative analysis phase

algorithm as shown in (8) can be used to calculate the criticality index of a component [12].

$$FV_i = \frac{R_0 - R_i^-}{R_0} \tag{8}$$

where R_0 is the probability of the top event for overall basic events and R_i^- is the probability of the top event by setting the probability of basic event i to 0. Decision makers use this criticality index to improve the safety features of the analyzed NPP. Figure 4 shows the flowchart of this phase.

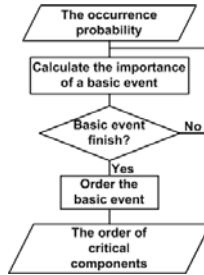


Fig. 4. The flowchart for criticality analysis phase

4 A Case Study for Nuclear Power Plants

The failure of a typical emergency core cooling system (ECCS) which is adopted from [1] is used to demonstrate the applicability of this approach. ECCS works to mitigate the consequences of a loss of coolant accident (LOCA). There are two operation modes of this ECCS, automatic operation and manual operation. The manual operation will only work when the automatic operation fails to mitigate the accident. Further details on this ECCS operation can be found in [1].

Phase 1: System Analysis Phase. In this case study, only the failure scenario of the automatic operation is analyzed. The fault tree describing the failure scenario of this operation mode is shown in Fig. 5. The top undesired event is the failure of automatic ECCS (FAECCS).

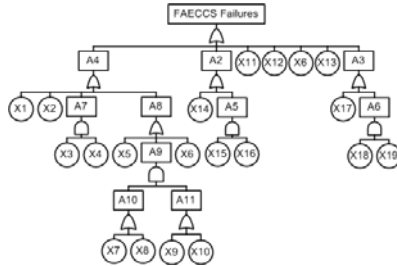


Fig. 5. The fault tree for FAECCS

Phase 2: Qualitative Analysis Phase. FAECCS in Fig. 5 still has one repeating event, which is a basic event X6. This repeating event can be removed using the combination of MOCUS algorithm and the rules of Boolean algebra to obtain cut sets and minimal cut sets [13].

The MOCUS algorithm evaluates events starting from the top events down to the bottom events. Every intermediate events found in the evaluation process will be substituted by the lower events. In Fig. 5, FAECCS can be represented by $(A4 + A2 + X11 + X12 + X6 + X13 + A3)$ where A4, A2 and A3 are intermediate events. A4 then can be represented by $(X1 + X2 + A7 + A8)$, A2 by $(X14 + A5)$ and A3 by $(X17 + A6)$. Therefore the FAECCS can be represented by $(X1 + X2 + A7 + A8 + X14 + A5 + X11 + X12 + X6 + X13 + X17 + A6)$. Using the same procedures for the next intermediate events found and by implementing the law of Boolean algebra absorption, the final representation of FAECCS are $(X1 + X2 + (X3 \cdot X4) + X5 + ((X7 + X8) \cdot (X9 + X10)) + X6 + X14 + (X15 \cdot X16) + X11 + X12 + X13 + X17 + (X18 \cdot X19))$ where symbol dot (.) represents AND gate and symbol plus (+) represents OR gate in the tree. This final representation as depicted in Fig. 6 is a simplified fault tree of the fault tree in Fig. 5.

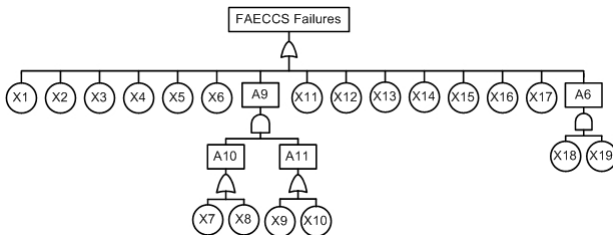


Fig. 6. The simplified fault tree

Phase 3: Quantitative Analysis Phase. In this phase, basic events are classified into two evaluation groups, which are probabilistic method and fuzzy numbers. For illustrative purposes only, we just simply evaluate basic events X1, X2, X3, X4, X5, X6, X7, X8, X11, and X12 using probabilistic method and all these events are assumed to have probabilistic failure rates of 1.5×10^{-4} . Other basic events are evaluated using fuzzy numbers. Basic events X9, X10, X12, X13, and X14 are evaluated using triangular fuzzy numbers of (0.1,0.15,0.25,0.3); meanwhile basic events X15, X16, X17, X18, and X19 are evaluated using trapezoidal fuzzy numbers of (0.1,0.25,0.3).

By solving (1) and (2) and substitute $a = 0.1$, $b = 0.15$, $c = 0.25$ and $d = 0.3$ for trapezoidal fuzzy numbers, we obtain $x_0 = 0.2$ and $y_0 = 0.444444$. FPS is then calculated using (3) and the result is 0.48737. Finally, this FPS is substituted into (5) to calculate FFR and the result is 4.57×10^{-3} . With the same procedures, FFR for the triangular fuzzy numbers is 2.275×10^{-3} .

Using (6), the occurrence probability of intermediate event A10 and A11 are 2.0×10^{-4} and 4.54×10^{-3} , respectively. Using (7), the occurrence probability of intermediate event A9 is 1.36×10^{-6} . With the same procedures, the occurrence probability of FAECCS is 2.1414×10^{-2} .

Phase 4: Criticality Analysis Phase. Using (8), the FV importance of basic event X1 is:

$$FV_{X1} = \frac{0.021413921 - 0.02126711}{0.021413921} = 6.8558 \times 10^{-3}$$

The FV importances of other basic events in Table 1 are calculated using the same procedures. Basic events X15, X16 and X17 are the most critical components followed by basic events X12, X13 and X14. These basic events need to be redesigned or changed in order to improve the reliability of FAECCS.

Table 1. The FV importance and the critical order of basic events

Component	FV importance	Critical order
X1,X2,X3,X4,X5,X6,X11	6.8558E-3	3
X7,X8,X9,X10	3.1149E-5	5
X12,X13,X14	1.0420E-1	2
X15,X16,X17	2.0980E-1	1
X18,X19	9.5443E-4	4

5 Summary and Future Works

Three advantages are gained from this FNFTA approach. (1) Probabilistic failure rate can be combined with fuzzy numbers to solve the limitation of the conventional FTA. (2) The calculation of the failure probability of the top undesired event is more accurate than the previous fuzzy approach because cut sets and minimal cut sets are evaluated first in the qualitative analysis phase prior to the quantitative analysis phase. (3) The approach can be used to estimate

the critical components and therefore decision makers can redesign or change critical components to improve the safety features of the system being analyzed. For future works, the conversion functions from fuzzy numbers into FFR need to be investigated in real-case studies.

Acknowledgement

The work presented in this paper was partially supported by the Australian Research Council (ARC) Discovery Grant PD0880739.

References

1. Misra, K.B., Weber, G.G.: Use of fuzzy set theory for level-I studies in probabilistic risk assessment. *Fuzzy Sets and Systems* 37(2), 139–160 (1990)
2. Suresh, P.V., Babar, A.K., Venkat Raj, V.: Uncertainty in fault tree analysis: A fuzzy approach. *Fuzzy Sets and Systems* 83(2), 135–141 (1996)
3. Guimaraes, A.C.F., Ebecken, N.F.F.: FuzzyFTA: A fuzzy fault tree system for uncertainty analysis. *Annals of Nuclear Energy* 26(6), 523–532 (1999)
4. Guimaraes, A.C.F., Lapa, C.M.F.: Parametric fuzzy study for effects analysis of age on PWR containment cooling system. *Applied Soft Computing* 8(1), 1562–1571 (2008)
5. Dubois, D., Prade, H.: Operations on fuzzy numbers. *International Journal of Systems Science* 9, 613–626 (1978)
6. Yuhua, D., Datao, Y.: Estimation of failure probability of oil and gas transmission pipelines by fuzzy fault tree analysis. *Journal of Loss Prevention in the Process Industries* 18, 83–88 (2005)
7. Lin, C.T., Wang, M.J.J.: Hybrid fault tree analysis using fuzzy sets. *Reliability Engineering & System Safety* 58(3), 205–213 (1997)
8. Wang, Y.M., Yang, J.B., Xu, D.L., Chin, K.S.: On the centroids of fuzzy numbers. *Fuzzy Sets and Systems* 157(7), 919–926 (2006)
9. Onisawa, T.: An approach to human reliability in man-machine systems using error possibility. *Fuzzy Sets and Systems* 27(2), 87–103 (1988)
10. Pan, N.F., Wang, H.: Assessing failure of bridge construction using fuzzy fault tree analysis. In: *IEEE International Conference on Fuzzy Systems and Knowledge Discovery*, Haikou, China (2007)
11. Haimes, Y.Y.: Fault trees. In: *Risk modeling, assessment and management*, pp. 525–569. John Wiley & Sons, Inc., New Jersey (2004)
12. Vinod, G., Kushwaha, H.S., Verma, A.K., Srividya, A.: Importance measures in ranking piping components for risk informed in-service inspection. *Reliability Engineering & System Safety* 80(2), 107–113 (2003)
13. Ericson, C.A.: Fault tree analysis. In: *Hazard analysis techniques for system safety*, pp. 183–221. John Wiley & Sons, Inc., Virginia (2005)

Imputing Missing Values in Nuclear Safeguards Evaluation by a 2-Tuple Computational Model

Rosa M. Rodríguez¹, Da Ruan², Jun Liu³, Alberto Calzada¹, and Luis Martínez^{1,*}

¹ University of Jaén, (Jaén-Spain)

{rmrodrig,martin,acalzada}@ujaen.es

² Belgium Nuclear Research Centre (SCK • CEN), (Mol-Belgium)

druan@sckcen.be

³ University of Ulster, (Northern Ireland-UK)

j.liu@ulster.ac.uk

Abstract. Nuclear safeguards evaluation aims to verify that countries are not misusing nuclear programs for nuclear weapons purposes. Experts of the International Atomic Energy Agency (IAEA) evaluate many indicators by using diverse sources, which are vague and imprecise. The use of linguistic information has provided a better way to manage such uncertainties. However, missing values in the evaluation are often happened because there exist many indicators and the experts have not sufficient knowledge or expertise about them. Those missing values might bias the evaluation result. In this contribution, we provide an imputation process based on collaborative filtering dealing with the linguistic 2-tuple computation model and a trust measure to cope with such problems.

Keywords: missing values, nuclear safeguards, fuzzy sets, imputation, trust worthy.

1 Introduction

Nuclear safeguards are a set of activities accomplished by the International Atomic Energy Agency (IAEA) in order to verify that a State is living up to its international undertakings not to use nuclear programs for nuclear weapons purposes. The safeguards system is based on assessments of the correctness and completeness of the State's declarations to the IAEA concerning nuclear material and related nuclear activities [7]. As a part of the efforts to strengthen international safeguards, including its ability to provide credible assurance of the absence of undeclared nuclear material and activities, IAEA uses large amounts and different types of information about States' nuclear and related nuclear activities.

IAEA evaluates nuclear safeguards by using a hierarchical assessment system that assesses indicators about critical activities [6] in the nuclear fuel cycle and processes required for their activities. According to the existence of the processes is inferred a decision about the development of nuclear proposes for weapon purposes.

* Corresponding author.

Experts and IAEA evaluate indicators on the basis of their analysis of the available information sources such as declarations of States, on-site inspections, IAEA non-safeguards databases [13,14]. This information is often uncertain and hard to manage by experts and it might happen that experts cannot provide either evaluations about some indicators or do so in an accurate way. In the latter case, the fuzzy linguistic approach [21] to deal with uncertain information provided good results [13]. However, when experts cannot provide all assessments for indicators, it is necessary to manage these missing values. To do this, there exist different ways in the literature, such as deletion, imputation or using as it is [8,17,18,19].

This contribution aims to present an imputation model based on collaborative filtering and the linguistic 2-tuple computational model to deal with missing values in nuclear safeguards. The imputed values are not real ones, therefore, we also introduce a trust measure to clarify the trustworthy of the imputed values and of the final result.

This paper is structured as follows: In Section 2, we review some related works in nuclear safeguards problems. In Section 3, we briefly outline a linguistic background that will be used in our model. In Section 4, we propose both an imputation model and a trust measure. In Section 5, we show a numerical example of the proposed approach, and finally, we conclude the research in Section 6.

2 Related Works

Different approaches for the evaluation and synthesis of nuclear safeguards have been proposed. We have focused on those that deal with nuclear safeguards problems and have briefly reviewed some of them. In [6] the IAEA Physical Model provides a structure for organizing the safeguards relevant information, which is used by IAEA experts to evaluate in a better way the safeguards significance of information on some State's activities. In [13] an evaluation model for the treatment of nuclear safeguards used linguistic information based on a hierarchical analysis of States under activities in a multi-layer structure. This proposal is the basis of our model in this contribution. The hierarchical model based on the IAEA Physical Model is divided into several levels with lower complexity, from which a global assessment by using a multi-step linguistic aggregation process is obtained. A latest proposal to manage the nuclear safeguards problem was proposed in [14], where a framework for modeling, analysing and synthesising information on nuclear safeguards under different types of uncertainty was presented. Such a framework makes use of the multi-layer evaluation model presented in [13] and a new inference model based on a belief inference methodology (RIMER) to handle hybrid uncertain information in nuclear safeguards evolution process. Recently, the focus on nuclear safeguards has moved to the management of missing values [9]. Different proposals about dealing with missing values for various purposes have been published in the literature [3,16,18,20]. We have focused our proposal on this problem by using a collaborative filtering view.

3 Linguistic Approach Background

Nuclear safeguards deals with a huge amount of information that usually involves uncertainties, which are mainly related to human cognitive processes. The use of linguistic

information has provided good results for managing such types of information in nuclear processes [13,15]. We shall use the linguistic modeling for nuclear safeguards evaluation by extending the work in [13,14]. The Fuzzy Linguistic Approach [21] represents the information with linguistic descriptors and their semantics. The former can be chosen by supplying a term set distributed on a scale with an order [2]. For example, a set of seven terms, S , could be:

$$S = \{s_0 : \text{nothing}(n), s_1 : \text{very low}(vl), s_2 : \text{low}(l), s_3 : \text{medium}(m), s_4 : \text{high}(h), \\ s_5 : \text{very high}(vh), s_6 : \text{perfect}(p)\}$$

Usually, in these cases, it is required that in the linguistic term set there exist the operators: (i) Negation: $\text{Neg}(s_i) = s_j$ such that $j = g - i$ ($g + 1$ is the cardinality), (ii) Maximization: $\max(s_i, s_j) = s_i$ if $s_i \geq s_j$, (iii) Minimization: $\min(s_i, s_j) = s_i$ if $s_i \leq s_j$.

The semantics of the terms is represented by fuzzy numbers defined in the interval $[0, 1]$, described by membership functions. A way to characterize a fuzzy number is to use a representation based on parameters of its membership function [2].

The use of linguistic information implies processes of computing with words (CW). There exist different computational models to accomplish them. We use the 2-tuple computational model presented in [5] to improve the precision of such processes of CW.

The 2-tuple model represents the linguistic information by means of a pair of values, called 2-tuple, (s_i, α) , where s_i is a linguistic term and α is a numerical value representing the symbolic translation.

Definition 1. *The symbolic translation is a numerical value assessed in $[-0.5, 0.5]$ that supports the “difference of information” between a counting of information β assessed in the interval of granularity $[0, g]$ of the term set S and the closest value in $\{0, \dots, g\}$ which indicates the index of the closest linguistic term in S .*

This linguistic model defines a set of functions to carry out transformations between numerical values and 2-tuple [5].

$$\Delta : [0, g] \longrightarrow S \times [-0.5, 0.5] \\ \Delta(\beta) = (s_i, \alpha), \text{ with } \begin{cases} i = \text{round}(\beta), \\ \alpha = \beta - i, \end{cases} \quad (1)$$

where *round* is the usual round operation, s_i has the closest index label to β , and α is the value of the symbolic translation.

We note that Δ is bijective [5] and $\Delta^{-1} : S \times [-0.5, 0.5] \longrightarrow [0, g]$ is defined by $\Delta^{-1}(s_i, \alpha) = i + \alpha$. In this way, the 2-tuple of S is identified with the numerical values in the interval $[0, g]$.

Besides, together this representation model, a computational model based on the functions aforementioned was also introduced in [5].

4 A Model to Impute Missing Values in Nuclear Safeguards

So far, the main interest in nuclear safeguards has been focused on the development of evaluation processes whose general structure is shown in Fig. 1. The sub-factors are

aggregated for levels and then, these results are aggregated again to obtain a global assessment. However, recently the focus on nuclear safeguards evaluation has moved to the treatment of missing values [9], because it has been noted that such a treatment is the key for obtaining reliable results. The treatment of missing values can be considered in different ways: deletion, imputation and using as it is [8][7][19]. We are interested in an imputation process, in the literature can be found different imputation processes for different proposes [3][16][18][20]. Here, we propose an imputation process for imputing missing values in nuclear safeguards based on collaborative filtering and a trust measure to compute the reliability of the imputed values. Hence, the general structure of the nuclear safeguards evaluation process is extended according to the Fig. 2.

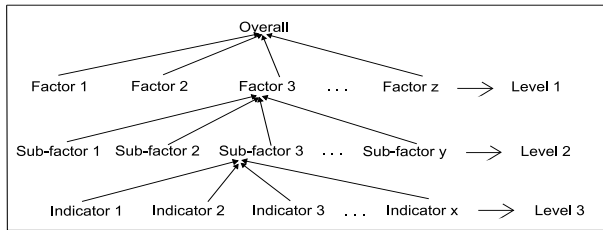


Fig. 1. Structure of the overall evaluation

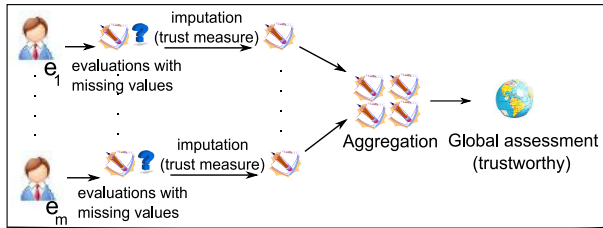


Fig. 2. General steps for nuclear safeguards evaluation with missing values

In the following section we present the CF process to estimate the imputed values and define a trust measure to compute the trustworthiness of such imputed values in order to know how reliable will be the final result.

4.1 Imputation Process

The imputation process is based on a k-NN scheme and an estimation similar to the process used by a collaborative recommender system [14] (see Fig. 3).

The main idea is to group the indicators according to their similarities by using a similitude measure on expert assessments. To do this, a new proposal for nuclear safeguards which utilises a collaborative filtering technique based on the k-NN algorithm (K nearest neighbours) is applied. To obtain those similarities is used the cosine distance [11] (see Eq. 1). In order to impute a plausible value for the missing one of an

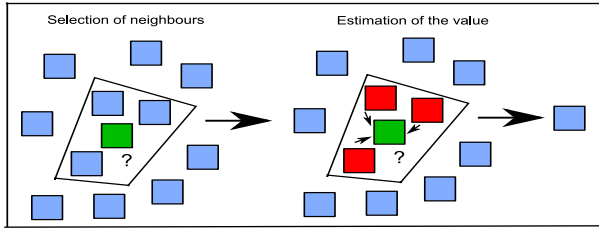


Fig. 3. Scheme of a Collaborative Recommender System

indicator is used the weighted mean (see Eq. 2). This imputation is expressed by means of a linguistic 2-tuple [5] to improve the precision of the linguistic imputed value.

$$w(i, j) = \cos(\bar{v}_i \bar{v}_j) = \frac{\bar{v}_i \bar{v}_j}{\|\bar{v}_i\|^2 \|\bar{v}_j\|^2} \quad (1) \quad v(e, i) = \frac{\sum_{j=1}^{j=k} w_{i,j} v_{e,j}}{\sum_{j=1}^{j=k} w_{i,j}} \quad (2)$$

In Eq. 1 and Eq. 2 i and j are indicators and e the expert who has not provided his/her assessment.

The imputed values are not real expert’s values, but rather an approximation to them. So, there exist some imprecision. Different precision metrics for collaborative filtering can be found in the literature such as MAE (Mean Absolute Error) [12], ROC [10] and so on. These metrics compute an error between the imputed values and real values. But that is not enough in our problem, because we need to know how trustworthy are those imputed values rather than its possible error.

4.2 Trustworthy of the Imputed Values

Therefore, in order to know the trustworthy of the imputed values computed by the previous imputation process, we define the following trust measure:

Definition 2. Assume that an expert provides his/her vector of linguistic assessments for the m indicators in nuclear safeguards evaluation, $X = \{x_1, \dots, x_m\}, x_i \in S = \{s_0, \dots, s_g\}$ and, there exists a set of missing values $\bar{X} = \{\bar{x}_1, \dots, \bar{x}_n\} \subseteq X$, that has been imputed in the imputation process. The trustworthy of an imputed value, $T(\bar{x}_j)$, is defined as:

$$T(\bar{x}_j) = (1 - \bar{S}_j)V + \bar{S}_j \frac{k}{K}, \quad T(\bar{x}_j) \in [0, 1] \quad (2)$$

where $\bar{S}_j = \frac{\sum_{l=1}^{l=k} w(j,l)}{k}$ is the arithmetic mean of the similarities among the indicator, j , and the k nearest indicators. And $V = \frac{g-sd(x_j)}{g}$, being sd the standard deviation of the assessments used to compute the imputed value and $g + 1$ the granularity of S . Eventually k indicates the real number of neighbours involved in the computation of \bar{x}_j from the initial K computed by the k -NN algorithm.

The definition of $T(\bar{x}_j)$ is based on the different cases of study whose results show that the more assessments are used to compute the imputed value the more trustworthy is.

Similarly, the more homogeneous are the assessments the more reliable is the imputed value. Thus, the more $T(\bar{x}_I)$ the more reliable is the imputed value. This measure will be used in the process presented in Fig. 3 in order to obtain the trustworthiness of the global assessment.

5 A Case Study

Here we present a case study that shows the results obtained by the proposed imputation model in a reduced dataset of four experts and 22 indicators in a nuclear safeguards problem (see Table 1). For the imputation model we have fixed the following parameters: $K=15$, the similarity measure is calculated by utilizing the cosine distance (Eq. 1) and for the imputation algorithm the weighted mean (Eq. 2). Therefore, the model imputes values and the results obtained are shown in Table 2. Additionally, in Table 2 is shown the trustworthiness of each imputed value as well.

Table 1. Experts evaluations

ind.	1	2	3	4	5	6	7	8	9	10	11
e1	h	p	vh	m	?	h	m	p	m	l	?
e2	l	vh	?	m	l	m	l	vl	l	l	m
e3	h	h	p	vh	m	vh	vh	h	?	m	vh
e4	p	p	p	?	p	m	vh	vh	h	h	m
ind.	12	13	14	15	16	17	18	19	20	21	22
e1	vl	p	vh	?	l	h	m	m	p	h	m
e2	m	p	m	l	l	m	m	?	m	m	vl
e3	l	p	p	vl	l	p	m	vh	p	vh	?
e4	?	vh	h	l	m	?	l	vh	h	vh	l

Once we have obtained the imputed values and their trustworthiness, we keep applying the safeguards process shown in Fig. 2 based on [13], in which the experts assessments are synthesized in a multi-step aggregation process to obtain a global value and its trustworthiness is also obtained by aggregating the trustworthiness of each assessment. To obtain the global assessment, first it is computed a collective assessment for these indicators and then, a global assessment is obtained by aggregating such collective assessments. In this case, we have obtained a linguistic 2-tuple for the nuclear safeguards result (**high, 0.23**) with a trustworthiness **0.86**.

If we use the safeguards model presented in Fig. 1 without the imputation of missing values the result obtained is (**medium, 0.33**). We can then observe the relevance of the missing values. Therefore, the treatment of such values must be a cornerstone in nuclear safeguards evaluation.

Table 2. Predictions and trust measures

ind.	e1	e2	e3	e4	
	pred.	trust pred.	trust pred.	trust pred.	trust
3		(m,-.154)	.866		
4				(h,.313)	.866
5	(h,-.46)	.863			
9			(vh,-.361)	.931	
11	(h,-.132)	.93			
12				(h,.242)	.796
15	(m,.493)	.798			
17				(vh,-.459)	.866
19		(m,-.16)	.866		
22			(vh,-.4)	.858	

6 Conclusions

Nuclear safeguards evaluation is a complex problem where the experts use different sources of information to evaluate related indicators. This evaluation is usually inaccurate due to the uncertainty of the sources of information and the huge amount of information to manage. Such uncertainties and inaccuracies make that experts sometimes cannot provide assessments for all indicators appearing missing values. In this contribution, we have presented a linguistic nuclear safeguards evaluation model that manages these missing values by means of an imputation process based on a collaborative filtering algorithm. Additionally, we have provided a trust measurement to measure the goodness of the imputed values.

Acknowledgements

This work is partially supported by the Research Project TIN-2009-08286, P08-TIC-3548 and FEDER funds.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17(6), 734–749 (2005)
2. Bonissone, P.P., Decker, K.S.: Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-Off Precision and Complexity. In: Kanal, L.H., Lemmer, J.F. (eds.) *Uncertainty in Artificial Intelligence*. North-Holland, Amsterdam (1986)
3. Dubois, D., Prade, H.: Incomplete conjunctive information. *Computers and Mathematics with Applications* 15(10), 797–810 (1988)
4. Herlocker, J.L., Konstan, J.A., Riedl, J.: An Empirical Analysis of Design Choices in Neighborhood-based Collaborative Filtering Algorithms. In: *Information Retrieval*, pp. 287–310. Kluwer Academic, Dordrecht (2002)

5. Herrera, F., Martínez, L.: A 2-tuple fuzzy linguistic representation model for computing with words. *IEEE Transactions on Fuzzy Systems* 8(6), 746–752 (2000)
6. Physical model. Int. Atomic Energy Agency, IAEA, Vienna, Rep. STR-314 (1999)
7. Nuclear Security and Safeguards. In: IAEA Bulletin, Annual Report. IAEA, vol. 43 (2001)
8. Jiang, N., Gruenwald, L.: Estimating Missing Data in Data Streams. In: Kotagiri, R., Radha Krishna, P., Mohania, M., Nantajeewarawat, E. (eds.) *DASFAA 2007*. LNCS, vol. 4443, pp. 981–987. Springer, Heidelberg (2007)
9. Kabak, Ö., Ruan, D.: A cumulative belief-degree approach for nuclear safeguards evaluation. In: *Proc. of IEEE Conference on Systems, Man and Cybernetics, San Antonio, TX-USA (2009)*
10. Landgrebe, T.C.W., Duin, R.P.W.: Efficient multiclass ROC approximation by decomposition via confusion matrix perturbation analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30(5), 810–822 (2008)
11. Lee, T.Q., Park, Y., Pack, Y.T.: A Similarity Measure for Collaborative Filtering with Implicit Feedback. In: Huang, D.-S., Heutte, L., Loog, M. (eds.) *ICIC 2007*. LNCS (LNAI), vol. 4682, pp. 385–397. Springer, Heidelberg (2007)
12. Lin, J.H., Sellke, T.M., Coyle, E.J.: Adaptive stack filtering under the mean absolute error criterion. In: *Advances in Communications and Signal Processing*, pp. 263–276 (1989)
13. Liu, J., Ruan, D., Carchon, R.: Synthesis and evaluation analysis of the indicator information in nuclear safeguards applications by computing with words. *International Journal of Applied Mathematics and Computer Science* 12(3), 449–462 (2002)
14. Liu, J., Ruan, D., Wang, H., Martínez, L.: Improving nuclear safeguards evaluation through enhanced belief rule-based inference methodology. *Int. J. Nuclear Knowledge Management* 3(3), 312–339 (2009)
15. Martínez, L.: Sensory evaluation based on linguistic decision analysis. *International Journal of Approximated Reasoning* 44(2), 148–164 (2007)
16. Nowicki, R.: On combining neuro-fuzzy architectures with the rough set theory to solve classification problems with incomplete data. *IEEE Transactions on Knowledge and Data Engineering* 20(9), 1239–1253 (1989)
17. Oltman, L.B., Yahia, S.B.: Yet another approach for completing missing values. In: Yahia, S.B., Nguifo, E.M., Belohlavek, R. (eds.) *CLA 2006*. LNCS (LNAI), vol. 4923, pp. 155–169. Springer, Heidelberg (2008)
18. Pawlak, M.: Kernel classification rules from missing data. *IEEE Transactions on Information Theory* 39(3), 979–988 (1993)
19. Siddique, J., Belin, R.: Using and approximate bayesian bootstrap to multiply impute nonignorable. *Computational Statistics and Data Analysis* 53(2), 405–415 (2008)
20. Slowinski, R., Stefanowski, J.: Rough classification in incomplete information-systems. *Mathematical and Computer Modelling* 12(10-11), 1347–1357 (1989)
21. Zadeh, L.A.: The concept of a linguistic variable and its applications to approximate reasoning. *Information Sciences, Part I, II, III* 8, 8, 9, 199–249, 301–357, 43–80 (1975)

Neuro-fuzzy Systems with Relation Matrix

Rafał Scherer

¹ Academy of Management (SWSPiZ), Institute of Information Technology
ul. Sienkiewicza 9, 90-113 Łódź, Poland

<http://www.swspiz.pl/>

² Department of Computer Engineering, Częstochowa University of Technology
al. Armii Krajowej 36, 42-200 Częstochowa, Poland

rafal@ieee.org

<http://kik.pcz.pl>

Abstract. Neuro-fuzzy systems are eagerly used for classification and machine learning problems. Researchers find them easy to use because the knowledge is stored in the form of the fuzzy rules. The rules are relatively easy to create and interpret for humans, unlike in the case of other learning paradigms e.g. neural networks. The most commonly used neuro-fuzzy systems are Mamdani (linguistic) and Takagi Sugeno systems. There are also logical-type systems which are well suited for classification tasks. In the paper, another type of fuzzy systems is proposed, i.e. multi-input multi-output systems with additional binary relation for greater flexibility. The relation bonds input and output fuzzy linguistic values. Thanks to this, the system is better adjustable to learning data. The systems have multiple outputs which is crucial in the case of classification tasks. Described systems are tested on several known benchmarks and compared with other machine learning solutions from the literature.

1 Introduction

Fuzzy and neuro-fuzzy systems are often used in various science, economics, manufacturing and medicine applications [8], [9], [13], [14]. Neuro-fuzzy systems (NFS) are synergistic fusion of fuzzy logic and neural networks, hence NFS can learn from data and retain the inherent interpretability of fuzzy logic. The knowledge in the form of fuzzy rules is transparent and high accuracy performance is achieved. Different forms of fuzzy rules and inference methods constitute various types of NFS. They act similarly to neural networks and most of them are universal approximators. Therefore NFS can perform well in tasks of classification, prediction, approximation and control, similarly to neural networks. In neural networks it is very difficult to tell where the knowledge is exactly stored and they can not use prior knowledge. NFS can use almost the same learning methods and achieve the same accuracy as neural networks yet the knowledge in the form of fuzzy rules is easily interpretable for humans. Most popular NFS are Mamdani type linguistic NFS, where consequents and antecedents are related by the min operator or generally by a t-norm. Takagi Sugeno systems with consequents being functions of inputs are also often used. Less common are logical type linguistic NFS. Here consequents and antecedents are related by fuzzy implications, e.g. binary, Łukasiewicz, Zadeh.

Another approach, rarely studied in the literature, is based on fuzzy relational systems (see e.g. [5], [12], [15], [16]). The systems have additional fuzzy relation which binds input fuzzy linguistic values with output fuzzy linguistic values. Thanks to this the system is more flexible.

2 Fuzzy Systems with Additional Relation

Fuzzy systems considered in this section have an additional relation that binds input and output fuzzy sets. Rules in such systems have more than one linguistic value defined on the same output variable, in its consequent. Fuzzy rules in a MIMO relational model have the following form

$$R^k : \text{IF } \mathbf{x} \text{ is } A^k \text{ THEN} \\ y_c \text{ is } B_c^1(r_{k1}), y_c \text{ is } B_c^m(r_{km}), \dots, y_c \text{ is } B_c^M(r_{kM}), \quad (1)$$

where r_{km} is a weight, responsible for the strength of connection between input and output fuzzy sets, c is the output number. Relational fuzzy systems store associations between the input and the output linguistic values in the form of a discrete fuzzy relation

$$\mathbf{R}_c(A, B) \in [0, 1] . \quad (2)$$

In case of a multi-input multi-output system (MIMO), the relation \mathbf{R}_c is a matrix containing degree of connection for every possible combination of input and output fuzzy sets for the output c . We consider a fuzzy system with multidimensional input linguistic values, where input fuzzy sets are the same for all outputs. Thus, we have only one set A of fuzzy linguistic values

$$A = \{A^1, A^2, \dots, A^K\} , \quad (3)$$

and the relational matrix \mathbf{R}_c is only two-dimensional, because every output has its own relation \mathbf{R}_c . Output variable y_c has a set of M_c linguistic values B_c^m with membership functions $\mu_{B_c^m}(y)$, for $m_c = 1, \dots, M_c$

$$B_c = \{B_c^1, B_c^2, \dots, B_c^M\} . \quad (4)$$

Sets A and B_c are related to each other with a certain degree by the $K \times M_c$ relation matrix

$$\mathbf{R}_c = \begin{bmatrix} r_{11} & r_{11} & \cdots & r_{1M_c} \\ r_{21} & r_{22} & \cdots & r_{2M_c} \\ \vdots & \vdots & r_{km} & \vdots \\ r_{K1} & r_{K2} & \cdots & r_{KM_c} \end{bmatrix} . \quad (5)$$

In this section we present neuro-fuzzy systems for classification.

Having given vector \bar{A} of K membership values $\mu_{A^k}(\bar{\mathbf{x}})$ for a crisp observed feature values $\bar{\mathbf{x}}$, vector \bar{B}_c of M_c crisp memberships μ_{m_c} is obtained through a fuzzy relational composition

$$\bar{B}_c = \bar{A} \circ \mathbf{R}_c , \quad (6)$$

implemented element-wise by a generalized form of sup-min composition [2], i.e. s-t composition

$$\mu_{mc} = \bigvee_{k=1}^K [\mathsf{T}(\mu_{A^k}(\bar{\mathbf{x}}), r_{km}^c)] . \quad (7)$$

The crisp output of the relational system is computed by the weighted mean

$$\bar{y}_c = \frac{\sum_{m=1}^{M_c} \{\bar{y}_c^m \bigvee_{k=1}^K [\mathsf{T}(\mu_{A^k}(\bar{\mathbf{x}}), r_{km}^c)]\}}{\sum_{m=1}^{M_c} \bigvee_{k=1}^K [\mathsf{T}(\mu_{A^k}(\bar{\mathbf{x}}), r_{km}^c)]} , \quad (8)$$

where \bar{y}_c^m is a centre of gravity (centroid) of the fuzzy set B_c^m . The neuro-fuzzy structure of the relational system is depicted in Fig. 1. Indexes c are omitted for simplicity. The first layer of the system consists of K multidimensional fuzzy membership functions. The second layer is responsible for s-t composition of membership degrees from previous layer and KM crisp numbers from the fuzzy relation. Finally, the third layer realizes center average defuzzification. Depicting the system as a net structure allows learning or fine-tuning system parameters through the backpropagation algorithm. S-norm in (8) can be replaced by OWA operators [20], [21] which further extends the versatility of the relational neuro-fuzzy system.

3 Numerical Simulations

We used two well known dataset taken from [1] to show the ability of the proposed systems to fit to data. The systems were initialized randomly by the fuzzy c -means clustering and then trained the backpropagation algorithm.

3.1 Iris Dataset

First numerical simulations were carried out on the Iris dataset. The set consists of 3 classes, with 150 instances (50 in each of three classes). The instances are described by four features: sepal length and width and petal length and width. Our system had three outputs, according to the number of classes (iris setosa, iris versicolor, and iris virginica). Each input had three fuzzy sets and each output subsystem had 3×3 relational matrix. The dataset was divided into 100 training instances and 50 testing ones. Classification results are provided in Table 1.

Table 1. Root mean square error for the Iris Dataset

Method	testing RMSE
Rutkowski [14]	97.78
Proposed approach	96.31

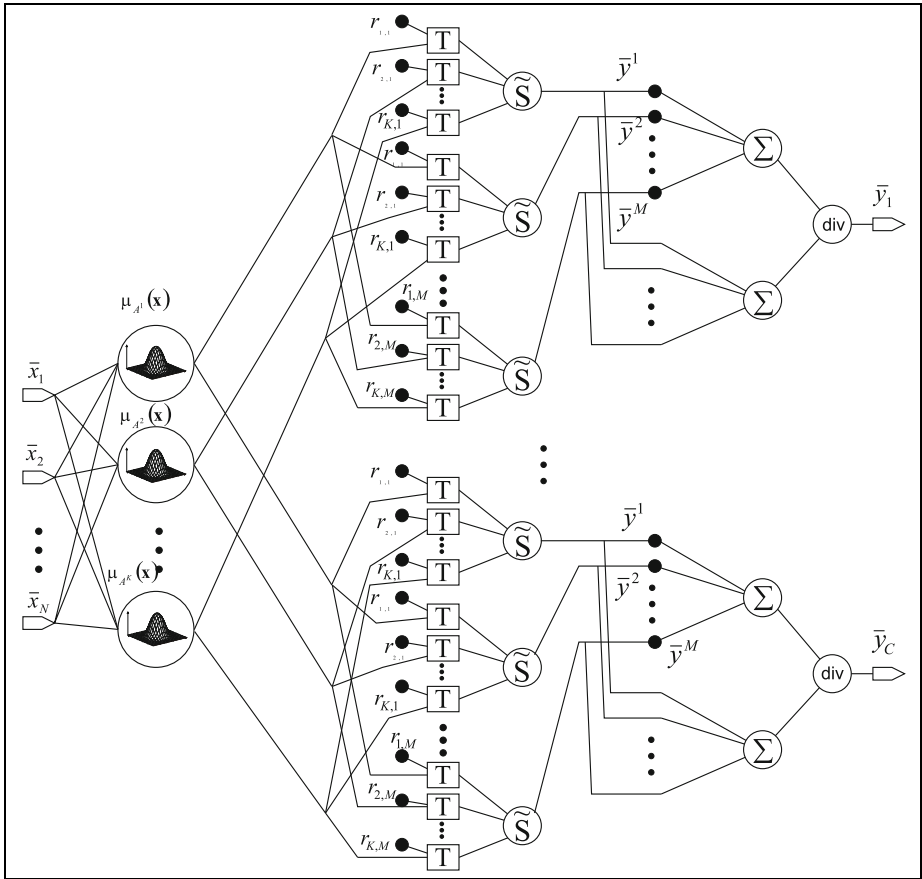


Fig. 1. The neuro-fuzzy structure of the relational system with multiple outputs. Indexes c denoting output number are omitted for simplicity.

3.2 Glass Identification Problem

The Glass Identification problem [1] consists in an identification of glass type on the basis of 9 attributes for forensic purposes. The dataset has 214 instances and each instance is described by nine attributes (RI: refractive index, Na: sodium, Mg: magnesium, Al: aluminium, Si: silicon, K: potassium, Ca: calcium, Ba: barium, Fe: iron). All attributes are continuous. There are two classes: the window glass and the non-window glass. In our experiments, all sets are divided into a learning sequence (150 sets) and a testing sequence (64 sets). The obtained accuracy is presented in Table 2.

Table 2. Root mean square error for Glass Identification problem

Method	testing RMSE
Dong et al [6]	93.10
Proposed approach	94.21

4 Conclusions

In the paper, we presented a new fuzzy relational system with multiple outputs. Rules in the system are more flexible because of the additional weights in rule consequents. The weights come from binary relations \mathbf{R}_e , one for each output. Multioutput relational fuzzy systems presented in the paper can be easily fitted to data by setting input and output linguistic terms and elements of the relation. Apart from purely data-driven designing we can use some expert knowledge in the form of fuzzy rules. Simulation results confirmed the system ability to fit to any data.

Acknowledgments

This work was partly supported by the Polish Ministry of Science and Higher Education (Habilitation Project 2007-2010 Nr N N516 1155 33, Polish-Singapore Research Project 2008-2010 and Research Project 2008-2011) and the Foundation for Polish Science – TEAM project 2010-2014.

References

1. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine (2007), <http://www.ics.uci.edu/~mllearn/MLRepository.html>
2. Babuska, R.: Fuzzy Modeling For Control. Kluwer Academic Press, Boston (1998)
3. Bezdek, J.C., Pal, S.K.: Fuzzy Models for Pattern Recognition. IEEE Press, New York (1992)
4. Bezdek, J., Keller, J., Krisnapuram, R., Pal, N.R.: Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. Kluwer Academic Press, Dordrecht (1999)
5. Branco, P.J.C., Dente, J.A.: A Fuzzy Relational identification Algorithm and its Application to Predict the Behaviour of a Motor Drive System. Fuzzy Sets and Systems 109, 343–354 (2000)
6. Dong, M., Kothari, R.: Look-ahead based fuzzy decision tree induction. IEEE Trans. on Fuzzy Systems 9, 461–468 (2001)
7. Ischibuchi, H., Nakashima, T.: Effect of Rule Weights in Fuzzy Rule-Based Classification Systems. IEEE Transactions on Fuzzy Systems 9(4), 506–515 (2001)
8. Jang, R.J.-S., Sun, C.-T., Mizutani, E.: Neuro-Fuzzy and Soft Computing. A Computational Approach to Learning and Machine Intelligence. Prentice Hall, Upper Saddle River (1997)
9. Nauck, D., Klawon, F., Kruse, R.: Foundations of Neuro - Fuzzy Systems. John Wiley, Chichester (1997)
10. Nauck, D., Kruse, R.: How the Learning of Rule Weights Affects the Interpretability of Fuzzy Systems. In: Proceedings of 1998 IEEE World Congress on Computational Intelligence, FUZZ-IEEE, Alaska, pp. 1235–1240 (1998)
11. Nozaki, K., Ischibuchi, H., Tanaka, K.: A simple but powerful heuristic method for generating fuzzy rules from numerical data. Fuzzy Sets and Systems 86, 251–270 (1995)
12. Pedrycz, W.: Fuzzy Control and Fuzzy Systems. Research Studies Press, London (1989)
13. Pedrycz, W., Gomide, F.: An Introduction to Fuzzy Sets, Analysis and Design. The MIT Press, Cambridge (1998)
14. Rutkowski, L.: Flexible Neuro Fuzzy Systems. Kluwer Academic Publishers, Dordrecht (2004)

15. Scherer, R., Rutkowski, L.: Relational Equations Initializing Neuro-Fuzzy System. In: 10th Zittau Fuzzy Colloquium, Zittau, Germany (2002)
16. Scherer, R., Rutkowski, L.: Neuro-Fuzzy Relational Systems. In: 2002 International Conference on Fuzzy Systems and Knowledge Discovery, Singapore, November 18-22 (2002)
17. Setness, M., Babuska, R.: Fuzzy Relational Classifier Trained by Fuzzy Clustering. *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics* 29(5), 619–625 (1999)
18. Sugeno, M., Yasukawa, T.: A Fuzzy-Logic-Based Approach to Qualitative Modeling. *IEEE Transactions on Fuzzy Systems* 1(1), 7–31 (1993)
19. Wang, L.-X.: *Adaptive Fuzzy Systems And Control*. PTR Prentice Hall, Englewood Cliffs (1994)
20. Yager, R.R., Filev, D.P.: *Essentials of Fuzzy Modeling and Control*. John Wiley & Sons Inc., New York (1994)
21. Yager, R.R., Filev, D.P.: On a Flexible Structure for Fuzzy Systems Models. In: Yager, R.R., Zadeh, L.A. (eds.) *Fuzzy Sets, Neural Networks and Soft Computing*, Van Nostrand Reinhold, New York, pp. 1–28 (1994)

Fuzzy Multiple Support Associative Classification Approach for Prediction

Bilal Sowan, Keshav Dahal, and Alamgir Hussain

School of Computing, Informatics and Media
University of Bradford, Bradford BD7 1DP, UK
{b.sowan,k.p.dahal,m.a.hossain1}@Bradford.ac.uk

Abstract. The fact of building an accurate classification and prediction system remains one of the most significant challenges in knowledge discovery and data mining. In this paper, a Knowledge Discovery (KD) framework is proposed; based on the integrated fuzzy approach, more specifically Fuzzy C-Means (FCM) and the new Multiple Support Classification Association Rules (MSCAR) algorithm. MSCAR is considered as an efficient algorithm for extracting both rare and frequent rules using vertical scanning format for the database. Consequently, the adaptation of such a process sufficiently minimized the prediction error. The experimental results regarding two data sets; Abalone and road traffic, show the effectiveness of the proposed approach in building a robust prediction system. The results also demonstrate that the proposed KD framework outperforms the existing prediction systems.

Keywords: Knowledge Discovery, MSapriori, Apriori, Fuzzy C-Means, Classification, Prediction.

1 Introduction

Data mining technology contains many tasks/techniques such as classification and association rules. Classification is the task of categorizing the training data into predefined groups or classes with the aim of building a classifier model. It is grouping the records in training data, each record contains a collection of attributes and one of the attributes is considered as the output (class label) [11], [15]. The prediction is the task of forecasting a new data (the new input data of an unknown output) based on the current classifier model.

The association rules mining has been studied widely during the last few years in order to improve the rules learning problem. Apriori algorithm [1] is one of the most significant approaches in mining association rules. Apriori algorithm extracts the association rules based on two measurement objectives. One measure "minimum support threshold" (*minsupp*) is applied to assess the minimum quantity of transactions data included in the frequent itemsets (rules). While the other measure called "minimum confidence threshold" (*minconf*) is adapted to assess the accuracy/predictive strength of rules. *minsupp* is viewed as an important key for a successful association rules mining.

In this manner, the use of a single *minsupp* for a whole database considers and assumes all attributes (items) in database with the same frequency, which causes a dilemma of rare items problem if *minsupp* is set too high. However, in real applications, database contains some items of a high frequency; while other ones are of a low frequency. To overcome the dilemma of rare item problem, Liu et al. [5] proposed an algorithm called MSapriroi. It is a generalization for Apriori algorithm in applying multiple *minsupp* called Multiple Item Support (single *minsupp* assigns for each item).

Most of the well-known classification techniques are based on heuristic/greedy strategy approaches such as the decision tree C4.5 [12] and Repeated Incremental Pruning to Produce Error Reduction (RIPPER) [4]. These approaches aim at discovering a small sub-set of rules to represent the training data and build a classifier [14]. The greedy approach is employed through a traditional classification technique, which operates in stages. In each stage, the solution taken seems to be efficient, without considering the next stages. However, these techniques which play a vital role in some cases suffer from comprehensive rules. The rules generated from these techniques are different nature as they are hardly understandable by the user. Consequently, the prediction of a class label (output value of the new input data) using heuristic approaches is viewed as a black-box [11], [15].

A variety of techniques is developed through the integration of association rules and classification known as associative classifications such as CBA [6], MCAR [15], and CSMC [8]. However, the rules are extracted by associative classification techniques and cannot be discovered through the traditional classification. Therefore, the integration of association rules and classification is a promising field of research.

The first approach which adapted association rules for classification proposed by Liu et al. [6] so-called Classification Based on Association (CBA), applied the popular Apriori algorithm. Their approach is able to extract the Classification Association Rules (CAR) and each rule belongs to a specific class label. CARs should satisfy both *minsupp* and *minconf*. The improvement of CBA called msCBA [7] through the usage of multiple class *minsupp* in the rule generation process instead of using single *minsupp*. msCBA applies the decision tree method to divide the training data for each classifier with the aim of reducing the error rate. A new associative classification technique called Multi-class Classification based on Association Rules (MCAR) was developed by Thabtah et al. [15]. This technique used a single *minsupp* and utilized the vertical format for scanning database and extracting associative classification rules. As a result, MCAR is more accurate than CBA. Most recently, a new classifier approach based on fuzzy association rules was proposed by Pach et al. [11]. It used the fuzzy approach for data discretization. Thus, the classification model is more easily interpretable than other models that are complex to be understood.

An approach was implemented by Lu et al. [9] comparing two methods for the prediction of one output value, which is applied in Abalone data set. The first method, FCM and Apriori algorithms are used to extract fuzzy association rules

while Genetic Algorithm (GA) is applied to tune the fuzzy sets. The second one is proceed as the first method but using variable thresholds in the prediction. The prediction accuracy of the first approach is better than the second one.

The above state-of-the-art suffers from one or more of the following issues:

- The single *minsupp* approach is not fair when using it for the whole data. Single *minsupp* assumes the same frequency for all items in data. In this context, the real application data possesses some items of a high frequency; while other ones possess low frequency. Therefore, using a single *minsupp* will be missing some of the significant association rules.
- Apriori fashion works in level-wise search, which needs to scan the database for each iteration (level). Hence, the technical performance is reduced.
- Association rule mining techniques produce many association rules affecting the prediction result in an advanced step.

In our previous work, we proposed a Knowledge Discovery (KD) framework for prediction incorporating FCM and Apriori approach using a single *minsupp*. In this paper, we improve the Knowledge Discovery (KD) framework as referred in our previous work [13] through the integration of Fuzzy C-Means (FCM) and the new associative classification algorithm (Multiple Support Classification Association Rules (MSCAR)), aims to overcome these issues. Hence, this integration produces Fuzzy Associative Classification Rules (FACRs) of rare and frequent items that will be used later for prediction. Applying FCM as a pre-processing step for the transformation of the quantitative data into fuzzy sets (terms), while adapting the associative classification approach to facilitate the direct pruning of unnecessary rules. These rules adapt the multiple support approach to deal with unbalanced data through a vertical scanning format for the database. FACRs can be stored in Knowledge Base (KB) to build Fuzzy Inference System (FIS). This framework is applied in two quantitative data sets for prediction namely; Abalone and road traffic. The experimental results related to both case studies show that the proposed framework discovers FACRs that efficiently consider rare and frequent rules compared to the existing approaches.

2 Proposed Methodology

Our proposed KD framework for prediction consists in two main phases: discovering FACRs rules based on MSCAR and predicting the future value based on FIS. The framework steps are explained as follows: (i) Getting the data from the database. This data is analyzed for the consistency, but some noisy data will be kept to ensure the ability of the proposed KD framework for handling the real data applications. (ii) Transforming the quantitative (continuous) data into fuzzy data by using FCM. (iii) Applying MSCAR algorithm to extract FACRs, and then saving these rules in KB. (iv) Using FIS to command the KB for a prediction (v) Testing the KD framework feasibility in two case studies; Abalone and road traffic. The distinctive features for KD framework are depicting as follows:

- FCM [2] is used as an automatic system to transform the quantitative (continuous) data set into fuzzy data sets (terms).
- The multiple support concept is applied for extracting both frequent and rare terms.
- The vertical scanning format is utilized to scan the database one time [16].
- The associative classification approach is applied to facilitate the direct pruning of unnecessary rules.

Let $FTI = \{FTi_1, FTi_2, \dots, FTi_n\}$ be a fuzzy terms (sets) of input fuzzy data, $FTo = \{FTo_1, FTo_2, \dots, FTo_m\}$ be a fuzzy terms (sets) of output fuzzy data and $FD = \{U_{V_1, FTi_n, FTo_m}, U_{V_2, FTi_n, FTo_m}, \dots, U_{V_{nd}, FTi_n, FTo_m}\}$ be a set of transactions U in the fuzzy database FD of the fuzzy input terms FTI and the fuzzy output terms FTo . Each transaction U in FD is formed by a set of FTi in FTI where $FTi \in FTI$ and FTo in FTo where $FTo \in FTo$. A strong fuzzy associative classification rule (FACR) is defined as $X.FTi \rightarrow Y.FTo$ where $X.FTi \subseteq FTI$, $Y.FTo \subseteq FTo$ and $X.FTi \cap Y.FTo = \phi$. i.e. the FACR as $FTi_1, FTi_2, \dots, FTi_n \rightarrow FTo_m$, assuming that $MFTS$ is a Minimum Fuzzy

Input: Fuzzy Database (FD) contains Fuzzy Terms (sets) FT , β value, Least Support (LS), $M(FT) = \beta * (FT.Support)$, $minconf$.

Output: A set of fuzzy associative classification rules (FACRs).

Method: Scan FD once to calculate the support value for each FT , and then store FT in VF .

$$MFTS(FT) = \begin{cases} M(FT), & \text{IF } M(FT) > LS \\ LS, & \text{Otherwise} \end{cases}$$

```

foreach  $FT$  in  $VF$  do
  foreach fuzzy input term  $FTi$  and fuzzy output term  $FTo$  in  $VF$  do
    Join up pair of  $FTi$ ,  $FTo$ 
    if  $\langle FTi, FTo \rangle > .Support$  pass  $minimum(MFTS(FTi), MFTS(FTo))$ 
    then
      | Insert in  $VFS$ 
    end
  end
  foreach pair of disjoint  $FTi$ ,  $FT(i+1)$  in  $VF$  do
    if  $\langle FTi, FT(i+1) \rangle > .Support$  pass  $minimum(MFTS(FTi), \dots)$  then
      | Insert in  $VF$ 
    end
  end
end
foreach FACR  $FTi \rightarrow FTo$  in  $VFS$  do
  Calculate the confidence value for each FACR
  if  $confidence\_value(FACR) = \frac{\sum minimum(FTi \cap FTo)}{\sum minimum(FTi)}$  pass  $minconf$  then
    | Insert in  $KB$ 
  end
end

```

Fig. 1. MSCAR algorithm

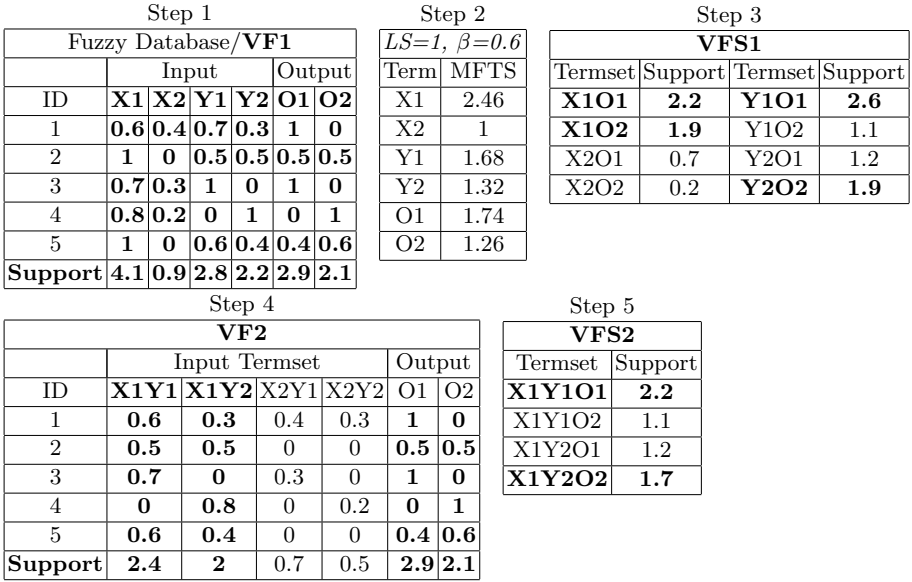


Fig. 2. An example of MSCAR

Term Support value for a fuzzy term of the input FTi and the output (class label) FTo , then the FACR is accepted if its support value is greater than or equal to the minimum value of $MFTS$ from all fuzzy input and output terms within each $FACR$ as $support_value(FACR) \geq minimum (MFTS(FTi_1), MFTS(FTi_2), \dots, MFTS(FTi_n), MFTS(FTo_m))$ and its confidence value is greater than or equal to $minconf$ as $confidence_value(FACR) \geq minconf$.

The MSCAR algorithm for extracting FACRs is presented in Fig. 1, and an example of MSCAR algorithm is illustrated in Fig. 2.

3 Experimental Results and Analysis

In order to demonstrate the technical feasibility of the proposed KD framework in a realistic scenario, we use the methodology discussed in the previous section in two case studies: the first case study regards the quantitative data set called Abalone while the second one is a quantitative road traffic data set.

The Abalone data has been taken from University of California, Irvine (UCI) of Machine Learning Repository [3]. Further analysis is applied on Abalone data for detecting an outlier data as follows:

- Calculating mean value and standard deviation for each attribute of the data set called $dataset_m$.
- Constructing two data matrix. First matrix represents a repeated mean value in each record called $mean_m$; its size equals to data set size. Second one represents repeated standard deviation value in each record called std_m ; its size equals to data set size.

- Finding outliers data by using the following equation:

$$outlier = |dataset_m - mean_m| > (3 * std_m). \tag{1}$$

The equation (1) considers the data value as an outlier data, when the absolute value that represents the difference between each attribute value of the data set and its mean value is to be greater than 3 times (3 is an assuming value, which is a threshold and it can be changed) of its standard deviation value. The role of using standard deviation is to evaluate the data set distribution from its mean value. The more distribution in data, the higher the deviation is. These outlier's data are kept to ensure that the proposed KD framework is working in case of noisy data (unbalanced data distribution). It is worth noting that the Abalone data is divided into 3133 records for training and 1044 records for testing.

The road traffic data has been generated using a traffic simulation model (called the METANET macroscopic flow model) [10]. Each record in the data set consists of:

- Traffic state represented by: traffic demands in road 1 (the numbers of vehicles that need to use the road 1), traffic demands in road 2 (the numbers of vehicles that need to use the road 2), traffic density in road 1 (the number of vehicles that are using road 1, per km), and traffic density in road 2 (the number of vehicles that are using road 2, per km). Fig. 3 shows information related to the input road traffic data set.
- The Average Total Time (ATT) required for a vehicle to cross the traffic network.

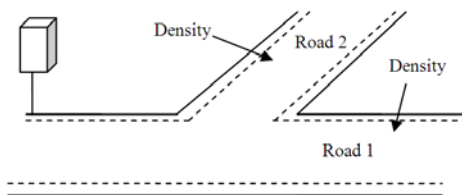


Fig. 3. Information related to the input road traffic data set

The same analysis that is applied on Abalone data is also adapted in the road traffic data for detecting an outlier data, but the noisy data found in road traffic data is less than the one existing in Abalone data. Therefore, these outlier's data are kept to ensure the proposed KD framework is working in case of noisy data (unbalanced data distribution). It is worth noting that the road traffic data is divided into 75 records for training and 25 records for testing.

The Proposed KD framework has been used to predict the Abalone ring that represents Abalone age in Abalone data and ATT in road traffic data. The results are compared with those of the integrated FCM and Apriori approach. Prediction quality is assessed using one of the statistical measures called Percentage Error (PE) of equation (2). Experimental results on the Abalone and road traffic data are summarized in Table 1.

$$PE = \left| \frac{PV - RV}{RV} \right| * 100. \quad (2)$$

where PV : the predicted output value, RV : the real output value.

Table 1. Calculation of APE

data set	Apriori	MSCAR
Abalone	29% [9]	21.9%
Road traffic	9.1% [13]	8.9%

Table 1 presents the results of using two different methodologies; first, it concerns the integrated FCM and Apriori while the second deals with FCM and MSCAR. In each methodology, the average percentage error (APE) is calculated, which is the result of the improvement of the sensitivity of future value prediction by minimizing the APE when using the integrated FCM and MSCAR. The result is compared to the result in existing work, which comes up with a better result for KD framework than the result in [9], [13]. This big difference in APE between two data sets, used in the experiment, is referred to the high number of attributes and noisy data existing in Abalone data set. However, the result is still better than the previous work as it is shown in Table 1.

4 Conclusion

The efficiency and accuracy are the important critical measures for prediction system in knowledge discovery. In this paper, a KD framework has been proposed; based on the integrated FCM and MSCAR in order to predict future value from a quantitative (continuous) data set. FCM is used to descrite the continuous data, while the MSCAR is applied for extracting Fuzzy Associative Classification Rules (FACRs). The advantages of MSCAR are summarized as follows: firstly, using vertical scanning format to scan the database one time. Secondly, adapting multiple supports instead of using single *minsupp* since the former outperforms the latter. Additionally, MSCAR utilized an associative classification approach for extracting fuzzy rules. The KD framework is applied in two data sets, Abalone and road traffic data sets. It is noted from the results that the framework has effectively minimized the average percentage error. The proposed framework also achieved a better result than the existing work. Our further work will shed light on conflicting rules in order to improve the proposed KD framework to treat a wide range of application problems.

References

1. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules in Large Databases. In: VLDB 1994: Proceedings of the 20th International Conference on Very Large Data Bases, pp. 487–499. Morgan Kaufmann Publishers Inc., San Francisco (1994)

2. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers, Norwell (1981)
3. Blake, C.L., Merz, C.J.: UCI Repository of Machine Learning Databases, <http://www.ics.uci.edu/~mlearn/MLRepository.html> (Access Date: 01/08/2009)
4. Cohen, W.W.: Fast effective rule induction. In: Proceedings of the Twelfth International Conference on Machine Learning, pp. 115–123. Morgan Kaufmann, Lake Tahoe (1995)
5. Liu, B., Hsu, W., Ma, Y.: Mining association rules with multiple minimum supports. In: KDD 1999: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 337–341. ACM Press, New York (1999)
6. Liu, B., Hsu, W., Ma, Y.: Integrating classification and association rule mining. In: KDD 1998: Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, pp. 80–86. AAAI Press, New York (1998)
7. Liu, B., Ma, Y., Wong, C.K.: Improving an association rule based classifier. In: Zighed, D.A., Komorowski, J., Żytkow, J.M. (eds.) PKDD 2000. LNCS (LNAI), vol. 1910, pp. 504–509. Springer, Heidelberg (2000)
8. Liu, Y., Jiang, Y., Liu, X., Yang, S.: CSMC: A combination strategy for multi-class classification based on multiple association rules. Knowledge-Based Systems 21(8), 786–793 (2008)
9. Lu, J., Xu, B., Jiang, J.: A prediction method of fuzzy association rules. In: IRI 2003: Proceedings of the 2003 IEEE International Conference on Information Reuse and Integration, pp. 98–103. IEEE Computer Society Press, Las Vegas (2003)
10. Messmer, A., Papageorgiou, M.: METANET: A macroscopic simulation program for motorway networks. Traffic Engineering and Control 31(8/9), 466–470 (1990)
11. Pach, F.P., Gyenesi, A., Abonyi, J.: Compact fuzzy association rule-based classifier. Expert Systems with Applications 34(4), 2406–2416 (2008)
12. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers Inc., San Francisco (1993)
13. Sowan, B., Dahal, K.P., Hossain, M.A.: Knowledge Discovery based on Integrated Fuzzy and Apriori Approach for Prediction. In: SKIMA 2009: Proceedings of the 3rd International Conference on Software, Knowledge, Information Management and Applications, Fes, Morocco, pp. 70–77 (2009)
14. Thabtah, F., Cowling, P., Hammoud, S.: Improving rule sorting, predictive accuracy and training time in associative classification. Expert Systems with Applications 31(2), 414–426 (2006)
15. Thabtah, F., Cowling, P., Peng, Y.: MCAR: multi-class classification based on association rule. In: Proceedings of the ACS/IEEE 2005 International Conference on Computer Systems and Applications, pp. 33–I. IEEE Computer Society Press, Washington (2005)
16. Zaki, M.J., Parthasarathy, S., Ogihara, M., Li, W.: New algorithms for fast discovery of association rules. In: KDD 1997: Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining, pp. 283–286. AAAI Press, Newport Beach (1997)

Learning Methods for Type-2 FLS Based on FCM

Janusz T. Starczewski^{1,2}, Łukasz Bartczuk², Piotr Dziwiński²,
and Antonino Marvuglia^{3,*}

¹ Academy of Management (SWSPiZ), Institute of Information Technology,
ul. Sienkiewicza 9, 90-113 Łódź, Poland

² Department of Computer Engineering, Czestochowa University of Technology,
Al. Armii Krajowej 36, 42-200 Czestochowa, Poland

{janusz.starczewski, lukasz.bartczuk, piotr.dziwinski}@kik.pcz.pl

³ Cork Constraint Computation Centre (4C), Western Gateway Building
University College Cork, Cork, Ireland

Abstract. This paper presents a new two-phase learning method for interval type-2 fuzzy logic systems. The method combines traditional learning approaches to type-1 fuzzy systems with fitting of interval memberships using FCM memberships. Two improving modifications of the proposed method are supplied additionally.

1 Introduction

Recently, many applications of fuzzy logic systems employ interval type-2 fuzzy sets with the ability of modelling membership uncertainty, e.g. [1,2,3,4]. Recall that an interval fuzzy set of type-2 in \mathbb{R} , is a set of pairs $\{x, \mu_{\tilde{A}}(x)\}$, denoted in the fuzzy union notation $\tilde{A} = \int_{x \in \mathbb{R}} \mu_{\tilde{A}}(x) / x$, where x is an element of the fuzzy set associated with the fuzzy membership function $\mu_{\tilde{A}} : \mathbb{R} \rightarrow \mathcal{F}_I([0, 1])$. $\mathcal{F}_I([0, 1])$ is the set of interval (fuzzy) subsets of the unit interval $[0, 1]$. Consequently, the values of $\mu_{\tilde{A}}$ are subintervals of $[0, 1]$ which may be represented by

$$\mu_{\tilde{A}}(x) = \left[\underline{\mu}_A, \overline{\mu}_A \right], \quad (1)$$

where $\underline{\mu}_A$ is referred as a lower membership function and $\overline{\mu}_A$ is referred as an upper membership function.

This paper offers a new two-phase learning method for interval type-2 fuzzy logic systems. The method combines traditional learning approaches to type-1 fuzzy systems with fitting of interval memberships using FCM memberships. Two improving modifications of the proposed method are supplied additionally.

2 Interval Type-2 FLSs

We start from the interval type-2 fuzzy logic system architecture [5] that consists of the type-2 fuzzy rule base, the type-2 fuzzifier, the inference engine modified to deal with type-2 fuzzy sets, and the defuzzifier, which is split into the

* This work was partly supported by Polish Ministry of Science and Higher Education (Habilitation Project N N516 372234 2008–2011).

type reducer and type-1 defuzzifier. The upper membership function, denoted by $\bar{\mu}_A$, is created as an upper bound of a support of secondary memberships, i.e., $\bar{\mu}_A(x) = \{\sup u : f_x(u) > 0\}$. The lower membership function is created as $\underline{\mu}_A(x) = \{\inf u : f_x(u) > 0\}$.

The first step in the proposed interval type-2 fuzzy logic system is a calculation of upper and lower degrees of compatibility between premises \tilde{A}'_n and antecedents \tilde{A}^k_n . Since the singleton fuzzification is used, the inputs are directly mapped into upper and lower membership functions. Obviously, the following constraint must be held:

$$\underline{\mu}_{A^k_n}(x_n) \leq \bar{\mu}_{A^k_n}(x_n), \forall x_n \in \mathbf{X}. \tag{2}$$

Then, the Cartesian product of premises is computed for upper and lower membership functions separately, i.e.,

$$\bar{\mu}_{A^k}(\mathbf{x}') = \prod_{n=1}^N \bar{\mu}_{A^k_n}(x'_n), \tag{3}$$

$$\underline{\mu}_{A^k}(\mathbf{x}') = \prod_{n=1}^N \underline{\mu}_{A^k_n}(x'_n). \tag{4}$$

Since we assume all consequents to be singletons, the application of the inference mechanism based on any arbitrary extended t-norm leads to equality between conclusion grades and activation grades, i.e.,

$$\bar{\tau}_k = \bar{\mu}_{B^k}(y) = \bar{\mu}_{A^k}(\mathbf{x}'), \tag{5}$$

$$\underline{\tau}_k = \underline{\mu}_{B^k}(y) = \underline{\mu}_{A^k}(\mathbf{x}'). \tag{6}$$

The intermediate outputs, maximal and minimal, are calculated by the height type defuzzification, i.e.,

$$y_{\max} = \frac{\sum_{k=1}^R y_k \underline{\tau}_k + \sum_{k=R+1}^K y_k \bar{\tau}_k}{\sum_{k=1}^R \underline{\tau}_k + \sum_{k=R+1}^K \bar{\tau}_k}, \tag{7}$$

$$y_{\min} = \frac{\sum_{k=1}^{L-1} y_k \bar{\tau}_k + \sum_{k=L}^K y_k \underline{\tau}_k}{\sum_{k=1}^{L-1} \bar{\tau}_k + \sum_{k=L}^K \underline{\tau}_k}. \tag{8}$$

where L and R are determined by the well known Karnik-Mendel type-reduction algorithm. If an interpretation of the outputs demands a final defuzzification, the overall output of the system can be calculated as the average of y_{\max} and y_{\min} .

3 A Two Phase Learning Method

In this paper, we would like to propose a new two phase general learning method dedicated to interval type-2 fuzzy logic systems.

3.1 Type-1 Learning Phase

The purpose of the first phase is to compute centers of membership functions for each fuzzy set $\mu_{A_n^k}$. In this paper, without loss of generality, we make use of Gaussian membership functions such that each $\mu_{A_n^k} = gauss(x, m_n^k, \sigma_n^k)$ has its center m_n^k and its spread σ_n^k , where

$$gauss(x, m_n^k, \sigma_n^k) = \exp\left(-\left(\frac{x - m_n^k}{\sigma_n^k}\right)^2\right). \tag{9}$$

The k -th antecedent type-1 fuzzy set is characterized by a membership function realized by the T-norm Cartesian product, i.e.

$$\mu_{A^k} = T_{n=1}^N \mu_{A_n^k}. \tag{10}$$

Tuning antecedent parameters, m_n^k and σ_n^k , as well as consequent parameters, y^k and σ^k , can be realized by one of the learning methods like gradient methods (e.g. Error Back Propagation), clustering methods or genetic algorithms.

3.2 Interval Type-2 Uncertainty Fitting Based on FCM

The purpose of the second stage is to create type-2 fuzzy membership functions, which should take into account the inner uncertainty of a modeled process. Let us assume that the number of cluster centers is equal to K . The cluster center is a vector that consists of centers of antecedents and a center of the consequent, i.e., $\mathbf{v}_k = [m_1^k, m_2^k, \dots, m_N^k, y^k]$. This vector is actually fixed by the type-1 learning phase, e.g. the BP algorithm. Correspondingly to the cluster center, a t -th pattern is represented by the extended vector $\mathbf{x}_t = [x_1(t), x_2(t), \dots, x_N(t), y(t)]$.

Therefore, we can use the fuzzy memberships defined by the FCM method

$$u_{kt} = \frac{1}{\sum_{j=1}^c \left(\frac{d_{kt}}{d_{jt}}\right)^{\frac{2}{m-1}}}, \tag{11}$$

$$d_{kt} = \|\mathbf{x}_t - \mathbf{v}_k\| > 0, \quad \forall k \text{ and } t, \tag{12}$$

$$\text{if } d_{kt} = 0 \text{ then } u_{kt} = 1 \text{ and } u_{it} = 0 \text{ for } i \neq k. \tag{13}$$

We assume that training data are corrupted by **no measurement error**. With this assumption, the difference between the FCM memberships and the membership functions achieved from the type-1 learning phase is caused by an inner uncertainty of a modeled process. Therefore, we can expand upper and lower membership functions over the training data, such that the memberships of all data points are covered by the so called footprint of uncertainty of type-2 fuzzy membership function. The upper membership function is a normal Gaussian function,

$$\begin{aligned} \bar{\mu}_{A_n^k}(x_n) &= gauss(x_n, m_n^k, \bar{\sigma}_n^k), \\ \bar{\mu}_{B^k}(y) &= gauss(y, y^k, \bar{\sigma}^k), \end{aligned}$$

being a **superior limit** of a function family drawn through points $(x_n(t), u_{kt})$, i.e.,

$$\overline{\sigma}_n^k = \max_t \sigma_t : \{gauss(x_n(t), m_n^k, \sigma_t) = u_{kt}\} \tag{14}$$

$$\overline{\sigma}_n^k = \max_t \frac{|x_n(t) - m_n^k|}{\sqrt{-\log u_{kt}}}, \tag{15}$$

or through points $(y(t), u_{kt})$, i.e.,

$$\overline{\sigma}^k = \max_t \frac{|y(t) - y^k|}{\sqrt{-\log u_{kt}}}. \tag{16}$$

The lower membership function is a **scaled** (by h) Gaussian function,

$$\underline{\mu}_{A_n^k}(x_n) = \underline{h}_n^k gauss(x_n, m_n^k, \sigma_n^k), \tag{17}$$

$$\underline{\mu}_{B^k}(y) = \underline{h}^k gauss(y, y^k, \sigma^k), \tag{18}$$

being an **inferior limit** of a function family drawn through points $(x_n(t), u_{kt})$, i.e.,

$$\underline{h}_n^k = \min_t h_t : \{h_t gauss(x_n(t), m_n^k, \sigma_n^k) = u_{kt}\} \tag{19}$$

$$\underline{h}_n^k = \min_t \frac{u_{kt}}{gauss(x_n(t), m_n^k, \sigma_n^k)}, \tag{20}$$

or in the consequent part

$$\underline{h}^k = \min_t \frac{u_{kt}}{gauss(y(t), y^k, \sigma^k)}. \tag{21}$$

Fig. 11 presents a graphical representation of the membership function obtained for the third dimension (Petal Length) of second cluster created by the FCM algorithm for the Iris Flower classification problem. By the "x" symbol, we marked the points which were used to determine the upper and lower membership function.

3.3 Modifications of Uncertainty Fitting

Contextual Type-2 Uncertainty Fitting. In Fig. 11, we can see that the lower membership function has its low magnitude. In this case, we obtain a wide interval of uncertainty, which does not give us meaningful information. This situation appears because we assumed that lower membership function should be an inferior limit of the function family drawn through points $(x_n(t), u_{kt})$, $\forall x_n(t) \in \mathbf{x}$. With this assumption, we are obligated to consider all patterns with no respect into which cluster they were assigned by FCM algorithm.

To overcome this drawback, we propose the following modification of the previous algorithm called a contextual uncertainty fitting. This algorithm computes

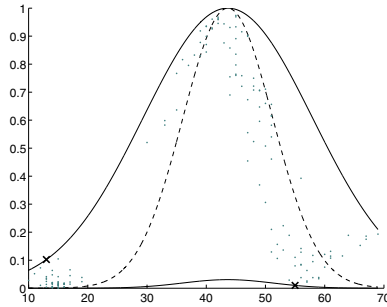


Fig. 1. Type-2 membership function obtained by the Gaussian function fitting

fuzzy memberships u_{kt} of each pattern in every cluster. We can say that pattern $x(t)$ is assigned to the κ cluster, $\kappa = 1, \dots, K$ if $\arg(\max_k u_{kt}) = \kappa$. If we assign an index κ to every pattern, we can split them into K disjoint sets:

$$\mathbf{x}^k = \{\mathbf{x}(t) : \kappa_t = k\}, \tag{22}$$

where κ_t means cluster to which pattern $\mathbf{x}(t)$ is assigned.

Now we can compute the width of an upper membership function and the scaled factor of a lower membership function for each cluster, but we take into consideration only those patterns that belong to the \mathbf{x}^k set:

$$\bar{\sigma}_n^k = \max_t \frac{|x_n^k(t) - m_n^k|}{\sqrt{-\log u_{kt}}}, \tag{23}$$

$$\underline{h}_n^k = \min_t \frac{u_{kt}}{\text{gauss}(x_n^k(t), m_n^k, \sigma_n^k)}. \tag{24}$$

The membership function obtained by this method is presented in fig. 2. As we can see, after this modification, the type-2 membership function is more intuitive than the former one.

Non-Symmetrical Gaussian Membership Function. A further improvement of the shape of a resultant membership function arises from the use of a piecewise Gaussian function as the upper membership function:

$$\mu(x) = \begin{cases} \exp\left[-\left(\frac{x-m}{\sigma_{left}^k}\right)^2\right] & \text{for } x < m, \\ \exp\left[-\left(\frac{x-m}{\sigma_{right}^k}\right)^2\right] & \text{otherwise.} \end{cases} \tag{25}$$

In this case, we compute σ_{left}^k and σ_{right}^k independently for $\{x^k(t) : x_n^k(t) \leq m_n^k\}$ and $\{x^k(t) : m_n^k < x_n^k(t)\}$. The resultant membership function is presented in fig. 3.

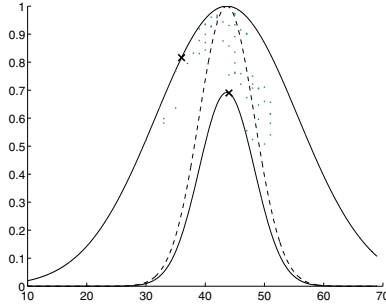


Fig. 2. Type-2 membership function obtained the contextual Gaussian function fitting

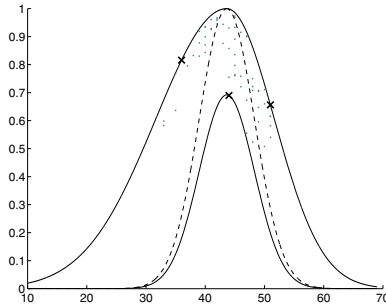


Fig. 3. Type-2 membership function obtained by the non-symmetrical Gaussian function fitting

4 Experiments

For experiments, we chose the Iris dataset from the UCI Repository of machine learning databases. The dataset is composed of 150 patterns uniformly distributed among three classes of iris species. The patterns are characterized by five attributes including the decision.

During the experiments, the whole dataset was divided into two parts, learning and testing, consisting respectively 120 and 30 randomly chosen patterns. The experiments were performed for four kinds of type-2 fuzzy logic systems. Each of them was composed of three rules describing objects from particular classes. The former two systems were created with the use of the method presented in section 3.2 for Gaussian and piecewise Gaussian upper membership functions. The latter two systems were created by applying the algorithm described in section 3.3 also for Gaussian and piecewise Gaussian upper membership functions. In the all modeled cases, lower membership functions were of the Gaussian type.

For each testing data, we determined interval firing degrees of rules, and then basing on them we assigned labels to the classes, to which the patterns fit. The assignment were performed with the use of one of the following methods: MAX-MIN, MAX, AVG, and a comparison of fuzzy intervals.

Table 1. Classification rates for Gaussian functions obtained by the method presented in section 3.2

	Type-1	MAX-MIN	MAX	AVG	Inequality
correct classifications	0.88	0.83	0.87	0.87	0.75
incorrect classifications	0.12	0.11	0.13	0.13	0.01
undecided	–	0.06	–	–	0.25

Table 2. Classification rates for piecewise Gaussian functions obtained by the method presented in section 3.2

	Type-1	MAX-MIN	MAX	AVG	Inequality
correct classifications	0.88	0.81	0.85	0.85	0.79
incorrect classifications	0.12	0.11	0.15	0.15	0.03
undecided	–	0.07	–	–	0.19

The MAX-MIN method detected rules with maximum values of upper and lower firing grades. If the values indicated the same rule, the pattern were assigned to the class corresponding to the rule. Otherwise, the classification, based on upper fired rules, was not certain.

The MAX and AVG methods of the output interpretation sought rules for the maximum value of upper firing grades and for average firing grades, respectively.

The last method was based on an algorithm comparing fuzzy intervals, proposed in [6]. This algorithm determined a degree from $[0, 1]$ of satisfying an inequality between two fuzzy quantities A_1 and A_2 . The value 1 indicated a full certainty that A_1 is greater than A_2 , 0 indicated $A_1 < A_2$, and 0.5 was reserved for the equal fuzzy quantities. We assumed that the correct classification is performed if a fuzzy interval firing grade of a rule satisfies the inequality with all remaining rules with the inequality degree greater than 0.6. Otherwise, the state of the system was set to "undecided".

Table 3. Classification rates for Gaussian functions obtained by the method presented in section 3.3

	Type-1	MAX-MIN	MAX	AVG	Inequality
correct classifications	0.90	0.86	0.88	0.89	0.75
incorrect classifications	0.10	0.08	0.12	0.11	0.02
undecided	–	0.06	–	–	0.23

Table 4. Classification rates for piecewise Gaussian functions obtained by the method presented in section 3.3

	Type-1	MAX-MIN	MAX	AVG	Inequality
correct classifications	0.90	0.89	0.89	0.89	0.83
incorrect classifications	0.10	0.09	0.11	0.11	0.03
undecided	–	0.02	–	–	0.14

The experiments are summarized in Tables 1-4. The inequality method of comparing interval fuzzy numbers has given the lowest number of incorrect classifications. Unfortunately, the greatest number of correct classifications is still a domain of the type-1 fuzzy logic.

5 Conclusion

In this paper, we have proposed a new two phase general learning method dedicated to interval type-2 fuzzy logic systems. The method was thought as a combination of traditional learning techniques dedicated to type-1 fuzzy systems, and a heuristic fitting of upper and lower membership functions according to data. The fitting relies on the use of the FCM membership partitioning. To achieve better fitting of membership functions, two modifications of the learning method have been proposed. One of them has put the fitting procedure in the context of particular classes. The second modification has used non-symmetrical upper membership functions.

References

1. Castillo, O., Aguilar, L., Cazarez-Castro, N., Boucherit, M.: Application of type-2 fuzzy logic controller to an induction motor drive with seven-level diode-clamped inverter and controlled infeed. *Electrical Engineering* 90(5), 347–359 (2008)
2. Hagrass, H.: A hierarchical type-2 fuzzy logic control architecture for autonomous robots. *IEEE Transactions on Fuzzy Systems* 12(4), 524–539 (2004)
3. Liang, Q., Mendel, J.: Interval type-2 fuzzy logic systems: Theory and design. *IEEE Transactions on Fuzzy Systems* 8, 535–550 (2000)
4. Uncu, O., Turksen, I.: Discrete interval type 2 fuzzy system models using uncertainty in learning parameters. *IEEE Transactions on Fuzzy Systems* 15(1), 90–106 (2007)
5. Karnik, N., Mendel, J., Liang, Q.: Type-2 fuzzy logic systems. *IEEE Transactions on Fuzzy Systems* 7(6), 643–658 (1999)
6. Dorohonceanu, B.: Comparing fuzzy numbers, algorithm alley. *Dr. Dobb's Journal* 343, 38–45 (2002)

On an Enhanced Method for a More Meaningful Ranking of Intuitionistic Fuzzy Alternatives

Eulalia Szmidt and Janusz Kacprzyk

Systems Research Institute, Polish Academy of Sciences,
ul. Newelska 6, 01-447 Warsaw, Poland

and

Warsaw School of Information Technology,
ul. Newelska 6, 01-447 Warsaw, Poland
{szmidt, kacprzyk}@ibspan.waw.pl

Abstract. We consider an approach for ranking alternatives represented by Atanassov's intuitionistic fuzzy sets (A-IFSs) which takes into account not only the amount of information related to an alternative (expressed here by the normalized Hamming and the normalized Euclidean distances from the ideal positive alternative) but also the reliability of information (how sure the information is) expressed here by a so-called hesitation margin.

1 Introduction

Atanassov's intuitionistic fuzzy sets, or an A-IFS, for short (cf. Atanassov [1]), are a tool to better model imperfect (imprecise) information. An important problem is how to rank the alternatives (options). A set of options (alternatives) is expressed by an extent μ to which each option fulfills a set of criteria, and an extent ν it does not fulfill it. This can be represented by an A-IFSs [cf. Section 2]. The ranking of intuitionistic fuzzy alternatives is non-trivial because there is no natural linear order among them as opposed to fuzzy sets (Zadeh [28]). For some approaches for ranking the intuitionistic fuzzy alternatives, cf. Chen and Tan [2], Hong and Choi [3], Li et al. [4], [5], and Hua-Wen Liu and Guo-Jun Wang [6] (for some pros and cons of the cited approaches cf. (Szmidt and Kacprzyk [25])).

Here we propose another approach. First, we employ the representation of A-IFSs taking into account all three functions (the membership, non-membership, and hesitation margin). Such a representation gives intuitively appealing results (cf. e.g., Szmidt and Kacprzyk [2], [15], [23], [24]). Second, we propose a ranking function which depends on two factors: the amount of information (given by a distance from the ideal positive alternative), and the reliability of information (given by the hesitation margin).

2 A Brief Introduction to Intuitionistic Fuzzy Sets

One of the possible generalizations of a fuzzy set in X (Zadeh [28]), given by $A' = \{ \langle x, \mu_{A'}(x) \rangle \mid x \in X \}$ where $\mu_{A'}(x) \in [0, 1]$ is the membership function of the fuzzy set A' , is Atanassov's intuitionistic fuzzy set (Atanassov [1]) A :

$$A = \{ \langle x, \mu_A(x), \nu_A(x) \rangle \mid x \in X \} \quad (1)$$

where: $\mu_A : X \rightarrow [0, 1]$ and $\nu_A : X \rightarrow [0, 1]$ such that $0 \leq \mu_A(x) + \nu_A(x) \leq 1$, and $\mu_A(x), \nu_A(x) \in [0, 1]$ denote the degree of membership and a degree of non-membership of $x \in A$, respectively, and the *hesitation margin* of $x \in A$ is:

$$\pi_A(x) = 1 - \mu_A(x) - \nu_A(x) \tag{2}$$

The $\pi_A(x)$ expresses a lack of knowledge of whether x belongs to A or not (Atanassov [11]); obviously, $0 \leq \pi_A(x) \leq 1$, for each $x \in X$;

The hesitation margin turns out to be important while considering the distances (Szmidt and Kacprzyk [10], [13], [21], entropy (Szmidt and Kacprzyk [15], [23]), similarity (Szmidt and Kacprzyk [24]) for the A-IFSs, etc. i.e., the measures that play a crucial role in virtually all information processing tasks. In this paper the hesitation margin is shown to be indispensable in ranking the intuitionistic fuzzy alternatives because it indicates how reliable (sure) the information represented by an alternative is.

The use of A-IFSs instead of fuzzy sets implies the introduction of another degree of freedom (non-memberships) into the set description. Such a generalization of fuzzy sets gives us an additional possibility to represent imperfect knowledge which leads to describing many real problems in a more adequate way. Applications of intuitionistic fuzzy sets to group decision making, negotiations, voting and other situations are presented in Szmidt and Kacprzyk [9], [11], [14], [16], [17], [18], [19], [22], Szmidt and Kukier [26], [27].

In our further considerations we use the normalized Hamming distance, and the normalized Euclidean distance between the A-IFSs A, B in X (Szmidt and Kacprzyk [13], [21], Szmidt and Baldwin [7], [8]):

$$l_{IFS}(A, B) = \frac{1}{2n} \sum_{i=1}^n (|\mu_A(x_i) - \mu_B(x_i)| + |\nu_A(x_i) - \nu_B(x_i)| + |\pi_A(x_i) - \pi_B(x_i)|) \tag{3}$$

$$e_{IFS}(A, B) = \left(\frac{1}{2n} \sum_{i=1}^n (\mu_A(x_i) - \mu_B(x_i))^2 + (\nu_A(x_i) - \nu_B(x_i))^2 + (\pi_A(x_i) - \pi_B(x_i))^2 \right)^{\frac{1}{2}} \tag{4}$$

For (3) and (4) we have: $0 \leq e_{IFS}(A, B) \leq 1$, and $0 \leq q_{IFS}(A, B) \leq 1$. Clearly, both distances satisfies the conditions of the metric.

A possible geometrical representations of an A-IFS is as in Fig. 1 (cf. Atanassov [11]). Although we use a 2D figure, we still adopt our approach (e.g., Szmidt and Kacprzyk [13], [21], [15], [23], [24]) with the membership, non-membership and hesitation margin. Any element in an A-IFS may be represented inside MNO . Each point belonging to MNO is described by: (μ, ν, π) . Points M and N represent crisp elements. Point $M(1, 0, 0)$ represents elements fully belonging to an A-IFS as $\mu = 1$, and may be seen as the ideal positive element. Point $N(0, 1, 0)$ represents elements fully not belonging to an A-IFS as $\nu = 1$. Point $O(0, 0, 1)$ represents elements unsure as to if they belong or not to an A-IFS as $\pi = 1$. Segment MN , with $\pi = 0$, represents elements belonging to the classic fuzzy sets ($\mu + \nu = 1$). For example, point $A(0.2, 0.8, 0)$

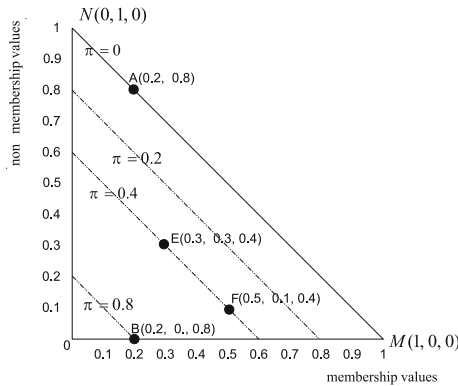


Fig. 1. Geometrical representation

(Fig. 1), like any element from segment MN represents an element of a fuzzy set. A line parallel to MN describes elements with the same hesitation margin. In Fig. 1 we can see point $E(0.3, 0.3, 0.4)$ representing an element with the hesitation margin equal 0.4, like $F(0.5, 0.1, 0.4)$. The closer a parallel line to MN is to O , the higher the hesitation margin.

3 A New Method for the Ranking of Intuitionistic Fuzzy Alternatives

Let an x be an intuitionistic fuzzy option (alternative) characterized via (μ, ν, π) , and suppose that it expresses a voting situation: μ is the proportion (from $[0, 1]$) of voters who vote for x , ν the proportion of those who vote against, and π of those who abstain. The simplest ranking of the alternatives may be to use a distance measure from the ideal voting situation $M = (x, 1, 0, 0)$ (100% voting for, 0% vote against and 0% abstain) to the alternatives considered. We will call M an *ideal positive alternative*.

However, there is a weak point in the ranking of alternatives by calculating the distances from the ideal positive alternative represented by M . For the normalized Hamming distance (3), and for a given value of the membership function (characterizing different alternatives), we obtain just the same value (distance) from the ideal positive alternative (for example, if the membership value μ is equal 0.8, for any intuitionistic fuzzy element, i.e. such that its non-membership degree ν and hesitation margin π fulfill $\nu + \pi = 0.2$, the distance is equal 0.2). It is shown in Fig. 2, a and b. To better see this, the distances (3) for any alternative from M (Fig. 2a) are presented for μ and ν for the whole range $[0, 1]$ (instead for $\mu + \nu \leq 1$ only). For the same reason (to better see the effect), in Fig. 2b) the contour plot of the distances (3) is given only for the range of μ and ν for which $\mu + \nu \leq 1$.

A similar situation occurs while we use the Euclidean distance (4) and rank the alternatives comparing their distances from the ideal positive alternative represented by M . Let $A = (x, 0.4, 0.2, 0.4)$ – 40% vote for, 20% against, and 40% abstain,

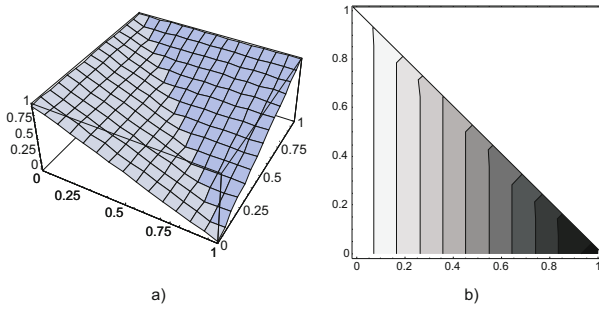


Fig. 2. a) Distances (3) of any IFS element from ideal alternative M ; b) contour plot

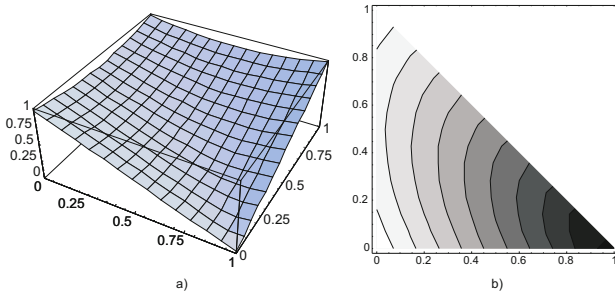


Fig. 3. a) Distances (4) of any IFS element from ideal alternative M ; b) contour plot

$B = (x, 0.4, 0.4, 0.2)$ – 40% vote for, 40% vote against and 20% abstain, The normalized Euclidean distance (4) gives:

$$e_{IFS}(M, A) = (0.5((1 - 0.4)^2 + (0 - 0.2)^2 + (0 - 0.4)^2))^{0.5} = 0.44 \quad (5)$$

$$e_{IFS}(M, B) = (0.5((1 - 0.4)^2 + (0 - 0.4)^2 + (0 - 0.2)^2))^{0.5} = 0.44 \quad (6)$$

The results seem to be counterintuitive as (4) suggests [cf. (5)–(6)] that the different alternatives (represented by) A, B seem to be “the same”. A general explanation of the above counterintuitive result follows from Fig. 3 as the results of (4) are not univocally given for a given membership value μ ; again, for clarity, the distances (4) for any x from M (Fig. 3a) are presented for μ and ν for $[0, 1]$ instead of for $\mu + \nu \leq 1$ only.

The conclusion is that the distances from the ideal positive alternative alone do not make it possible to meaningfully rank the alternatives in the intended way.

Now, analyze the essence of an alternative (an intuitionistic fuzzy alternative) using the operators of (cf. Atanassov [11]): necessity (\square), possibility (\diamond), D_α , and $F_{\alpha,\beta}$ (where $\alpha, \beta \in [0, 1]$; $\alpha + \beta \leq 1$) given as:

$$\square A = \{ \langle x, \mu_A(x), 1 - \mu_A(x) \rangle | x \in X \} \quad (7)$$

$$\diamond A = \{ \langle x, 1 - \nu_A(x), \nu_A(x) \rangle | x \in X \} \quad (8)$$

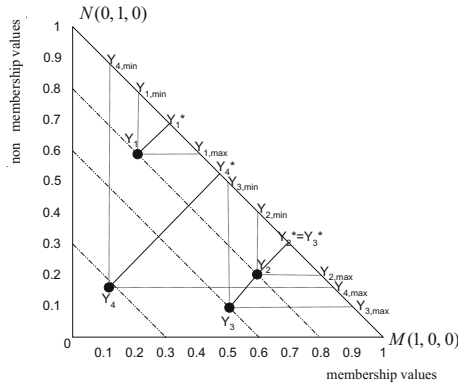


Fig. 4. Ranking alternatives Y_i

$$D_\alpha(A) = \{ \langle x, \mu_A(x) + \alpha\pi_A(x), \nu_A(x) + (1 - \alpha)\pi_A(x) \rangle \mid x \in X \} \tag{9}$$

$$F_{\alpha,\beta}(A) = \{ \langle x, \mu_A(x) + \alpha\pi_A(x), \nu_A(x) + \beta\pi_A(x) \rangle \mid x \in X \} \tag{10}$$

For example, for alternative Y_1 we obtain $\square Y_1 = Y_{1,min}$, and $\diamond Y_1 = Y_{1,max}$ (cf. Fig. 4). Operator $F_{\alpha,\beta}$ makes it possible for alternative Y_1 to become any alternative within the triangle $Y_1 Y_{1,max} Y_{1,min}$. By a similar reasoning, alternative $O(0, 0, 1)$ (because $\pi = 1$) may become any alternative (i.e. from within the whole area of MNO). Therefore, we could say that the smaller the area of the triangle $Y_i Y_{i,min} Y_{i,max}$ (Fig. 4), the better the alternative Y_i from Y . Alternatives on MN are the best in the sense that: 1) $\pi = 0$ here which means that the alternatives are fully reliable in the sense of the information represented, and 2) the alternatives are ordered – the closer an alternative to the ideal positive alternative $M(1, 0, 0)$, the better it is. This suggests that for the ranking of any intuitionistic fuzzy alternative Y_i , for a fixed π_i , we may convert them into fuzzy alternatives (naturally ordered) but still preserve knowledge of how sure this information is. For a fixed and specified π_i , each $Y_i Y_{i,min} Y_{i,max}$ is univocally given by: $Y_i^* = 0.5(Y_{i,min} + Y_{i,max})$ (Fig. 4). These Y_i^* 's are the orthogonal projections of Y_i on MN . (cf. Szmidt and Kacprzyk [12]). These orthogonal projections may be obtained via D_α (9) with α equal 0.5. In this context, a reasonable measure R that can be used for ranking the alternatives (represented by) Y_i seems to be

$$R(Y_i) = 0.5(1 + \pi_{Y_i})d_{IFS}(M, Y_i^*) \tag{11}$$

where $d_{IFS}(M, Y_i^*)$ is the normalized Hamming distance (3) or the normalized Euclidean distance (4) from the ideal positive alternative $M(1, 0, 0)$, Y_i^* is the orthogonal projection of Y_i on MN . Constant 0.5 was introduced in (11) so that $0 \leq R(Y_i) \leq 1$.

The values of function R (while using the distance (3)) for any intuitionistic fuzzy element are presented in Fig. 5a, and the counterpart contour plot – in Fig. 5b (similar results are obtained for (4)). It is easy to notice from Fig. 5 that for the alternatives for which the membership values are equal to zero, the ranking (11) gives “the same”

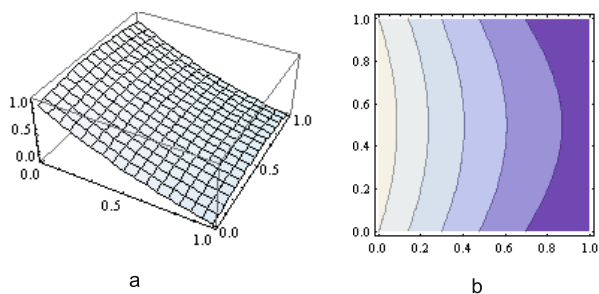


Fig. 5. a) $R(Y_i)$ as a function of a distance (3) Y_i^* from M and a hesitation margin; b) contour plot

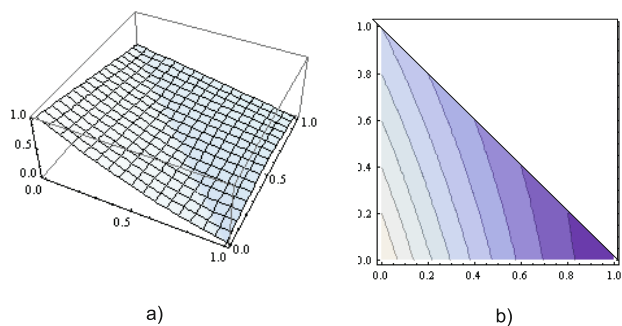


Fig. 6. a) $R_{IFS}(Y_i)$ as a function of a distance (4) from M and a hesitation margin; b) contour plot

result (white area in Fig. 5b). It is the reason that instead of (11) we use the following measure R_{IFS} :

$$R_{IFS}(Y_i) = 0.5(1 + \pi_{Y_i})d_{IFS}(M, Y_i) \tag{12}$$

where $d_{IFS}(M, Y_i)$ is the normalized Hamming distance (3) or normalized Euclidean distance (4) from the ideal positive alternative $M(1, 0, 0)$. The constant 0.5 was introduced in (12) to ensure that $0 \leq R_E(Y_i) \leq 1$. The values of R_{IFS} for any intuitionistic fuzzy element are in Fig. 6 and 7. In Fig. 6a there are results for the normalized Hamming distance (Fig. 6b – contour plot), in Fig. 7a – for the normalized Euclidean distance (Fig. 7b – contour plot).

Equation (12) reflects the “quality” of an alternative – the lower $R_{IFS}(Y_i)$, (12), the better the alternative in the sense of the amount and reliability of information.

For both (3) and (4), the best one is alternative $M(1, 0, 0)$ ($R_{IFS}(M) = 0$). For alternative $N(0, 1, 0)$ we obtain $R_{IFS}(N) = 0.5$ (N is fully reliable as the hesitation margin is equal 0 but the distance $d_{IFS}(M, N) = 1$). In general, on MN , the values of R_{IFS} decrease from 0.5 (for alternative N) to 0 (for the best alternative M). The maximal value of R_{IFS} , i.e. 1, is for $O(0, 0, 1)$ for which $d_{IFS}(M, O)$, $\pi_O = 1$ (alternative O “indicates” the whole triangle MNO). All other alternatives Y_i “indicate”

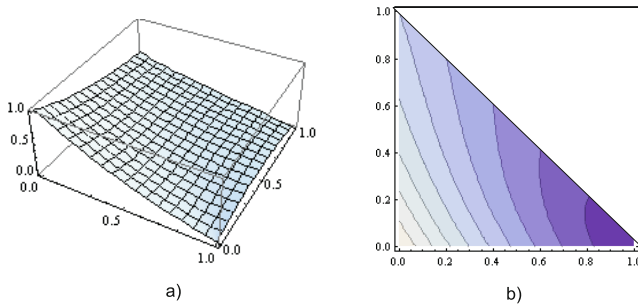


Fig. 7. a) $R_{IFS}(Y_i)$ as a function of a distance (4) from M and a hesitation margin; b) contour plot

smaller triangles $Y_i Y_{i,min} Y_{i,max}$ (Fig. 4), so that their R_{IFS} 's are smaller (better as to the amount of the reliable information).

4 Conclusions

A new method of ranking intuitionistic fuzzy alternatives was proposed that takes into account the amount and reliability of information connected with an alternative. In this way we were in a position to avoid counter-intuitive results of ranking for cases that may be described as related to the same values of the membership/non-membership functions but different values of the hesitation margin. Such situations, though they give the same (or highly similar) results of ranking by using the traditional tools involving the membership and non-membership functions only, concern clearly different cases that can be caught only by using additionally the hesitation margin.

References

1. Atanassov, K.: Intuitionistic Fuzzy Sets: Theory and Applications. Springer, Heidelberg (1999)
2. Chen, S.M., Tan, J.M.: Handling multi-criteria fuzzy decision-making problems based on vague-set theory. *Fuzzy Sets and Systems* 67(2), 163–172 (1994)
3. Hong, D.H., Choi, C.H.: Multicriteria fuzzy decision making problems based on vague set theory. *Fuzzy Sets and Systems* 114, 103–113 (2000)
4. Li, F., Lu, A., Cai, L.: Methods of multi-criteria fuzzy decision making base on vague sets. *J. of Huazhong Univ. of Science and Technology* 29(7), 1–3 (2001)
5. Li, F., Rao, Y.: Weighted methods of multi-criteria fuzzy decision making based on vague sets. *Computer Science* 28(7), 60–65 (2001)
6. Liu, H.-W., Wang, G.-J.: Multi-criteria decision making methods based on intuitionistic fuzzy sets. *EJOR* 179, 220–233 (2007)
7. Szmidt, E., Baldwin, J.: New Similarity Measure for Intuitionistic Fuzzy Set Theory and Mass Assignment Theory. *Notes on IFSs* 9(3), 60–76 (2003)
8. Szmidt, E., Baldwin, J.: Entropy for Intuitionistic Fuzzy Set Theory and Mass Assignment Theory. *Notes on IFSs* 10(3), 15–28 (2004)
9. Szmidt, E., Kacprzyk, J.: Remarks on some applications of intuitionistic fuzzy sets in decision making. *Notes on IFS* 2(3), 22–31 (1996c)

10. Szmidt, E., Kacprzyk, J.: On measuring distances between intuitionistic fuzzy sets. *Notes on IFS* 3(4), 1–13 (1997)
11. Szmidt, E., Kacprzyk, J.: Group Decision Making under Intuitionistic Fuzzy Preference Relations. In: *IPMU 1998*, pp. 172–178 (1998)
12. Szmidt, E., Kacprzyk, J.: A Fuzzy Set Corresponding to an Intuitionistic Fuzzy Set. *IJUFKIS* 6(5), 427–435 (1998)
13. Szmidt, E., Kacprzyk, J.: Distances between intuitionistic fuzzy sets. *Fuzzy Sets and Systems* 114(3), 505–518 (2000)
14. Szmidt, E., Kacprzyk, J.: On Measures on Consensus Under Intuitionistic Fuzzy Relations. In: *IPMU 2000*, pp. 1454–1461 (2000)
15. Szmidt, E., Kacprzyk, J.: Entropy for intuitionistic fuzzy sets. *Fuzzy Sets and Systems* 118(3), 467–477 (2001)
16. Szmidt, E., Kacprzyk, J.: Analysis of Consensus under Intuitionistic Fuzzy Preferences. In: *Proc. Int. Conf. in Fuzzy Logic and Technolog*, De Montfort Univ. Leicester, UK, pp. 79–82 (2001)
17. Szmidt, E., Kacprzyk, J.: Analysis of Agreement in a Group of Experts via Distances Between Intuitionistic Fuzzy Preferences. In: *Proc. 9th Int. Conf. IPMU 2002*, pp. 1859–1865 (2002a)
18. Szmidt, E., Kacprzyk, J.: An Intuitionistic Fuzzy Set Based Approach to Intelligent Data Analysis (an application to medical diagnosis). In: Abraham, A., Jain, L., Kacprzyk, J. (eds.) *Recent Advances in Intelligent Paradigms and Applications*, pp. 57–70. Springer, Heidelberg (2002b)
19. Szmidt, E., Kacprzyk, J.: An Intuitionistic Fuzzy Set Based Approach to Intelligent Data Analysis (an application to medical diagnosis). In: Abraham, A., Jain, L., Kacprzyk, J. (eds.) *Recent Advances in Intelligent Paradigms and Applications*, pp. 57–70. Springer, Heidelberg (2002c)
20. Szmidt, E., Kacprzyk, J.: A New Concept of a Similarity Measure for Intuitionistic Fuzzy Sets and its Use in Group Decision Making. In: Torra, V., Narukawa, Y., Miyamoto, S. (eds.) *MDAI 2005. LNCS (LNAI)*, vol. 3558, pp. 272–282. Springer, Heidelberg (2005)
21. Szmidt, E., Kacprzyk, J.: Distances Between Intuitionistic Fuzzy Sets: Straightforward Approaches may not work. In: *IEEE IS 2006*, pp. 716–721 (2006)
22. Szmidt, E., Kacprzyk, J.: An Application of Intuitionistic Fuzzy Set Similarity Measures to a Multi-criteria Decision Making Problem. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) *ICAISC 2006. LNCS (LNAI)*, vol. 4029, pp. 314–323. Springer, Heidelberg (2006)
23. Szmidt, E., Kacprzyk, J.: Some problems with entropy measures for the Atanassov intuitionistic fuzzy sets. In: Masulli, F., Mitra, S., Pasi, G. (eds.) *WILF 2007. LNCS (LNAI)*, vol. 4578, pp. 291–297. Springer, Heidelberg (2007)
24. Szmidt, E., Kacprzyk, J.: A New Similarity Measure for Intuitionistic Fuzzy Sets: Straightforward Approaches may not work. In: *2007 IEEE Conf. on Fuzzy Systems*, pp. 481–486 (2007a)
25. Szmidt, E., Kacprzyk, J.: Amount of information and its reliability in the ranking of Atanassov's intuitionistic fuzzy alternatives. In: Rakus-Andersson, E., Yager, R., Ichalkaranje, N., Jain, L.C. (eds.) *Recent Advances in decision Making, SCI 222*, pp. 7–19. Springer, Heidelberg (2009)
26. Szmidt, E., Kukier, M.: Classification of Imbalanced and Overlapping Classes using Intuitionistic Fuzzy Sets. In: *IEEE IS 2006, London*, pp. 722–727 (2006)
27. Szmidt, E., Kukier, M.: A New Approach to Classification of Imbalanced Classes via Atanassov's Intuitionistic Fuzzy Sets. In: Wang, H.-F., (ed.) *Intelligent Data Analysis: Developing New Methodologies Through Pattern Discovery and Recovery*, Idea Group (2008)
28. Zadeh, L.A.: Fuzzy sets. *Information and Control* 8, 338–353 (1965)

I-Fuzzy Partitions for Representing Clustering Uncertainties

Vicenç Torra¹ and Ji-Hee Min²

¹ IIIA, Institut d'Investigació en Intel·ligència Artificial,
CSIC, Spanish Council for Scientific Research,
Campus de Bellaterra, 08193 Bellaterra, Catalonia, Spain
vtorra@iia.csic.es

² School of Electrical Engineering and Computer Science,
Hanyang University, Korea
jhmin@fuzzy.hanyang.ac.kr

Abstract. In a recent paper we introduced intuitionistic fuzzy partitions or interval-valued fuzzy partitions as a way to represent the uncertainty of fuzzy clustering. In this paper we reconsider these definitions so that these fuzzy partitions can be used to represent other uncertainties on the clustering processes.

Keywords: Intuitionistic fuzzy sets, intuitionistic fuzzy partitions, fuzzy clustering, fuzzy partitions, fuzzy c -means.

1 Introduction

Clustering algorithms [4,5] have been considered for a long time. Different methods exist that use different representation formalisms to visualize the structure found in the data. Among the existing formalisms, partitions (either crisp or fuzzy partitions) and dendrograms are some of the most used ones.

Hard c -means is the most representative method for partitioning data into crisp classes (i.e. to define a crisp partition). In contrast, fuzzy c -means [3] is, probably, the most representative one for partitioning data into fuzzy partitions.

The introduction of fuzzy clustering methods [5,9] required the generalization of crisp partitions, as elements can belong at the same time to different clusters.

In a recent paper [12] we proposed a definition for intuitionistic fuzzy partitions. The goal was to introduce intuitionistic fuzzy partitions and show how they can be built from data. From a practical perspective, the goal was to represent the uncertainty that we found when different clustering algorithms are applied to the same data set due to local optima. Note that due to the effect of local optima, different executions of the same fuzzy clustering algorithm with the same parameters can lead to different fuzzy partitions. We showed in [12] that intuitionistic fuzzy partitions can be used to model this situation, solving some of the difficulties faced in [7,10] are solved.

However, the definition in [12] and in [11] only permits us to represent uncertainty based on different executions of the same algorithm using the same

parameters (e.g., m and c for, say, fuzzy c -means). Some recent papers [6,8] consider uncertainty caused by other factors. [8] considers the fact that the execution of the method (formally, the PCM fuzzy clustering approach) requires fixing parameter m and we can have uncertainty on the appropriate parameter m . [6] follows the same approach for the FCM algorithm.

In this paper, we reconsider the definition of intuitionistic fuzzy partition given in [12], so that it accommodates this other type of uncertainty.

The structure of the paper is as follows. In Section 2 we review some concepts that are needed later on in this paper. In Section 3 we reconsider the definition of I-fuzzy partitions, we present some of their properties and show how to construct these partitions from different executions of the FCM algorithm. The paper finishes with some conclusions.

2 Preliminaries

In this section we focus on a few previous results that are needed in the paper. We review intuitionistic fuzzy sets, and also a few topics on fuzzy clustering.

2.1 Interval Type-2 Fuzzy Sets and Intuitionistic Fuzzy Sets

Intuitionistic fuzzy sets, by Atanassov, or interval type-2 fuzzy sets, IFS for conciseness, are defined below. They are defined in terms of two functions.

Definition 1. [12] *An IFS (intuitionistic fuzzy set by Atanassov or interval type-2 fuzzy set) A in X is defined by*

$$A := \{ \langle x, \mu_A(x), \gamma_A(x) \rangle \mid x \in X \}$$

where $\mu_A : X \rightarrow [0, 1]$ and $\gamma_A : X \rightarrow [0, 1]$ with

$$0 \leq \mu_A(x) + \gamma_A(x) \leq 1.$$

In this definition, $\mu_A(x)$ represents the degree of membership of x to the subset $A \subset X$, and $\gamma_A(x)$ is the degree of non-membership to A . In fuzzy sets, we have that the degree of non-membership is $1 - \mu$, thus $\gamma_A = 1 - \mu_A$. When this equality holds for an IFS, it reduces to a fuzzy set. Interval type-2 fuzzy sets corresponds to fuzzy sets where the membership function is an interval. They are mathematically equivalent to IFS and correspond to the case of a membership equal to the interval $[\mu_A(x), 1 - \gamma_A(x)]$.

Definition 2. *For each IFS $A := \{ \langle x, \mu_A(x), \gamma_A(x) \rangle \mid x \in X \}$, we define the intuitionistic fuzzy index for $x \in X$ by*

$$\nu_A(x) := 1 - \mu_A(x) - \gamma_A(x).$$

Here, ν corresponds to the uncertainty between the membership and the non-membership to the subset A . Note that when $\nu(x) = 0$, we have no uncertainty on the membership of x . Using ν , interval type-2 fuzzy sets correspond to $[\mu_A(x), \mu_A(x) + \nu_A(x)]$. [6,8] use the convention of $[\underline{\mu}, \bar{\mu}]$. Thus, here, $\underline{\mu} = \mu$ and $\bar{\mu} = \mu + \nu$. From now on, we will use the notation $A = \langle \mu, \nu \rangle$, and when appropriate, we will also use $\bar{\mu}$ as equivalent to $\mu + \nu$.

2.2 Fuzzy Clustering

In this section we review the basics of fuzzy clustering for the particular case of using fuzzy c -means.

Definition 3. *Let X be a reference set. Then, a set of membership functions $\mathcal{M} = \{\mu_1, \dots, \mu_c\}$ is a fuzzy partition of X if for all $x \in X$ it holds*

$$\sum_{i=1}^c \mu_i(x) = 1$$

Fuzzy clustering methods construct a fuzzy partition of a given data set. There exist several fuzzy clustering methods. Fuzzy c -means is one of the most well known methods. This method, first proposed in [3], is described in most books on fuzzy sets and fuzzy clustering. See, e.g., [5,9]. We will give a brief description below. To do so, we consider the following notation. We have a set of objects $X = \{x_1, \dots, x_n\}$ and we want to build c clusters from this data. Such parameter c is expected to be given by the user. Then, the method builds the clusters which are represented by membership functions μ_{ik} , where μ_{ik} is the membership of the k th object (x_k) to the i th cluster.

Fuzzy c -means uses c (the number of clusters) and also a value m as its parameters. The parameter m , that should be such that $m > 1$, plays a central role. The larger the value m , the larger the fuzziness in the clusters. In contrast, with values near to 1, solutions tend to be crisp.

The fuzzy c -means clustering algorithm constructs the fuzzy partition μ from X solving a minimization problem. The problem is formulated below. In the formulation, we use v_i to represent the cluster center, or centroid, of the i -th cluster.

$$\text{Minimize } J_{FCM}(\mu, V) = \left\{ \sum_{i=1}^c \sum_{k=1}^n (\mu_{ik})^m \|x_k - v_i\|^2 \right\} \tag{1}$$

subject to the constraints $\mu_{ik} \in [0, 1]$ and $\sum_{i=1}^c \mu_{ik} = 1$ for all k .

A (local) optimal solution of this problem is obtained using the iterative process described in Algorithm 1. This process interleaves two steps. One that estimates the optimal membership functions of elements to clusters (when centroids are fixed) and another that estimates the centroids for each cluster (when membership functions are fixed).

2.3 I-Fuzzy Partitions

Definition 3, which corresponds to the definition of a fuzzy partition, is based on fuzzy sets. We now review the concept of I-fuzzy partition, as proposed in [12,11].

The concept of I-fuzzy partition generalizes the one of standard fuzzy partitions losing the condition that the addition of all memberships are equal to one. If we consider a set of IFS, it is not possible that all μ add to one and at the same time that all $\mu + \nu$ also add to one (unless $\nu = 0$).

The generalization requires that only the μ add to one, and that $\mu + \nu$ do not have a complete overlap. This is formalized in the next definition.

Algorithm 1. Fuzzy c -means

Step 1: Generate initial V

Step 2: Solve $\min_{\mu \in M} J(\mu, V)$ computing:

$$\mu_{ik} = \left(\sum_{j=1}^c \left(\frac{\|x_k - v_i\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m-1}} \right)^{-1}$$

Step 3: Solve $\min_V J(\mu, V)$ computing:

$$v_i = \frac{\sum_{k=1}^n (\mu_{ik})^m x_k}{\sum_{k=1}^n (\mu_{ik})^m}$$

Step 4: If the solution does not converge, go to step 2; otherwise, stop

Definition 4. Let X be a reference set. Then, a set of IFS $A = \{A_1, \dots, A_m\}$, where $A_i = \langle \mu_i, \nu_i \rangle$, is an intuitionistic fuzzy partition or interval type-2 fuzzy partition (I-fuzzy partition, for short) if

- (1) $\sum_{i=1}^m \mu_i(x) = 1$ for all $x \in X$,
- (2) for all $x \in X$, $|\{i | \mu_i(x) + \nu_i(x) = 1\}| \leq 1$
 (there is at most one IFS such that $\mu_i(x) + \nu_i(x) = 1$ for all x)

It is straightforward to see that this definition generalizes fuzzy partitions because an I-fuzzy partition with $\nu_i(x) = 0$ for all i and x is a fuzzy partition. This is stated in the next proposition.

Proposition 1. I-fuzzy partitions generalize fuzzy partitions.

3 On a New Definition for I-Fuzzy Partitions

The first condition in Definition 4 requires μ_i to add to one. This condition is not satisfied if intuitionistic fuzzy partitions are constructed according to the approach described in [6,8]. We describe this situation for the case of fuzzy c -means.

Let us consider a definition of an interval-valued membership function according to [6,8] when we have uncertainty on the m but the cluster centers v_i are already known. In fact, following [8], the resulting fuzzy clustering for two different m_1 and m_2 leads to one single set of cluster centers. Then, let us consider the case of two possible values for m , say m_1 and m_2 . Then, we can represent the membership of x_k to the clusters in terms of the following functions μ_{ik} and ν_{ik} :

$$\mu_{ik} = \min \left(\left(\sum_{j=1}^c \left(\frac{\|x_k - v_i\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_1-1}} \right)^{-1}, \left(\sum_{j=1}^c \left(\frac{\|x_k - v_i\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_2-1}} \right)^{-1} \right)$$

$$\bar{\mu}_{ik} = \max \left(\left(\sum_{j=1}^c \left(\frac{\|x_k - v_j\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_1-1}} \right)^{-1}, \left(\sum_{j=1}^c \left(\frac{\|x_k - v_j\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_2-1}} \right)^{-1} \right)$$

Thus, $\nu_{ik} = \bar{\mu}_{ik} - \mu_{ik}$. It is clear that $[\mu, \bar{\mu}]$ define an IFS, nevertheless, this definition does not satisfy the requirements of fuzzy partitions as expressed in Definition 4. This is so because the μ_{ik} do not add to one for all x_k .

This is illustrated in the next example. For the sake of simplicity, data in the examples are on \mathbb{R} instead of being in a multidimensional space.

Example 1. Let us consider a data set X on \mathbb{R} , clustered using fuzzy c -means with parameters $m_1 = 1.5$, $m_2 = 2$ and $c = 2$. Let us consider the two resulting cluster centers represented by $v_1 = 2$ and $v_2 = 4$. Then, for $x = 2.5$, we have

$$\begin{aligned} \mu_1(x) + \mu_2(x) &= \\ &= \min \left(\left(\left(\frac{0.25}{\|2.5-2\|^2} \right)^2 + \left(\frac{0.25}{\|2.5-4\|^2} \right)^2 \right)^{-1}, \left(\left(\frac{0.25}{\|2.5-2\|^2} \right) + \left(\frac{0.25}{\|2.5-4\|^2} \right) \right)^{-1} \right) + \\ &= \min \left(\left(\left(\frac{2.25}{\|2.5-2\|^2} \right)^2 + \left(\frac{2.25}{\|2.5-4\|^2} \right)^2 \right)^{-1}, \left(\left(\frac{2.25}{\|2.5-2\|^2} \right) + \left(\frac{2.25}{\|2.5-4\|^2} \right) \right)^{-1} \right) = \\ &= 0.9 + 0.0121951215 = 0.9121951215 < 1 \end{aligned}$$

To permit I-fuzzy partitions to represent this situation, we reconsider below their definition. An alternative definition follows.

Definition 5. Let X be a reference set. Then, a set of IFS $A = \{A_1, \dots, A_m\}$, where $A_i = \langle \mu_i, \nu_i \rangle$, is an intuitionistic fuzzy partition or an interval type-2 fuzzy partition (I-fuzzy partition, for short) if

- (1) $\sum_{i=1}^m \mu_i(x) \leq 1$ for all $x \in X$,
- (2) $\sum_{i=1}^m (\mu_i(x) + \nu_i(x)) \geq 1$ for all $x \in X$,
- (3) for all $x \in X$, $|\{i | \mu_i(x) + \nu_i(x) = 1\}| \leq 1$
(there is at most one IFS such that $\mu_i(x) + \nu_i(x) = 1$ for all x)

This new definition is more general than the previous one in the sense that all I-fuzzy partitions that satisfy Definition 4 also satisfy Definition 5. Note that this is so because $\sum_{i=1}^m \mu_i(x) = 1$ in Definition 4 implies, of course, $\sum_{i=1}^m \mu_i(x) \geq 1$ and, as $\mu_i + \nu_i \geq \mu_i$, we have $\sum_{i=1}^m (\mu_i(x) + \nu_i(x)) \geq \sum_{i=1}^m \mu_i(x) = 1$.

Proposition 2. All I-fuzzy partitions satisfying Definition 4 also satisfy Definition 5.

We can now show that this alternative definition is suitable when we have uncertainty on the value m .

Proposition 3. Given real values m_1 and m_2 , cluster centers v_i , and data elements X , we have that the following definitions for μ_{ik} and $\bar{\mu}_{ik}$ define an I-fuzzy partition.

- (1) $\mu_{ik} = \min \left(\left(\sum_{j=1}^c \left(\frac{\|x_k - v_j\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_1-1}} \right)^{-1}, \left(\sum_{j=1}^c \left(\frac{\|x_k - v_j\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_2-1}} \right)^{-1} \right)$
- (2) $\bar{\mu}_{ik} = \max \left(\left(\sum_{j=1}^c \left(\frac{\|x_k - v_j\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_1-1}} \right)^{-1}, \left(\sum_{j=1}^c \left(\frac{\|x_k - v_j\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_2-1}} \right)^{-1} \right)$

Proof. Let $\mu_i^m(x_k) = \left(\sum_{j=1}^c \left(\frac{\|x_k - v_i\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m-1}} \right)^{-1}$. Then, we have $\sum_i \mu_i^{m_j}(x) = 1$ for all parameters m_j . So, if we define $\mu_i(x) = \min_j \mu_i^{m_j}(x)$, we have $\sum_i \mu_i(x) = \sum_i \min_j \mu_i^{m_j}(x) \leq 1$. Similarly, if we define $\bar{\mu}_i(x) = \max_j \mu_i^{m_j}(x)$, we have $\sum_i \bar{\mu}_i(x) = \sum_i \max_j \mu_i^{m_j}(x) \geq 1$.

This proposition can be generalized easily when we consider a set of m_l , instead of only two, when constructing μ_{ik} and ν_{ik} . I.e., $M = \{m_1, \dots, m_n\}$ with μ_{ik} and $\bar{\mu}_{ik}$ defined as follows:

- (1) $\mu_{ik} = \min_{m_l \in M} \left(\left(\sum_{j=1}^c \left(\frac{\|x_k - v_i\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_l-1}} \right)^{-1} \right)$
- (2) $\bar{\mu}_{ik} = \max_{m_l \in M} \left(\left(\sum_{j=1}^c \left(\frac{\|x_k - v_i\|^2}{\|x_k - v_j\|^2} \right)^{\frac{1}{m_l-1}} \right)^{-1} \right)$

In [12,11] we described how to construct I-fuzzy partitions from the results of different applications of the same clustering algorithm with the same parameter. Such definitions are also valid for the new definition we have proposed in this paper. We review below this approach, and establish the proposition that shows that the approach is also valid here.

Definition 6. *Let us consider the application of r fuzzy c -means with parameters c and m to a data set X . This results into r fuzzy partitions. Let $\mu^{i,j}$ represent the membership function obtained by the i th clustering algorithm for the j th cluster. Let $c^{i,j}$ represent the centroid obtained by the i th clustering algorithm for the j th cluster. Let OF_i be the value of the objective function obtained by the i th clustering algorithm.*

Let p be an integer such that $1 \leq p \leq r$. Then, we define the I-fuzzy partition inferred from μ and c as the one obtained from the application of the next steps.

- (1) *Define the centroids $cn_j := c^{r^*,j}$ and the membership functions $\mu_j := \mu^{r^*,j}$ for all $j = 1, \dots, c$ where $r^* = \arg \min_{i=1}^r OF_i$.*
- (2) *Define an alternative centroid cn_j for each cluster j as the average of the p centroids (for that cluster) with minimal OF . Formally, $cn_j := (1/p) \sum_{i=1}^p c^{s(i),j}$ where s is a permutation of $\{1, \dots, r\}$ such that $OF_{s(1)} \leq \dots \leq OF_{s(r)}$.*
- (3) *Define a radius rad_j for each cluster j as the maximum distance between cn_j and $c^{s(i),j}$ for all $i \leq p$. Formally, $rad_j = \max_{i \leq p} \|c^{s(i),j} - cn_j\|$. That is, rad_j is the minimal radius that permits to encompass all p cluster centers when we build a ball with center $c^{s(i),j}$.*
- (4) *Check whether the sets $\{x \mid \|x - cn_j\| \leq rad_j\}$ overlap. If they overlap, reduce p and go to step (2).*
- (5) *Define $b_j := \{x \mid \|x - cn_j\| \leq rad_j\}$ for all clusters $j = \{1, \dots, c\}$, and define $d(x, y) = \|x - y\|$ and $d(x, b_j) = \min_{y \in b_j} \|x - y\|$. It is easy to see that $d(x, b_j) = \|x - cn_j\| - rad_j$. Then,*
 $\bar{\mu}_j(x) = 1$ for all $x \in b_j$, and,
 $\bar{\mu}_j(x) = \left(1 + \sum_{t \neq j} \left(\frac{d(x, b_t)^2}{d(x, c^{r^*,t})^2} \right)^{\frac{1}{m-1}} \right)^{-1}$ for all x not in b_j .
- (6) *Define $\nu_j(x) = \bar{\mu}_j(x) - \mu_j(x)$.*

Proposition 4. [11] When the set of optimal centroids do not coincide (i.e., when $j_1 \neq j_2$ implies that $c^{r^*,j_1} \neq c^{r^*,j_2}$), the construction in Definition 6 finishes and builds a I-fuzzy partition according to Definition 4.

Proposition 5. When the set of optimal centroids do not coincide (i.e., when $j_1 \neq j_2$ implies that $c^{r^*,j_1} \neq c^{r^*,j_2}$), the construction in Definition 6 finishes and builds a I-fuzzy partition according to Definition 5.

Proof. As according to [11], this definition satisfies the requirements of Definition 4, and as, according to Proposition 2, this implies that the requirements of Definition 5 are also fulfilled, the proposition is proven.

Example 2. Let us consider the output of four executions of fuzzy c -means on data on the real line using $c = 2$. Let us consider the construction of the I-fuzzy partition taking into account the best 3 solutions (i.e., $p = 3$).

As we have four executions, we will obtain four cluster centers. Let these cluster centers be denoted by (a_1, a_2) , (b_1, b_2) , (c_1, c_2) , and (d_1, d_2) . Where, $(a_1, a_2) = (3, 12)$, $(b_1, b_2) = (4, 13)$, $(c_1, c_2) = (5, 17)$, and $(d_1, d_2) = (1, 18)$.

Let the objective functions of these four sets of clusters be such that $OF_b \leq OF_a \leq OF_c \leq OF_d$ where OF_a is the objective function associated to cluster centers (a_1, a_2) , OF_b the one of (b_1, b_2) , and so on.

Then (Step (1)), the centroids to be selected are (b_1, b_2) , as these are the ones with a minimal objective function. Thus, $cm_1 = b_1$ and $cm_2 = b_2$. In addition, we define μ_j in terms of the memberships of b_1 and b_2 .

In Step (2), we define cn_j as the average centroid of the $p = 3$ best centroids. That is, we define cn_1 as the average of $a_1, b_1,$ and c_1 , and cn_2 as the average of a_2, b_2 and c_2 . d_1 and d_2 are not taken into consideration because the best $p = 3$ objective functions are OF_b, OF_a and OF_c . So, let $cn_1 = (a_1 + b_1 + c_1)/3 = 4$ and $cn_2 = (a_2 + b_2 + c_2)/3 = 14$.

In Step (3), we define the radius of each cluster as the maximum distance between the cluster center and the centroids to be included in the cluster. In our case, this means that the cluster on cn_1 should include centroids $a_1, b_1,$ and c_1 ; and that cluster cn_2 should include centroids a_2, b_2 and c_2 . See Figure 1.

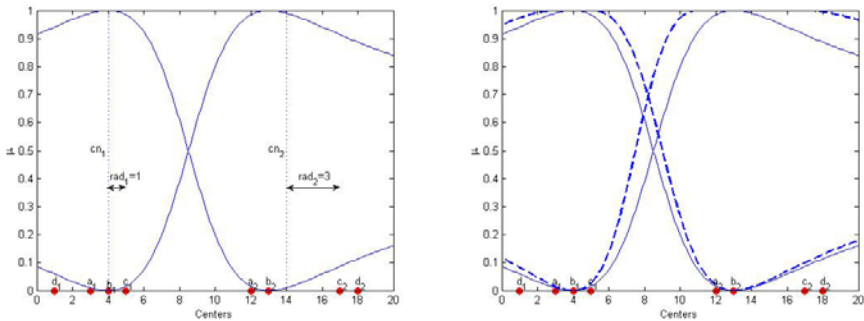


Fig. 1. Steps (3) and (6) for Example 2 when $m = 2$

Step (4) checks whether the sets overlap (Step (5) names these sets as b_j). Here we have no overlapping. Note that in our example, the sets are as following: $b_1 = [3, 5]$, $b_2 = [11, 17]$. These sets encompass the best three centroids.

Step (5) defines the membership functions $\bar{\mu}$ in terms of the centroids cn and sets b . The differences between $\bar{\mu}$ and μ define the interval. See Figure [11](#)

4 Conclusions

In this paper we have revised the definition of I-fuzzy partitions on the light of the application of fuzzy c -means to real data, to represent two types of uncertainties. The new definition is appropriate to deal with two types of uncertainties: uncertainty on the parameter m and uncertainty on the fact that fuzzy c -means can lead to local optima.

Acknowledgements

Partial support by the Spanish MEC (ARES - CONSOLIDER INGENIO 2010 CSD2007-00004 - and eAEGIS - TSI2007-65406-C03-02) is acknowledged.

References

1. Atanassov, K.T.: Intuitionistic fuzzy sets. VII ITKR's Session, Sofia (1983) (in Bulgarian)
2. Atanassov, K.T.: Intuitionistic fuzzy sets. *Fuzzy Sets and Systems* 20, 87–96 (1986)
3. Bezdek, J.C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York (1981)
4. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning*. Springer, Berlin (2001)
5. Höppner, F., Klawonn, F., Kruse, R., Runkler, T.: *Fuzzy cluster analysis*. Wiley, Chichester (1999)
6. Hwang, C., Rhee, F.C.-H.: Uncertain Fuzzy Clustering: Interval Type-2 Fuzzy Approach to C-Means. *IEEE Trans. on Fuzzy Systems* 15(1), 107–120 (2007)
7. Ladra, S., Torra, V.: On the comparison of generic information loss measures and cluster-specific ones. *Intl. J. of Unc., Fuzz. and Know. Based Syst.* 16(1), 107–120 (2008)
8. Min, J.-H., Shim, E.-A., Rhee, F.C.-H.: An Interval Type-2 Fuzzy PCM Algorithm for Pattern Recognition. In: *Proc. of FUZZ-IEEE 2009, Korea*, pp. 480–483 (2009)
9. Miyamoto, S., Ichihashi, H., Honda, K.: *Algorithms for fuzzy clustering*. Springer, Heidelberg (2008)
10. Torra, V., Endo, Y., Miyamoto, S.: On the comparison of some fuzzy clustering methods for privacy preserving data mining: towards the development of specific information loss measures. *Kybernetika* 45(3), 548–560 (2009)
11. Torra, V., Miyamoto, S.: *On Intuitionistic Fuzzy Partitions* (2009) (manuscript)
12. Torra, V., Miyamoto, S., Endo, Y., Domingo-Ferrer, J.: On intuitionistic fuzzy clustering for its application to privacy. In: *Proc. FUZZ-IEEE/WCCI 2008, Hong-Kong, China*, pp. 1042–1048 (2008)

A Quantitative Approach to Topology for Fuzzy Regions

Jörg Verstraete

Institut Badań Systemowych, Polska Akademia Nauk (Systems Research Institute,
Polish Academy of Sciences); Ul. Newelska 6, 01-447 Warszawa, Poland
DDCM, Dept. Telecommunications and Information Processing, Ghent University;
Sint Pietersnieuwstraat 41, 9000 Ghent, Belgium
jorg.verstraete@{ibspan.waw.pl, telin.ugent.be}
<http://www.ibspan.waw.pl>, <http://telin.ugent.be/ddcm>

Abstract. There has been lots of research in the field of fuzzy spatial data and the topology of fuzzy spatial objects. In this contribution, an extension to the 9-intersection model is presented, to allow for the relative position of overlapping fuzzy regions to be determined. The topology will be determined by means of a new intersection matrix, and a set of numbers, expressing the similarity between the topology of the given regions and a number of predefined cases. The approach is not merely a conceptual idea, but has been built on our representation model and can as such be immediately applied.

1 Preliminaries

1.1 Introduction

A common problem in spatial reasoning, is describing the position of one object or feature in relative to another object or feature. "Do both overlap, is one contained within the other, or do they touch?" are some examples. For crisp regions, it is fairly easy to see that the different possibilities are mutually exclusive: if two regions *touch*, then one does not *contain* the other. The concept of a broad or undetermined boundary ([1], [2]), in which the boundary was considered to be a region delimited by an inner and an outer boundary, rather than a thin line was a first extension. The topology of such regions is similar in approach to crisp topology; all intersection cases are mutually exclusive. Allowing for truly fuzzy regions however, implies that there is no certainty or precision regarding the points of to the regions. As such, statements about topology are prone to similar uncertainty and imprecision, resulting in the fact that two regions can resemble multiple intersection cases at once (regions can for instance touch and overlap). It is important to first find the cases that match, and then to generate quantitative measures to indicate how well each matches. In this paper, we will first describe the fuzzy topology model, list some of cases, and illustrate by means of an example.

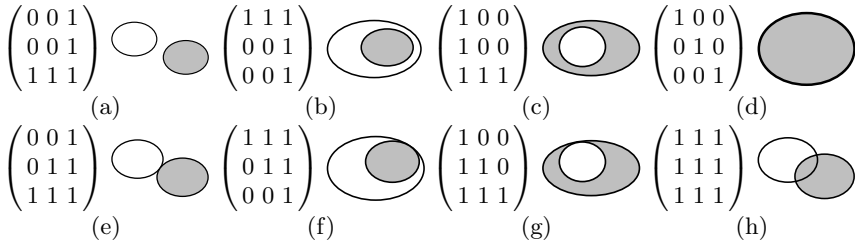


Fig. 1. Topological relations for crisp regions: disjoint (a), contains (b), inside (c), equal (d), meet (e), covers (f), coveredBy (g) and overlap (h), with their intersection matrices

1.2 9-Intersection Model

One approach to model crisp topology, is the 9-intersection model [5]. It uses the concepts of interior (points inside the region, denoted \cdot°), exterior (points the region, denoted \cdot^-) and boundary (denoted $\partial\cdot$), then considers every possible intersection between them. This yields a total of 9 possible intersections, commonly grouped in the matrix shown below:

$$\begin{pmatrix} A^\circ \cap B^\circ & A^\circ \cap \partial B & A^\circ \cap B^- \\ \partial A \cap B^\circ & \partial A \cap \partial B & \partial A \cap B^- \\ A^- \cap B^\circ & A^- \cap \partial B & A^- \cap B^- \end{pmatrix} \tag{1}$$

By assigning each matrix element 0 if the intersection is empty, and 1 if the intersection is not empty, $2^9 = 512$ matrices are possible. Depending on imposed restrictions (e.g. presence of holes), only a subset of the 512 relations is possible. For crisp regions without holes and no disconnected parts in a two-dimensional space \mathbb{R}^2 , only eight intersection matrices are meaningful, yielding the relations: disjoint, contains, inside, equal, meet, covers, coveredBy and overlap; illustrated on fig. 1. An alternative way of describing topology is the RCC calculus, but the nine-intersection model lends itself easier toward qualitative approach.

2 Fuzzy Region Model

2.1 Concept

Regions are often represented by means of an outline, represented by a curve. The region is defined as all the points located inside this curve [1]. For our definition of fuzzy regions, a different point of view is necessary: a region is considered mathematically as a set of locations (all the locations inside the curve). It then is a small step to extend it to a fuzzy set [10] of locations, where each location is represented by a point and each location has a membership grade associated [2]. The membership grades for regions are interpreted in a veristic way [4]:

¹ Possibly, the region can have holes, but in this contribution only regions without holes are considered.
² Note that the membership grade are not derived from coordinates of the point, but assigned which each point individually.

Definition 1 (*fuzzy region \tilde{A}*).

$$\tilde{A} = \{(p, \mu_{\tilde{A}}(p)) \mid p \in U, \mu_{\tilde{A}}(p) > 0\}$$

where

$$\begin{aligned} \mu_{\tilde{A}} : U &\rightarrow [0, 1] \\ p &\mapsto \mu_{\tilde{A}}(p) \end{aligned}$$

Here U is the universe (commonly \mathbb{R}^2).

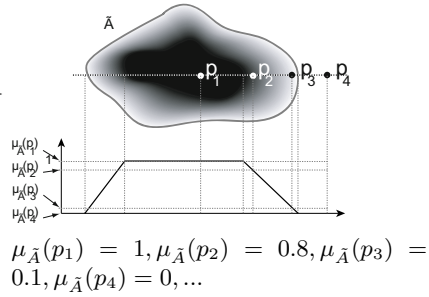


Fig. 2. A fuzzy region, for illustration purposes the fuzzy region is delimited by a grey line. The membership grades for points belonging to the region are shaded, ranging from black (membership grade 1) to white (membership grade 0). A cross section shows how the membership grades along the dotted line evolve.

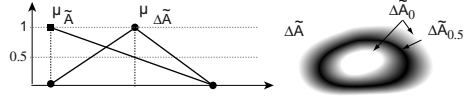
all locations belong to the region, but some more than others. A possibilistic interpretation can be used to represent fuzzy points, and although a similar approach can be applied, this contribution focusses on fuzzy regions. On fig. 2, an example of a fuzzy region is shown.

3 Topology

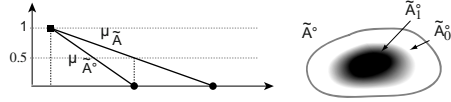
3.1 Fuzzy Concepts

To extend the 9-intersection model, appropriate definitions for interior, exterior and boundary are needed. The approach is similar to the extension in [1] for regions with broad boundaries. The definitions bear resemblance to the work of Du [3], where a fuzzy border around a crisp region was defined to create a fuzzy region. This is similar to our work in [7], but with such regions it is impossible to make a closed calculus: the intersection for instance cannot always be represented by a new such region. For the fuzzy boundary, several considerations are in place: as the region itself is fuzzy, it is logical that its boundary will be a fuzzy entity. It should be defined such that it remains compatible with the crisp topology. First, consider the case of a fuzzy region which has a membership grade 1 in some central part, and has continuously, decreasing membership grades away from this central part outward. Points p with membership grade $\mu_{\tilde{A}}(p) = 0$ or $\mu_{\tilde{A}}(p) = 1$ in the original region can be considered not to belong to the boundary, as they are completely outside, respectively completely inside the region. Points with a membership grade that is closer to 0 or 1 belong less to the boundary than points with a membership grade closer to 0.5. Because of this 0.5 will play a crucial part: points p for which $\mu_{\tilde{A}}(p) = 0.5$ will be said to completely belong to the boundary (and thus $\mu_{\Delta\tilde{A}}(p) = 1$). The more $\mu_{\tilde{A}}(p)$ differs from 0.5, the lower $\mu_{\Delta\tilde{A}}(p)$ should be. This can be accomplished with e.g. the function: $2(0.5 - |0.5 - x|), \forall x \in [0, 1]$; illustrated on fig. 3a, the resulting boundary is shown on fig. 3b and fig. 3c. The particular function was chosen as it keeps some properties of the original

Definition 2 (boundary $\Delta\tilde{A}$).
 $\Delta\tilde{A} = \{(p, \max(\sup\{\alpha | p \in \partial\tilde{A}_\alpha, 2(0.5 - |\mu_{\tilde{A}}(p) - 0.5|)\})\}$



Definition 3 (interior \tilde{A}°).
 $\tilde{A}^\circ = \{(p, \mu_{\tilde{A}^\circ}(p))\}$ where
 $\mu_{\tilde{A}^\circ} : U \rightarrow [0, 1]$
 $p \mapsto \begin{cases} 0 & \mu_{\tilde{A}}(p) \leq 0.5 \\ 1 - \mu_{\Delta\tilde{A}}(p) & \text{elsewhere} \end{cases}$



Definition 4 (exterior \tilde{A}^-).
 $\tilde{A}^- = \{(p, \mu_{\tilde{A}^-}(p))\}$ where
 $\mu_{\tilde{A}^-} : U \rightarrow [0, 1]$
 $p \mapsto \begin{cases} 0 & \mu_{\tilde{A}}(p) \geq 0.5 \\ 1 - \mu_{\Delta\tilde{A}}(p) & \text{elsewhere} \end{cases}$

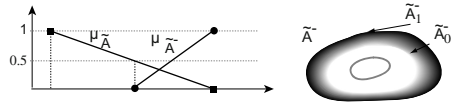


Fig. 3. Illustration of the fuzzy boundary, interior and exterior; for each, the definition, a plot of the membership function (with the membership function for \tilde{A}) and a graphical representation of the resulting regions using grey scales

function when only considering the values in $[0, 0.5]$ or $[0.5, 1]$ (e.g. linearity). As crisp regions (and broad boundary regions) are special cases of fuzzy regions, the definition must be such that remains compatible (this is also required if for instance a fuzzy region's has a sudden transition of membership grade at some part). This is achieved by considering the boundaries $\partial\tilde{A}_\alpha$ at every α -level α . The definition of the boundary of a fuzzy region \tilde{A} is given in fig. 3. In general, it is not mandatory to have points with a membership grade 0.5. For a region it may therefore be possible that at some sides there exists a path from the core to the outside which does not cross points for which $\mu_{\Delta\tilde{A}}(p) = 1$ (the current definition guarantees there always will be points for which $\mu_{\Delta\tilde{A}}(p) > 0$ that will be crossed). The full impact of this is currently under investigation. For the interior, first consider the points that are completely part of the region ($\mu_{\tilde{A}}(p) = 1$); this will also be the core of the interior. Points p just outside this core, but still belonging to a substantial extent to the region (i.e. $\mu_{\tilde{A}}(p) > 0.5$) are also considered to be part of the interior to a lesser extent. Points p with a membership grade $\mu_{\tilde{A}}(p) \leq 0.5$, are considered not to belong to the interior, fig. 3. The exterior is defined analogously, fig. 3.

3.2 Intersection Matrices

Description. Once the concepts of interior, boundary and exterior are known, the 9-intersection matrix for fuzzy regions is:

$$\begin{pmatrix} h(\tilde{A}^\circ \tilde{\cap} \tilde{B}^\circ) & h(\tilde{A}^\circ \tilde{\cap} \Delta\tilde{B}) & h(\tilde{A}^\circ \tilde{\cap} \tilde{B}^-) \\ h(\Delta\tilde{A} \tilde{\cap} \tilde{B}^\circ) & h(\Delta\tilde{A} \tilde{\cap} \Delta\tilde{B}) & h(\Delta\tilde{A} \tilde{\cap} \tilde{B}^-) \\ h(\tilde{A}^- \tilde{\cap} \tilde{B}^\circ) & h(\tilde{A}^- \tilde{\cap} \Delta\tilde{B}) & h(\tilde{A}^- \tilde{\cap} \tilde{B}^-) \end{pmatrix} = \begin{pmatrix} d & c_1 & e_1 \\ c_2 & b & a_1 \\ e_2 & a_2 & 1 \end{pmatrix} \quad (2)$$

Here $h(X)$ is the notation for the *height*(X) of a fuzzy set X , i.e. the highest membership grade in the set [6]. The intersection is the fuzzy intersection, by means of a t-norm (e.g. minimum). The matrix elements can have any value in the range $[0, 1]$, which impacts how the matrices will be interpreted; and are named (apart from the bottom right element, which is always 1). A full case study has been made, yielding a large number of cases; but as there are similarities there are ways of grouping them. We opted for groups more or less resembling the 44 cases Clementini listed for the broad boundary model ([1]). Two cases are illustrated below, their numbers referring to their number in the 44 cases.

- Case 1: The first case is when both regions \tilde{A} and \tilde{B} are completely disjoint.

$$\tilde{A} \text{ disjoint } \tilde{B} \quad \begin{array}{c} \tilde{A} \\ \tilde{B} \end{array} \quad \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \tilde{B} \\ \tilde{A} \end{array} \quad \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & a_1 \\ 1 & a_2 & 1 \end{pmatrix} \text{ with } \begin{cases} a_1, a_2 \in]0, 1[\\ b, d, c_1, c_2 = 0 \\ e_1, e_2 = 1 \end{cases}$$

This matrix is similar to the nine-intersection matrix of disjoint crisp regions, and to the nine-intersection matrix of disjoint regions with broad boundaries. The main difference is that the elements a_1 and a_2 are both in the range $]0, 1[$, rather than 1. We can only be sure that they equal 1 if there are elements in the region with membership grade 0.5 (membership grade in the boundary is then 1). While this is the case in this assumption, we cannot be sure of this in general.

- Case 3 occurs when the boundary $\Delta\tilde{A}$ intersects with the boundary $\Delta\tilde{B}$ and when the interior \tilde{A}° also intersects with the boundary $\Delta\tilde{B}$ (or vice versa: case 6). An intersection between the boundaries implies that $h(\Delta\tilde{A} \tilde{\cap} \Delta\tilde{B}) > 0$. It is possible for this matrix element to equal 1, when there are points for which $\mu_{\tilde{A}}(p) = \mu_{\tilde{B}}(p) = 0.5$. Second, it is possible for $\Delta\tilde{A}$ to intersect with \tilde{B}° . Even further, if the interior of \tilde{A} intersects with the boundary of \tilde{B} , this means that $h(\tilde{A}^\circ \tilde{\cap} \Delta\tilde{B}) > 0$. It is possible for this element to equal 1, if there are points p such that $\mu_{\tilde{A}}(p) = 1 \wedge \mu_{\tilde{B}}(p) = 0.5$; but it is not possible for this element to equal 0 (as this would yield a different case).

$$\begin{array}{l} \Delta\tilde{A} \tilde{\cap} \tilde{B}^\circ = \emptyset \\ \tilde{A}^\circ \tilde{\cap} \tilde{B}^\circ = \emptyset \end{array} \quad \begin{array}{c} \tilde{A} \\ \tilde{B} \end{array} \quad \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \tilde{B} \\ \tilde{A} \end{array} \quad \begin{pmatrix} 0 & c_1 & 1 \\ 0 & b & a_1 \\ 1 & a_2 & 1 \end{pmatrix} \text{ with } \{ a_1, a_2, b, c_1 \in]0, 1[\}$$

$$\begin{array}{l} \Delta\tilde{A} \tilde{\cap} \tilde{B}^\circ \neq \emptyset \\ \tilde{A}^\circ \tilde{\cap} \tilde{B}^\circ = \emptyset \end{array} \quad \begin{array}{c} \tilde{A} \\ \tilde{B} \end{array} \quad \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \tilde{B} \\ \tilde{A} \end{array} \quad \begin{pmatrix} 0 & c_1 & 1 \\ c_2 & b & a_1 \\ 1 & a_2 & 1 \end{pmatrix} \text{ with } \begin{cases} a_1, a_2, b, c_1 \in]0, 1[\\ c_2 \in]0, 1[\end{cases}$$

$$\begin{array}{l} \Delta\tilde{A} \tilde{\cap} \tilde{B}^\circ \neq \emptyset \\ \tilde{A}^\circ \tilde{\cap} \tilde{B}^\circ \neq \emptyset \end{array} \quad \begin{array}{c} \tilde{A} \\ \tilde{B} \end{array} \quad \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \text{---} \\ \text{---} \end{array} \quad \begin{array}{c} \tilde{B} \\ \tilde{A} \end{array} \quad \begin{pmatrix} d & c_1 & 1 \\ c_2 & b & a_1 \\ 1 & a_2 & 1 \end{pmatrix} \text{ with } \begin{cases} a_1, a_2, b, c_1 \in]0, 1[\\ c_2, d \in]0, 1[\end{cases}$$

The above matrices can be combined: $\begin{pmatrix} d & c_1 & 1 \\ c_2 & b & a_1 \\ 1 & a_2 & 1 \end{pmatrix}$ with $\begin{cases} a_1, a_2, b, c_1 \in]0, 1[\\ c_2 \in]0, 1[\\ d \in [0, 1[\end{cases}$

Other cases. Due to lack of space, we cannot list all the cases in this contribution. To help with the example, we will only list the conditions on the matrix elements for cases 4,5,10 and 11.

- case 4: the boundary $\Delta\tilde{A}$ intersects with the boundary $\Delta\tilde{B}$ and the interior \tilde{A}° is entirely located inside the broad boundary $\Delta\tilde{B}$ (case 7 is the symmetrical case); $a_1 \in]0, 1[$, $a_2 \in]0, 1[$, $b \in]0, 1[$, $c_1 \in]0, 1[$, $c_2 \in]0, 1[$, $d \in]0, 1[$, $e_1 \in]0, 1[$, $e_2 = 1$.
- case 5: the boundary $\Delta\tilde{A}$ and a fortiori the interior \tilde{A}° are completely inside the boundary $\Delta\tilde{B}$ (case 8 is the symmetrical case); $a_1 \in]0, 1[$, $a_2 \in]0, 1[$, $b \in]0, 1[$, $c_1 \in]0, 1[$, $c_2 \in]0, 1[$, $d \in]0, 1[$, $e_1 \in]0, 1[$, $e_2 = 1$.
- case 10: the boundary $\Delta\tilde{A}$ intersects with the interior \tilde{B}° , the boundary $\Delta\tilde{B}$ and the exterior \tilde{B}^- . The interior \tilde{A}° can intersect with either $\Delta\tilde{B}$ (to a degree of up to 1) and with \tilde{B}° , \tilde{B}^- (to a degree strictly less than 1); $a_1 \in]0, 1[$, $a_2 \in]0, 1[$, $b \in]0, 1[$, $c_1 \in]0, 1[$, $c_2 \in]0, 1[$, $d \in]0, 1[$, $e_1 \in]0, 1[$, $e_2 = 1$.
- case 11: the boundary $\Delta\tilde{A}$ intersects with the interior \tilde{B}° and the boundary $\Delta\tilde{B}$. An intersection with the exterior \tilde{B}^- is possible, but only to a degree strictly less than 1. The interior \tilde{A}° can intersect with either \tilde{B}° , $\Delta\tilde{B}$ and \tilde{B}^- ; $a_1 \in]0, 1[$, $a_2 \in]0, 1[$, $b \in]0, 1[$, $c_1 \in]0, 1[$, $c_2 \in]0, 1[$, $d \in]0, 1[$, $e_1 \in]0, 1[$, $e_2 \in]0, 1[$

In the conceptual neighbourhood graph (a graph where two cases are considered neighbours and thus connected when the changes between them are as small as possible), cases 4 and 11 are connected with cases 5 and 10.

Interpretation. When grouped in 44 cases, each matrix element is restricted to one of the following intervals: $[0]$, $[0, 1[$, $]0, 1]$, $[1]$. These can be said to have an intuitive order: $[0, 1[$ can be said to be *smaller* than $]0, 1]$ because the largest possible value is smaller; $]0, 1]$ *smaller* than 1 as smaller values are possible, and similarly 0 is smaller than $[0, 1[$. The case matrices can be grouped according to each element restriction, and a conceptual neighbourhood graph can be constructed. For two regions, their intersection matrix will contain 9 values in the range $[0, 1]$; for each value the matching cases are sought: if the value satisfies the element constraint, the case is retained. After this, the number of matching cases will be reduced (not necessarily to 1); and they will be neighbours in the conceptual neighbourhood graph. A number will determine how well the each one matches with the given matrix. To find this number, we apply the rule that values smaller than 0.5 belong more $[0, 1[$ than to $]0, 1]$; whereas values greater than 0.5 have the opposite property (this is an intuitive rule, the value 0.5 is present in both intervals). Now, *match values* are assigned for every matrix element x that distinguishes two groups and for every case i that is in either of the two groups. These match values represent how well each matrix element matches; aggregating them for a case i (using a t-norm), yields a single value expressing how well the given matrix matches this case.

Definition 5 (Match value m_x^i for a case i and a matrix element x)

$$m_x^i = \begin{cases} x & \text{if range of case } i =]0, 1[\\ 1 - x & \text{if range of case } i = [0, 1[\end{cases}$$

3.3 Example

To illustrate, consider two regions that yield the following intersection matrix:

$$\begin{pmatrix} d & c_1 & e_1 \\ c_2 & b & a_1 \\ e_2 & a_2 & 1 \end{pmatrix} = \begin{pmatrix} 0.6 & 0.7 & 0.4 \\ 0.4 & 0.6 & 0.3 \\ 1 & 0.6 & 1 \end{pmatrix}$$

This example will be examined in further detail.

- a_1 and a_2 : For the element a_1 , there are only two groups: $]0, 1]$ and $[0, 1[$, and both groups are possible for the given value. We come to the same conclusion for a_2 . The possible cases with these values are all intersection cases.
- b : For the value of b , there are three groups: $0,]0, 1]$ and $[0, 1[$. Obviously, the value for b is not 0, so the cases with $b = 0$ (only case 1) are not possible.
- c_1 and c_2 : For the value of c_1 , there are three groups: $0,]0, 1]$ and $[0, 1[$. The cases with $c_1 = 0$ (cases 1 and 39) will not match our given matrix. Similarly, cases 1 and 40 don't match with the value of c_2 .
- d : For the value d , there are three groups: $0, [0, 1[$ and 1 . In the example, $d = 0.6$; only the cases with $[0, 1[$ remain: $\{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 17\}$
- e_1 and e_2 : For e_1 , there are three groups: $0, [0, 1[$ and 1 ; for $e_1 = 0.4$, only $[0, 1[$ is applicable, leaving: $\{4, 5, 10, 11, 14, 17, 19, 20, 23, 24, 25, 26, 29, 30, 32, 35, 36, 38, 41, 42, 43, 44\}$. For $e_2 = 1$ it is similar, but now the cases for which $e_2 = 1$ is required. This leaves: $\{1, 2, 3, 4, 5, 6, 9, 10, 11, 18, 19, 20, 27, 28, 39\}$

The intersection of all of the above cases results in the cases which are appropriate for the example: $\{4, 5, 10, 11\}$. The differences between the cases 4, 5, 10 and 11 is in the elements that match with the values of $a_1 = 0.3$, $c_1 = 0.7$ and $c_2 = 0.4$. To find the closest match, we need the different match values.

$$\begin{array}{ll}
 a_1 = 0.3 \text{ cases } 4,10: & 0 < a_1 \leq 1 \Rightarrow m_{a_1}^4 = m_{a_1}^{10} = 0.3 \\
 & \text{cases } 5,11: & 0 \leq a_1 < 1 \Rightarrow m_{a_1}^5 = m_{a_1}^{11} = 0.7 \\
 c_1 = 0.7 \text{ case } 5: & 0 \leq c_1 < 1 \Rightarrow m_{c_1}^5 = 0.3 \\
 & \text{cases } 4,10,11: & 0 < c_1 \leq 1 \Rightarrow m_{c_1}^4 = m_{c_1}^{10} = m_{c_1}^{11} = 0.7 \\
 c_2 = 0.4 \text{ cases } 4,5: & 0 \leq c_2 < 1 \Rightarrow m_{c_2}^4 = m_{c_2}^5 = 0.6 \\
 & \text{cases } 10,11 & 0 < c_2 \leq 1 \Rightarrow m_{c_2}^{10} = m_{c_2}^{11} = 0.4
 \end{array}$$

The match values for a_1 indicate that cases 5 and 11 are a better match for the example than cases 4 and 10. The match values for c_1 , show that cases 4,10,11 are a better match than case 5; according to c_2 cases 4 and 5 are a better match than cases 10 and 11. The aggregation of the match values yields:

$$\begin{array}{l}
 \text{case } 4 : \min\{m_{a_1}^4, m_{c_1}^4, m_{c_2}^4\} = \min\{0.3, 0.7, 0.6\} = 0.3 \\
 \text{case } 5 : \min\{m_{a_1}^5, m_{c_1}^5, m_{c_2}^5\} = \min\{0.7, 0.3, 0.6\} = 0.3 \\
 \text{case } 10 : \min\{m_{a_1}^{10}, m_{c_1}^{10}, m_{c_2}^{10}\} = \min\{0.3, 0.7, 0.4\} = 0.3 \\
 \text{case } 11 : \min\{m_{a_1}^{11}, m_{c_1}^{11}, m_{c_2}^{11}\} = \min\{0.7, 0.7, 0.4\} = 0.4
 \end{array}$$

As the aggregated match value is the highest for case 11, the topology for the example is closer to this case than to any of the other three cases. However, the differences between the aggregated match values are very small, so the regions in the example still resemble the other three cases quite closely.

4 Conclusion

In this paper, we presented a qualitative approach for judging the topology of fuzzy regions, represented by our model. The approach yields a soft classification in which predefined cases are matched and a degree of this match is provided.

The topology is a generalization of Clementini's broad boundary model: the fuzzy region model can be used to represent broad regions, and if all points inside the inner region are assigned membership grade 1, all points of the broad boundary are assigned a membership grade 0.5, and all points outside of the outer boundary membership grade 0, the topology cases and matrices match. It also generalizes the topology for crisp regions: assigning points inside the region the membership grade 1, and points outside the crisp region membership grade 0, the fuzzy methodology results in the classical 9-intersection model. The methodology has been illustrated using our theoretical model, but is equally applicable on the models for implementation purposes we derived from this model ([8], [9]).

References

1. Clementini, E., Di Felice, P.: An algebraic model for spatial objects with undetermined boundaries. In: GISDATA Specialist Meeting - revised version (1994)
2. Cohn, A.G., Gotts, N.M.: Spatial regions with undetermined boundaries. In: Proceedings of the Second ACM Workshop on Advances in GIS, pp. 52–59 (1994)
3. Du, S., Qin, Q., Wang, Q., Li, B.: Fuzzy description of topological relations I: a unified fuzzy 9-Intersection model. In: Wang, L., Chen, K., S. Ong, Y. (eds.) ICNC 2005. LNCS, vol. 3612, pp. 1261–1273. Springer, Heidelberg (2005)
4. Dubois, D., Prade, H.: Fundamentals of Fuzzy Sets. Kluwer Academic Pub., Dordrecht (2000)
5. Egenhofer, M.J., Sharma, J.: Topological Relations Between Regions in R^2 and Z^2 . In: Abel, D.J., Ooi, B.-C. (eds.) SSD 1993. LNCS, vol. 692, pp. 316–336. Springer, Heidelberg (1993)
6. Klir, G.J., Yuan, B.: Fuzzy sets and fuzzy logic: Theory and applications. Prentice Hall, New Jersey (1995)
7. Verstraete, J., Van der Cruyssen, B., De Caluwe, R.: Assigning Membership Degrees to Points of Fuzzy Boundaries. In: NAFIPS 2000 Conf. Proc., Atlanta, pp. 444–447 (2000)
8. Verstraete, J., De Tré, G., Hallez, A.: Adapting TIN-layers to Represent Fuzzy Geographic Information. In: The Seventh Meeting of the EURO Working Group on Fuzzy Sets, pp. 57–62 (2002)
9. Verstraete, J., Hallez, A., De Tré, G.: Bitmap Based Structures for the modelling of Fuzzy Entities. Special issue of Control & Cybernetics 35(1), 147–164 (2006)
10. Zadeh, L.A.: Fuzzy Sets. Information and Control 1 3, 338–353 (1965)

Fuzzy $Q(\lambda)$ -Learning Algorithm

Roman Zajdel

Faculty of Electrical and Computer Engineering,
Rzeszow University of Technology, W. Pola 2, 35-959 Rzeszow, Poland
rzajdel@prz-rzeszow.pl

Abstract. The adaptation of temporal differences method $TD(\lambda > 0)$ to reinforcement learning algorithms with fuzzy approximation of action-value function is proposed. Eligibility traces are updated using the normalized degree of activation of fuzzy rules. The two types of fuzzy reinforcement learning algorithm are formulated: with discrete and with continuous action values. These new algorithms are practically tested in the control of two typical models of continuous object, like ball-beam and cart-pole system. The achievement results are compared with two popular reinforcement learning algorithms with CMAC and table approximation of action-value function.

Keywords: reinforcement learning, Fuzzy $Q(\lambda)$ -learning, CMAC.

1 Introduction

Typical reinforcement learning algorithms use table form of value function [1] or action-value function [11]. These approaches are simple but are limited to problems with small numbers of states and actions [9]. The problem is the number of episodes needed to achieve a stable and complete strategy. If some action-state pairs are not represented in learning process, the table representation of strategy will not be complete there. This problem can be solved by generalization from previously experienced states to ones that have never been seen. The good form of generalization are the cerebellar model articulation controller (CMAC) [3], [8], [13] and fuzzy logic systems [2], [4]. CMAC has attractive characteristics of local generalization and moreover, in learning process only a small number of weights are updated per time. The disadvantages of CMAC is its memory wear, which grows exponentially together with rise of number of input variables [7]. The use of fuzzy logic in learning system causes that the gained knowledge are human friendly. There also exist some fuzzy CMAC models, that are often the particular case of Takagi-Sugeno fuzzy inference system [7], [6]. The implementation of eligibility traces method $TD(\lambda)$ in reinforcement learning algorithms with CMAC approximation [8], [9] or fuzzy inference systems [4], [10] speeds up the learning process. $TD(\lambda)$ method applied to fuzzy systems does not utilize information about activation degrees of fuzzy rules so far. This article proposes the modified method of eligibility traces update that uses information about the value of fuzzy rules activity. Moreover, this modification enables to modify all

‘active’ eligibility traces corresponding not only to single rule [4], [10] but to all active fuzzy rules in single learning step.

2 Reinforcement Learning

Reinforcement learning addresses the problem of the agent that must learn to perform a task through trial and error interaction with an unknown environment [9]. The agent and the environment interact continuously until the terminal state is not reached. The agent senses the environment through its sensors and, based on its current sensory inputs, selects an action to perform in the environment. Depending on the effect of its action, the agent obtains a reward. Its goal is to maximize the discounted sum of future reinforcements r_t received in long run, what is usually formalized as $\sum_{t=0}^{\infty} \gamma^t r_t$, where $\gamma \in [0, 1]$ is the agent’s discount rate. The application of nonlinear models is influenced by the fact of the complexity of training data. The models presented below seem to be most suitable for this particular problem.

2.1 Q(λ)-Learning

There exist different types of reinforcement learning algorithms. The Q-learning proposed by Watkins [11] is one of the most often used. This algorithm computes the table of all values $Q(\mathbf{s}, a)$ (called Q-table), by successive approximations. $Q(\mathbf{s}, a)$ represents the expected payoff that agent can obtain in state $\mathbf{s} = \{s_1, s_2, \dots, s_N\}$ after it performs action a . The Q-table is updated according to the following formula:

$$Q(\mathbf{s}, a) \leftarrow Q(\mathbf{s}, a) + \beta \Delta Q(\mathbf{s}, a), \quad (1)$$

where

$$\Delta Q(\mathbf{s}, a) = r + \gamma \max_{a'} Q(\mathbf{s}', a') - Q(\mathbf{s}, a). \quad (2)$$

and the maximization operator refers to the action value a' which may be performed in next state \mathbf{s}' and $\beta \in (0, 1]$ is the learning rate. The basic Q-learning algorithm (1-step Q-learning) can be significantly improved considering the history of state activation represented by eligibility traces. The eligibility trace is parametrized by recency factor $\lambda \in [0, 1]$, therefore this enriched learning method is called Q(λ)-learning [9], [11]. The eligibility trace of each state becomes large after state activation and then it decreases exponentially until the state is not visited again [3].

2.2 CMACQ(λ)-Learning

CMAC also called tile coding is one of the most popular and widely used form of function approximation [3], [5]. The universe of each state variable is covered by several overlapping table approximator layers called tilings and signed further

as L . The tiling consists from some number of elements called a tile. The action-value function approximated by CMAC is the sum of the active tails q_i :

$$Q(\mathbf{s}, a) = \sum_{i=1}^I \varphi_i(\mathbf{s}, a) q_i, \tag{3}$$

where I is the number of the CMAC elements and $\varphi_i(\mathbf{s}, a)$ is the activity indicator of i -th tail q_i as follows:

$$\varphi_i(\mathbf{s}, a) \leftarrow \begin{cases} 1 & \text{if } i\text{-th tile is active in state } \mathbf{s} \text{ and action } a \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

The update rule for each tile is [5]

$$q_i \leftarrow q_i + \alpha \phi_i(\mathbf{s}, a) \Delta Q(\mathbf{s}, a), \tag{5}$$

in which α is a constant step size. Similarly to $Q(\lambda > 0)$ -learning algorithm with tabular representation of action-value function Q , the $TD(\lambda > 0)$ algorithm can be applied to CMAC, and then the update rule is

$$q_i \leftarrow q_i + \alpha e_i(\mathbf{s}, a) \Delta Q(\mathbf{s}, a), \tag{6}$$

where

$$e_i(\mathbf{s}, a) \leftarrow \begin{cases} \lambda \gamma e_i(\mathbf{s}, a) + 1 & \text{if } i\text{-th tile is active} \\ \lambda \gamma e_i(\mathbf{s}, a) & \text{otherwise.} \end{cases} \tag{7}$$

It is worth to notice that the implementation of eligibility traces in CMAC causes that not only L -active tiles is actualised in the current step, but the tiles active in the previous steps are also updated. Considering the history of tiles activity represented by eligibility traces the efficiency of the algorithm can be significantly improved.

2.3 Fuzzy $Q(\lambda)$ -Learning

Application of the Takagi-Sugeno fuzzy inference system to approximate action value function is presented in [2]. This subsection is mainly based on this work and moreover, contain an adaptation to this eligibility traces method.

The goal of a fuzzy approximator is to transform a N -dimensional state space \mathbf{s} to a real-valued action function a_i and its corresponding action-value function q_i . Each n -th ($n = 1, \dots, N$) input state domain is covered by fuzzy sets denoted by B_{ni} , where $i = 1, \dots, I$ is the rule index. Each fuzzy set is characterised by membership function $\mu_{B_{ni}}$. In order to fuzzy approximate the action-value function $Q(\mathbf{s}, a)$, the Takagi-Sugeno fuzzy inference system is used with rule base as *if ... then ...*. For the system with N input states the rule base consisted of I rules can be written as follows:

$$R_i: \text{ If } s_1 \text{ is } B_{1i} \text{ and } \dots s_N \text{ is } B_{Ni} \text{ then } a \text{ is } a_i \text{ and } q \text{ is } q_i,$$

where a and q are the outputs of the fuzzy system and q_i is the discrete action-value associated to the action value a_i . Thus the action-value prediction is an output of the Takagi-Sugeno inference system and it is given by [2]

$$Q(\mathbf{s}, A(\mathbf{s})) = \frac{\sum_{i=1}^I \beta_i(\mathbf{s}) \cdot q_i}{\sum_{i=1}^I \beta_i(\mathbf{s})}, \tag{8}$$

where

$$\beta_i(\mathbf{s}) = \mu_{B_{1i}}(s_1) \cdot \mu_{B_{2i}}(s_2) \cdot \dots \cdot \mu_{B_{Ni}}(s_N) \tag{9}$$

is the degree of activation of the premise of i -th rule and $A(\mathbf{s})$ is the continuous action. In order to simplify formula (8) the normalized activation of the premise (or simple fuzzy rule) is defined as follows

$$\bar{\beta}_i(\mathbf{s}) = \frac{\beta_i(\mathbf{s})}{\sum_{i=1}^I \beta_i(\mathbf{s})} \tag{10}$$

and then (8) can be rewritten as [2]

$$Q(\mathbf{s}, A(\mathbf{s})) = \sum_{i=1}^I \bar{\beta}_i(\mathbf{s}) \cdot q_i. \tag{11}$$

The action executed by an agent can be determined in two manners. The first method utilizes the normalized degree of activation of the fuzzy rules in order to calculate the weighted sum of the active actions which are the consequences of fuzzy rules

$$A(\mathbf{s}) = \sum_{i=1}^I \bar{\beta}_i(\mathbf{s}) \cdot a_i. \tag{12}$$

Since the action obtained from (12) is a continuous function of state variables this system will be signed from now as FQCA(λ)-learning (Fuzzy Q-learning with Continuous Action). The continuous action corresponds to the continuous action-value function $Q(\mathbf{s}, A(\mathbf{s}))$ (11).

The second manner of obtaining the action relies to rounding the action given by (12) to the nearest action from the discrete set of actions $\mathbb{A} = \{a_1, \dots, a_i\}$. This procedure is closer to the definition of Markov decision problem, and such a system will be further signed as FQ(λ)-learning. The update of function q_i for all active states is performed as

$$q_i \leftarrow q_i + \alpha \bar{\beta}_i(\mathbf{s}) \Delta Q(\mathbf{s}, a). \tag{13}$$

Adaptation to described above algorithm of the eligibility traces method gives the following update rule

$$q_i \leftarrow q_i + \alpha e_i(\mathbf{s}, a) \Delta Q(\mathbf{s}, a), \tag{14}$$

in which the eligibility trace e_i of fuzzy rule is defined as

$$e_i(\mathbf{s}, a) \leftarrow \begin{cases} \lambda \gamma e_i(\mathbf{s}, a) + \beta_i(\mathbf{s}) / \max_i(\beta_i(\mathbf{s})) & \text{if } i\text{-th rule is active} \\ \lambda \gamma e_i(\mathbf{s}, a) & \text{otherwise.} \end{cases} \tag{15}$$

If the input states variables are covered by triangular fuzzy sets as on Fig. 2 and only one i -th rule reaches a maximum activity, the activity degree of the rule $\beta_i(\mathbf{s}) = 1$ and hence the quotient $\beta_i(\mathbf{s}) / \max_i(\beta_i(\mathbf{s})) = 1$ and equation (15) becomes (7). The additional advantage taken from triangular fuzzy sets from z Fig. 2 is that $(\forall \mathbf{s}) \sum_{i=1}^I \beta_i(\mathbf{s}) = 1$ and then from (10) one obtains $\tilde{\beta}_i(\mathbf{s}) = \beta_i(\mathbf{s})$, which simplifies the determination of $Q(\mathbf{s}, A(\mathbf{s}))$ and $A(\mathbf{s})$.

3 Empirical Results

Computational experiments were performed using two nonlinear objects: cart-pole [1] and ball-beam system [12] (Fig. 1). It was also assumed that if the pole of the cart-pole system does not deflect greater than 12° from the true ($|\theta| \leq 12^\circ$, see Fig. 1 (a)) or the cart does not reach the end of the track allowed ($|x| \leq 2.4m$) and the ball does not roll down from the beam ($|x| \leq 1m$, see Fig. 1 (b)) for 100000 time steps (2000s), then the system learns to control the object. Each experiment was repeated 10 times. For each of the control objects four algorithms were tested: $Q(\lambda)$ -learning, CMAC $Q(\lambda)$ -learning, $FQ(\lambda)$ -learning and $FQCA(\lambda)$ -learning. For both objects controlled by a fuzzy algorithms $FQ(\lambda)$ -learning and $FQCA(\lambda)$ -learning, the domains of all state variables were covered by five triangular fuzzy sets (see Fig. 2). In CMAC $Q(\lambda)$ -learning algorithm the domains of input states variables in first tiling were split into five equal intervals and the number of tilings was treated as a parameter from the range $\langle 2, 12 \rangle$.

For ball-beam system the learning factors for algorithms: $Q(\lambda)$ -learning, $FQCA(\lambda)$ -learning and $FQ(\lambda)$ -learning were the same: $\alpha = 0.1$, $\gamma = 0.995$ and $\lambda = 0.1$. For CMAC (λ) algorithm only the step size of action-value function was different and was assumed as $\alpha = 0.005/L$ [9]. In $Q(\lambda)$ -learning algorithm each of the state variables $x \in \langle -1; 1 \rangle$ and $\dot{x} \in \langle -1; 1 \rangle$ were divided in domain points $\{-0.5, -0.1, 0.1, 0.5\}$, which gave five intervals of state variables: $\langle -1; -0.5 \rangle$, $\langle -0.5; -0.1 \rangle$, $\langle -0.1; 0.1 \rangle$, $\langle 0.1; 0.5 \rangle$ and $\langle 0.5; 1 \rangle$. The actions (market by angle θ in Fig. 1 (b)) were chosen from the set $\mathbb{A} = \{-\pi/4, -\pi/8, 0, \pi/8, \pi/4\} [^\circ]$ using ε -greedy method ($\varepsilon = 0.1$). For the cart-pole system and $Q(\lambda)$ -learning

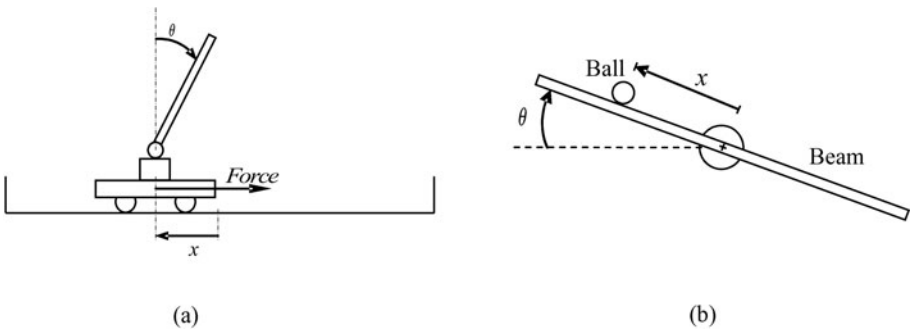


Fig. 1. The cart-pole (a) and the ball-beam system (b)

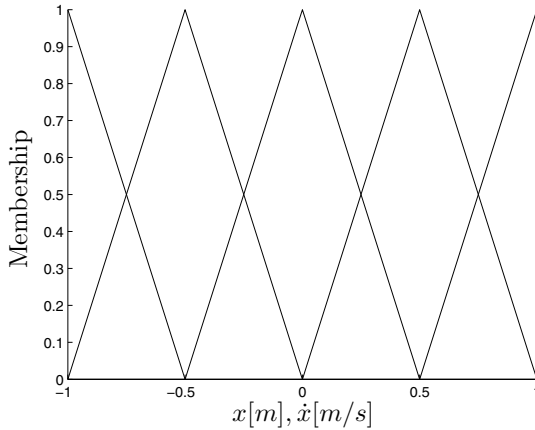


Fig. 2. Fuzzy sets for state variables of the ball-beam system. Except domains, the fuzzy sets for all state variables of the cart-pole system were the same.

Table 1. Performance comparison of the different learning control algorithms

Algorithms	Tailings	Ball-Beam			Cart-Pole		
		Trails		Avg. time	Trails		Avg. time
		Min	Avg		Min	Avg	
CMACQ(λ)-learning	2	2	5.7	24%	369	656	6%
	3	2	4.6	29%	90	446	7%
	4	2	3.8	35%	501	705	8%
	5	1	4.1	41%	240	433	9%
	6	1	3.6	49%	450	707	12%
	7	1	2.4	57%	267	551	12%
	8	1	3.6	63%	51	597	13%
	9	2	3.5	72%	322	470	16%
	10	1	3.3	81%	222	682	17%
	11	1	4	92%	452	576	20%
	12	1	3.9	100%	110	407	22%
Q(λ)-learning		3	11	2%	182	289	1%
FQ(λ)-learning		1	4	47%	41	181	95%
FQCA(λ)-learning		3	5.9	53%	31	180	100%

algorithm state variables $x \in \langle -2.4; 2.4 \rangle$ [m], $\dot{x} \in \langle -2; 2 \rangle$ [m/s], $\theta \in \langle -12; 12 \rangle$ [$^\circ$] and $\dot{\theta} \in \langle -200; 200 \rangle$ [$^\circ/s$] were divided in domain points $\{-1.4, -0.4, 0.4, 1.4\}$ [m], $\{-1.2, -0.2, 0, 2, 1.2\}$ [m/s], $\{-6, -1, 1, 6\}$ [$^\circ$] and $\{-100, -10, 10, 100\}$ [$^\circ/s$], respectively, which gave five intervals for each state variable. For example, for angle θ these were: $\langle -12; -6 \rangle$ [$^\circ$], $\langle -6; -1 \rangle$ [$^\circ$], $\langle -1; 1 \rangle$ [$^\circ$], $\langle 1; 6 \rangle$ [$^\circ$] and $\langle 6; 12 \rangle$ [$^\circ$]. The actions (marked by *Force* in Fig. 1 (a)) were chosen from the set $\mathbb{A} = \{-10, 10\}$ [N] by means of the ε -greedy method ($\varepsilon = 0.005$).

The results of the experiments are summarized in the Table III. In the second column the numbers of tiling of CMAC are placed. The minimal and average number of trials needed to obtain semioptimal strategy are put in the third and fourth column for ball-beam system and in the sixth and seventh column for cart-pole system. The time of a single iteration of each algorithm expressed as a percentage of the longest time for each control system is shown in fifth column for ball-beam system and in eighth column for cart-pole one. One observes that for cart-pole system and a large number of tilings the effectiveness of $\text{CMAC}(\lambda)$ algorithm expressed by both trials number slightly increases. The minimal and average number of trials needed to achieve a stable strategy is for fuzzy algorithms significantly lower than for other. However, this occurs at the expense of the time of the single iteration, which in the case of fuzzy algorithms is 4-5 times greater than for 12-tiling CMAC. For the ball-beam system the time of a single iteration of fuzzy algorithms is comparable to CMAC with 6 or 7 tilings. One observes that when increasing the number of state variables, and therefore the number of fuzzy rules, the time complexity of fuzzy reinforcement learning algorithms significantly grows. The $Q(\lambda)$ -learning algorithm with tabular approximation of action-value function achieved the shortest learning time of a single iteration, however the correct partition of the domain of states variable required multiple attempts.

4 Conclusions

This paper proposed the adaptation of temporal differences method $\text{TD}(\lambda > 0)$ for reinforcement Q -learning algorithm with fuzzy approximation of action-value function. This adaptation consisted in the update of the eligibility traces proportionally to the fuzzy rules activity. Two forms of fuzzy approximators was considered, with discrete and with continuous action values. The verification experiments performed on cart-pole and ball-beam system were carried out to assess the correctness of the proposed method. The effectiveness of fuzzy reinforcement learning algorithm was compared to tabular $Q(\lambda)$ -learning and $\text{CMAC}(\lambda)$ algorithms. In the case of the simple ball-beam system, the achieved results were comparable for all algorithms. The advantage of fuzzy algorithm was evident in the experiments with more difficult cart-pole system. For the fuzzy algorithms the satisfactory control strategy was obtained after fewer number of trials in comparison to conventional $Q(\lambda)$ -learning and $\text{CMAC}(\lambda)$. The application of fuzzy reinforcement learning algorithm did not require the arduous consideration of the number of tiles for CMAC-based algorithm. The disadvantage of the fuzzy algorithm was the increased computational complexity.

Acknowledgments. This work was supported by Polish Ministry of Science and Higher Education under the grant 3745/B/T02/2009/36.

References

1. Barto, A.G., Sutton, R.S., Anderson, C.W.: Neuronlike adaptive elements that can solve difficult learning problem. *IEEE Trans. SMC* 13, 834–847 (1983)
2. Bonarini, A., Lazaric, A., Montrone, F., Restelli, M.: Reinforcement distribution in Fuzzy Q-learning. *Fuzzy Sets and Systems* 160, 1420–1443 (2009)
3. Cichosz, P.: Learning systems. WNT, Warsaw (2000) (in Polish)
4. Gu, D., Hu, H.: Accuracy based fuzzy Q-learning for robot behaviours. In: Proc. of the IEEE Int. Conf. on Fuzzy Systems, vol. 3, pp. 1455–1460 (2004)
5. Min, H., Zeng, J., Luo, R.: Fuzzy CMAC with automatic state partition for reinforcement learning. In: Proc. of the First ACM/SIGEVO Summit on Genetic and Evolutionary Computation, pp. 421–428 (2009)
6. Nguyen, M.N., Shi, D., Quek, C.: Self-Organizin Gaussian fuzzy CMAC with Truth Value Restriction. In: Proc. of the Third Int. Conf. on Information Technology and Applications (ICITA 2005), vol. 2, pp. 185–190 (2005)
7. Shi, D., Harkisanka, A., Quek, C.: CMAC with Fuzzy Logic Reasoning. In: Pal, N.R., Kasabov, N., Mudi, R.K., Pal, S., Parui, S.K. (eds.) *ICONIP 2004*. LNCS, vol. 3316, pp. 898–903. Springer, Heidelberg (2004)
8. Sutton, R.S.: Generalization in Reinforcement Learning: Successful Examples Using Sparse Coarse Coding. *Advances in Neural information Processing Systems* 8, 1038–1044 (1996)
9. Sutton, R.S., Barto, A.G.: Reinforcement learning: An Introduction. MIT Press, Cambridge (1998)
10. Theodoridis, T., Hu, H.: The Fuzzy Sars'a'(λ) Learning Approach Applied to a Strategic Route Learning Robot Behaviour. In: Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1767–1772 (2006)
11. Watkins, C.J.C.H.: Learning from delayed Rewards. PhD thesis, Cambridge University, Cambridge, England (1989)
12. Wellstead, P.E.: Introduction to Physical System Modelling, Control System Principles (2000)
13. Xu, X., Hu, D., He, H.: Accelerated Reinforcement learning control using modified CMAC neural networks. In: Proc. of the Ninth Int. Conf. on Neural Information Processing, vol. 5, pp. 2575–2578 (2002)

Part II

**Data Mining, Classification and
Forecasting**

Mining Closed Gradual Patterns

Sarra Ayouni^{1,2}, Anne Laurent², Sadok Ben Yahia¹, and P. Poncelet²

¹ Faculty of Sciences of Tunis, 1060, Campus Universitaire, Tunis, Tunisie
ayouni@lirmm.fr, sadok.benyahia@fst.rnu.tn

² Univ. Montpellier 2. LIRMM – CNRS, 161 rue Ada, Montpellier, France
laurent@lirmm.fr, poncelet@lirmm.fr

Abstract. Mining gradual rules of the form – “*the more A, the more B*” – is more and more grasping the interest of the data mining community. Several approaches have been recently proposed. Unfortunately, in all surveyed approaches, reducing the quantity of mined patterns (and, consequently, the quantity of extracted rules) was not the main concern. To palliate such a drawback, a possible solution consists in using results of Formal Concept Analysis to generate a lossless reduced size nucleus of gradual patterns. To do so, we introduce in this paper a novel closure operator acting on gradual itemsets. Results of the experiments carried out on synthetic datasets showed important profits in terms of compactness of the generated gradual patterns set.

1 Introduction

After having been extensively used in fuzzy command systems, *gradual patterns* have recently been studied within the data mining community. They convey knowledge of the form – “*the more A, the more B*” – [2]. Several approaches and semantics have been proposed in literature. However, there does not exist any work tackling the problem of the management of the many patterns that methods extract. In this paper, we thus propose a lossless reduction of the mined gradual patterns. To reach this goal, our solution consists in using results of Formal Concept Analysis. We introduce a novel Galois connection that is a *sine qua non* condition for extracting closed gradual patterns. These latter patterns will act as a lossless reduced-size nucleus of patterns. The rest of the paper is organized as follows. Section 2 reviews the related work focusing on gradual patterns and some basic notions of the FCA framework. Section 3 introduces our novel Galois connection definition and shows its validity and soundness. Section 4 validates the importance of our approach at reducing the huge number of the extracted gradual closed patterns through experiments carried out over synthetic datasets. Section 5 sketches our future perspectives and presents concluding remarks.

2 Related Work

2.1 Gradual Rules

We consider a data base defined on a schema containing m attributes (X_1, \dots, X_m) defined on domains $\text{dom}(X_i)$ provided with a total order (numeric data

are often considered). A data set \mathcal{D} is a set of m -tuples of $\text{dom}(X_1) \times \dots \times \text{dom}(X_m)$. In this scope, a gradual item is defined as a pair of an attribute and a variation $* \in \{\leq, \geq\}$. Let A be an attribute. The gradual item $A \geq$ means that the attribute A is increasing. It can be interpreted by “the more A ”. A gradual itemset, or gradual tendency, is then defined as a non-empty set list of several gradual items. For instance, the gradual itemset $M = A \geq B \leq$ is interpreted as “The more A and the less B ”.

Example 1. $(Salary) \geq$ is a gradual item meaning that the “Salary” is increasing. $((Age) \geq, (Salary) \geq)$ is a gradual itemset.

Gradual dependencies were introduced in [4], where they are called tendency rules and denoted by $A \rightarrow_t B$. Hüllermeier proposed to perform a linear regression analysis on the contingency diagram depicted from the data set. The validity of the rule is assessed on the basis of the regression coefficients α, β of the line that approximates the points in the contingency diagram.

Another definition has been proposed in [1]. The authors define a gradual dependence as being similar to a functional dependence by considering the degree variations between two objects. According to [1], the gradual dependence $A \Rightarrow B$ holds in a database \mathcal{D} if $\forall o=(x, y)$ and $o'=(x', y') \in \mathcal{D}$, $A(x) < A(x')$ implies $B(y) < B(y')$.

A new definition of gradual dependence was proposed in [8] using fuzzy association rules. The authors take into account the variation strength in the degree of fulfilment of an imprecise property by different objects. Hence, a gradual dependence holds in a database \mathcal{D} if $\forall o=(x, y)$ and $o'=(x', y') \in \mathcal{D}$, $v_{*1}(A(x), A(x'))$ implies $v_{*2}(B(y), B(y'))$, where v_* is a variation degree of an attribute between two different objects. In both propositions [1] and [8], the authors propose to build a modified data set \mathcal{D}' that contains as many rows as there are pairs of distinct objects in the initial data set \mathcal{D} .

Another definition of support and confidence of a gradual itemset, as defined above, was proposed in [5]. In fact, the support of a gradual itemset $A_1^{*1}, \dots, A_p^{*p}$, is defined as the maximal number of rows $\{r_1, \dots, r_l\}$ for which there exists a permutation π such that $\forall j \in [1, l - 1], \forall k \in [1, p]$, it holds $A_k(r_{\pi_j}) *_{*k} A_k(r_{\pi_{j+1}})$. More formally, denoting \mathcal{L} the set of all such sets of rows the support of a gradual itemset is defined as follows.

Definition 1. Let $s=A_1^{*1}, \dots, A_p^{*p}$ be a gradual itemset, we have:

$$supp(s) = \frac{\max_{L_i \in \mathcal{L}} |L_i|}{|\mathcal{D}|}.$$

The authors first proposed a heuristic to compute this support for gradual itemsets, in a level-wise process that considers itemsets of increasing lengths. More recently, [6] proposed an efficient method based on precedence graphs. In this method, called GRITE the data is represented through a graph where nodes represent the objects in the data, and vertices express the precedence relationships derived from the considered attributes.

In [7], the authors propose to calculate the support by using the *Kendall tau ranking correlation coefficient*. This coefficient calculates the number of pairs

of tuples that can be ordered (concordant) or not (discordant) in the database according to the considered gradual pattern. Unfortunately, in all surveyed approaches, reducing the quantity of mined patterns was not the main concern. In the following subsection, we recall some key settings from the FCA framework presenting some pioneering results towards defining a concise representation of gradual patterns.

2.2 Formal Concept Analysis

Formal Concept Analysis (FCA) is based on the mathematical theory of complete lattices (for further information please refer to [3]). This theory provides a powerful mathematical framework that has been used in many fields of computer science. Indeed, this theory has been employed in data mining to extract a representative set of itemsets that can be used to extract association rule in an efficient manner. To do so, a formal context must be defined as a triplet $(\mathcal{R}, \mathcal{O}, \mathcal{I})$ where \mathcal{O} is a set of transactions or objects, \mathcal{I} is a finite set of items and \mathcal{R} is a binary relation $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{I}$. The first step to construct the lattice is to define a Galois connection between two derivation operators: one mapping a set of objects into a set of items and the other one mapping a set of items into a set of objects.

For a set $O \in \mathcal{O}$ and $I \in \mathcal{I}$ the two mapping operators denoted by f and g are defined, respectively, as follows:

- $f : \mathcal{P}(\mathcal{O}) \rightarrow \mathcal{P}(\mathcal{I}), f(O) = \{i \in \mathcal{I} \mid (o, i) \in \mathcal{R}, \forall o \in O\}$
- $g : \mathcal{P}(\mathcal{I}) \rightarrow \mathcal{P}(\mathcal{O}), g(I) = \{o \in \mathcal{O} \mid (o, i) \in \mathcal{R}, \forall i \in I\}$

These two mapping operators f and g induce a Galois connection between the powerset of objects and the powerset of items. This Galois connection means that f and g are dually adjoint, *i.e.*, $O \subseteq g(I) \Rightarrow I \subseteq f(O)$ for a set of objects O and a set of items I . The two composite operators $f \circ g$ and $g \circ f$ are *closure operators* (*i.e.*, they keep the properties of monotonicity, extensivity and idempotency). A formal concept is a pair (O, I) of a set of objects $O \in \mathcal{O}$ and a set of items $I \in \mathcal{I}$, where $f(O)=I$ and $g(I)=O$. The set of all concepts that can be extracted from a gradual formal context form a complete *lattice* provided with a partial order relation \leq , such that $\forall c_1=(O_1, I_1)$ and $c_2=(O_2, I_2)$ two concepts, if $c_1 \leq c_2 \Leftrightarrow I_2 \subseteq I_1$ ($O_1 \subseteq O_2$).

3 Defining New Galois Mapping Operators for Gradual Patterns

To the best of our knowledge, no previous study in the literature has paid attention to apply the Galois connection in gradual pattern extraction problem. Given that classical FCA was developed for binary relationships, adapting the former results to the gradual case turns out to be an interesting formalization problem. Within this work we aim at using FCA theory in order to formalize a new closure system characterizing gradual data. A gradual rule is defined as a

special kind of association rule reflecting a variation in the degree of membership of itemsets in a sequence of objects. A gradual rule is formulated as “*The more/less X, the more/less Y*”, where X and Y are gradual patterns. An example of a gradual rule is “*The higher the Age, the higher the Salary*”. In order to satisfy the graduation property, we must consider a set of object sequences. Indeed, in the classical FCA case the domain of an itemset is a set of objects. In our case the domain of a gradual itemset is a set of sequences satisfying this itemset. The set of sequences will be ordered by the properly defined relation \preceq . In what follows, we propose to define the notion of sequence and the related mathematical operations that can be applied over the set of these sequences.

3.1 Handling Object Sequences

Let $\mathcal{O} = \{o_1, \dots, o_n\}$ be a set of objects. We consider a sequence to be an ordered list of objects described over attributes (items). This sequence can be represented as $\langle o_1, \dots, o_m \rangle$. This means that objects are sorted and each object o_i has an order in the sequence.

Definition 2. A sequence $S = \langle o_1, \dots, o_p \rangle$ is **included** in another sequence $S' = \langle o'_1, \dots, o'_m \rangle$, denoted by $S \subseteq S'$, if there exist integers $1 < i_1 < i_2, \dots, < i_p < m$ such that $o_1 = o'_{i_1}, \dots, o_p = o'_{i_p}$.

Definition 3. Let \mathcal{S} be a collection (i.e., a set) of sequences, $S \in \mathcal{S}$ is said to be maximal if $\nexists S' \in \mathcal{S}, S' \neq S$ such that $S \subset S'$.

Definition 4. The **intersection** of two sequences S_1 and S_2 is the set of all maximal subsequences of both S_1 and S_2 :

$$S_1 \cap S_2 = \{s_i | s_i \subseteq S_1, s_i \subseteq S_2 \text{ and } \nexists s_i \subset s'_i \text{ such that } s'_i \subseteq S_1 \text{ and } s'_i \subseteq S_2\}.$$

Definition 5. A set of sequences \mathcal{S} is **included** in another set \mathcal{S}' , denoted by $\mathcal{S} \preceq \mathcal{S}'$, if $\forall S$ in $\mathcal{S}, \exists S' \in \mathcal{S}'$ s.t. $S \subseteq S'$.

Based on the binary inclusion relation of sequences set defined in definition 5, the following proposition holds:

Proposition 1. Let \mathcal{S} be a set of maximal sequences. $\mathcal{P}(\mathcal{S})$ provided with the binary relation \preceq is a partially ordered set (or poset).

3.2 Gradual Galois Connection

In this paper, we propose a new definition of Galois connection taking graduality into account. Hence, we first define the notion of a gradual formal context.

Definition 6. Gradual formal context. A gradual formal context is defined as the quadruplet $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{Q}, R)$ describing a set of objects \mathcal{O} , a finite set \mathcal{I} of attributes (or items), a finite set of quantities or values \mathcal{Q} and a binary relation R (i.e., $R \subseteq \mathcal{O} \times \mathcal{I}$). Each pair $(o, i^q) \in R$, means that the value of the attribute (item) i belonging to \mathcal{I} in the object o belonging to \mathcal{O} is q .

Table 1. Formal gradual context

	Age	Salary	Loan
o_1	22	1200	4
o_2	24	1850	2
o_3	30	2200	3
o_4	28	3400	1

Example 2. An example of a gradual formal context $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{Q}, R)$ is sketched in Table 1. We have $(o_1, Age^{22}, Salary^{1200}) \in \mathcal{R}$.

Let $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{Q}, R)$ be a gradual formal context, we define below the two closure operators f and g :

$$f : \mathcal{P}(\mathcal{S}) \rightarrow \mathcal{P}(\mathcal{I})$$

$$f(S) = \{i^* \mid \forall s \in S, \forall o_l, o_k \in s \text{ s.t. } (o_l, i^{q_1}), (o_k, i^{q_2}) \in \mathcal{R} \text{ and } k < l \text{ we have } q_1 * q_2\}$$

The mapping function f returns all gradual items with their respective variation respecting all sequences in S .

$$g : \mathcal{P}(\mathcal{I}) \rightarrow \mathcal{P}(\mathcal{S})$$

$$g(I) = \{s \in S \mid s \text{ is maximal in } S \text{ and } \forall o_l, o_k \in s \text{ s.t. } k < l \text{ and } (o_l, i^{q_1}), (o_k, i^{q_2}) \in \mathcal{R}, \forall i^* \in I \text{ we have } q_1 * q_2\}$$

The mapping function g returns the set of maximal sequences respecting the variations of all items in I . The two mapping functions g and f are respectively defined over the power set of \mathcal{I} and the power set of sequences of \mathcal{S} . Given that the intersection of a set of object sequences may result more than one sequence, we consider the power set of sequences. The function f is applied on a set of sequences whereas g is applied on a set of gradual attributes. On the one hand, $f(S)$ returns the gradual itemset I , such that every gradual item $i \in I$ is provided with the corresponding variation $*$ through the sequences of S . On the other hand, $g(I)$ looks for all sequences verifying the variation of each item in I . The set of gradual itemsets can be ordered by the standard inclusion binary relation \subseteq . However, the set of sequences is ordered by the binary relation \preceq .

Example 3. Let us consider the context illustrated in Figure 1. Thus, we have for example $f(\langle o_1, o_2, o_4 \rangle, \langle o_1, o_2, o_3 \rangle) = \{Age^{\geq} Salary^{\geq}\}$ and $g(\{Age^{\geq} Loan^{\leq}\}) = \{\langle o_1, o_2, o_3 \rangle, \langle o_1, o_3 \rangle\}$

Based on the definitions and propositions introduced above, we can now demonstrate that we have construct a Galois Connection framework for the gradual case. As stated above, this will allow us to mine for concise representations, thus reducing the size of the results presented to end-users. It should be noted that this reduction is of great importance as users are often drawn in resulting patterns.

Proposition 2. For sets of sequences S and $S' \in \mathcal{S}$, and sets of gradual itemsets I and I' the following properties hold:

- 1) $S \preceq S' \Rightarrow f(S') \subseteq f(S)$
- 1') $I \subseteq I' \Rightarrow g(I') \subseteq g(I)$
- 2) $S \preceq g(f(S))$
- 2') $I \subseteq f(g(I))$

Proposition 3. The composite operators $f \circ g$ and $g \circ f$ form two closure operators, respectively defined on the sets of sequences and the set of itemsets.

The result of all these propositions define a new framework for gradual closed patterns which are a concise representation of gradual frequent patterns. In fact, let us consider the definitions given below:

Definition 7. Gradual formal concept The pair (S, I) , such that $S \in \mathcal{S}$ and $I \in \mathcal{I}$, is called a gradual concept if $f(S) = I$ and $g(I) = S$. O is called the extension and I the intension of the gradual concept.

Definition 8. Gradual closed itemset Let us consider the formal context $\mathcal{K} = (O, \mathcal{I}, \mathcal{Q}, R)$, a gradual subset $I \subseteq \mathcal{I}$, I is called gradual closed itemset if and only if it is equal to its closure, i.e., $f \circ g(I) = I$.

Definition 9. Minimal gradual generator A gradual itemset $h \subseteq \mathcal{I}$ is called minimal gradual generator of another gradual closed itemset I if $f \circ g(h) = I$ and does not exist $h' \subseteq \mathcal{I}$ such that $h' \subset h$. The set \mathcal{GGM} of all gradual minimal generators of a gradual closed itemset I is defined as bellow:
 $\mathcal{GGM} = \{h \subseteq \mathcal{I} \mid f \circ g(h) = I \wedge \nexists h' \subset h \text{ tel que } f \circ g(h') = I\}$

Proposition 4. A set of gradual formal concepts $\mathcal{GC}_{\mathcal{K}}$ extracted from a formal context \mathcal{K} , ordered using the set inclusion relation, form a complete lattice $\mathcal{L}_{\mathcal{K}} = (\mathcal{GC}_{\mathcal{K}}, \subseteq)$, called gradual Galois lattice.

4 Experiments

This section validates the interest of our approach. In fact, the number of gradual closed patterns that can be extracted from a dataset is much smaller than the total number of frequent gradual patterns. In the other hand, the set of gradual closed patterns and their respective gradual generators can be used to define an irreducible compact nuclei (i.e., generic basis) of gradual rules (this point will be discussed with more details in a further paper). To rate the importance of our approach, we ran experiments on synthetic datasets. These datasets were generated by a modified version of IBM Synthetic Data Generation Code for Associations and Sequential Patterns¹. Let us note that these datasets are very dense and, even for a high value of the minimal support, a huge number of gradual patterns can be extracted. In most of the cases, techniques allowing to obtain gradual knowledge are generally driven on bases containing a weak number of

¹ www.almaden.ibm.com/software/projects/hdb/resources.shtml

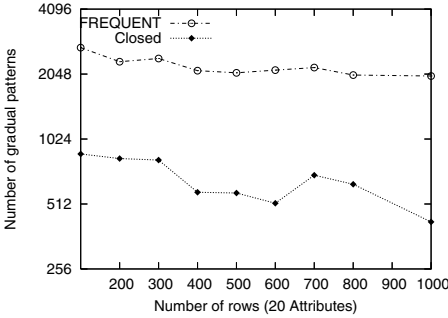


Fig. 1. Gradual closed vs. gradual frequent patterns with the variation of objects number

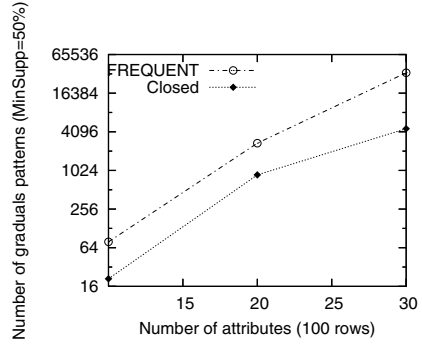


Fig. 2. Gradual closed patterns vs gradual frequent patterns with the variation of attributes number

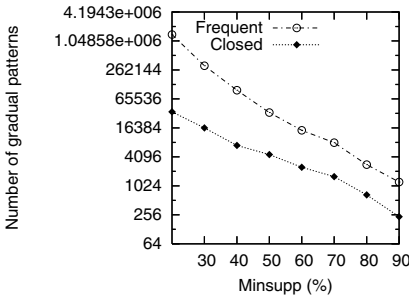


Fig. 3. Number of gradual closed patterns vs frequent patterns with minsup variation

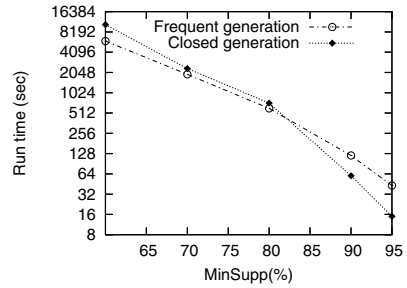


Fig. 4. Evolution of the computation time for discovering gradual closed and frequent patterns vs the variation of the minsup value

objects and attributes. In our experiments, we focus on the variation of gradual closed patterns with regard to frequent ones according to the minimal support (minsup) value, and the variation of the attributes and object number. Figure 1 shows, for 20 attributes and a minsup equal to 0.5, the number of closed patterns and frequent patterns according to the number of lines. The reported values depicted in this figure show that both closed and frequent gradual patterns are varying with the number of lines in a linear manner. We have to note that we have a logarithmic scalability and the difference is very large. However this number varies in an exponential manner with the number of attributes and with the *minsup* values as, respectively, shown in Figures 2 and 3 for a dataset of 100 lines and 40 attributes. In this paper, we aim at showing the importance reduction of the number of the extracted closed patterns with comparison with frequent ones. As mentioned above, our approach is a *post-treatment* of [6]. Runtimes are thus a little longer as shown in Figure 4.

5 Conclusion and Future Work

In this paper, we propose an approach for mining a concise representation of gradual patterns. We propose a novel closure system in order to extract gradual closed patterns. As it is expected, these gradual closed patterns provide a high rate of compactness regarding the total number of gradual patterns, which is assessed by the experiments we led.

Further works will mainly aim at improving the efficiency of our algorithms regarding runtimes, by embedding our method within the existing approaches. We will also consider the definition of a cover of the gradual association rules. This task is of paramount importance since it will allow to present to end-users only a manageable reduced quantity of gradual association rules.

References

1. Berzal, F., Cubero, J., Sánchez, D., Vila, M., Serrano, J.: An alternative approach to discover gradual dependencies. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 15(5), 559–570 (2007)
2. Galichet, S., Dubois, D., Prade, H.: Imprecise specification of ill-known functions using gradual rules. *International Journal of Approximate Reasoning* 35, 205–222 (2004)
3. Ganter, B., Wille, R.: *Formal Concept Analysis*. Springer, Heidelberg (1999)
4. Hüllermeier, E.: Association rules for expressing gradual dependencies. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) *PKDD 2002*. LNCS (LNAI), vol. 2431, pp. 200–211. Springer, Heidelberg (2002)
5. Jorio, L.D., Laurent, A., Teisseire, M.: Fast extraction of gradual association rules: a heuristic based method. In: *Proceedings of the 5th international conference on Soft computing as transdisciplinary science and technology (CSTST 2008)*, Cergy-Pontoise, France, pp. 205–210 (2008)
6. Jorio, L.D., Laurent, A., Teisseire, M.: Mining frequent gradual itemsets from large databases. In: Adams, N.M., Robardet, C., Siebes, A., Boulicaut, J.-F. (eds.) *IDA 2009*. LNCS, vol. 5772, pp. 297–308. Springer, Heidelberg (2009)
7. Laurent, A., Lesot, M.-J., Rifqi, M.: Graank: Exploiting rank correlations for extracting gradual dependencies. In: *Proc. of FQAS 2009* (2009)
8. Molina, C., Serrano, J., Sánchez, D., Vila, M.: Measuring variation strength in gradual dependencies. In: *Proceedings of the International Conference EUSFLAT 2007*, Ostrava, Czech Republic, pp. 337–344 (2007)

New Method for Generation Type-2 Fuzzy Partition for FDT

Lukasz Bartczuk¹, Piotr Dziwiński¹, and Janusz T. Starczewski^{1,2,*}

¹ Department of Computer Engineering, Czestochowa University of Technology,
Al. Armii Krajowej 36, 42-200 Czestochowa, Poland

{lukasz.bartczuk,piotr.dziwinski,janusz.starczewski}@kik.pcz.pl

² Academy of Management (SWSPiZ), Institute of Information Technology,
ul. Sienkiewicza 9, 90-113 Łódź, Poland

Abstract. One of the most important tasks during application of fuzzy decision tree algorithms is to generate a fuzzy partition. In this paper, we introduce a new method to perform this task. The proposed method is a two stage process. The second stage is based on the classical Fuzzy C-means (FCM) clustering.

1 Introduction

Decision trees are common methods in data mining and machine learning. They are appreciated because of their clarity and interpretability as well as for the attribute selection mechanism. The most popular algorithms to create such structures are CART [1] and ID3 [2] or its successor C4.5 [3].

Fuzzy decision trees (FDTs) of type-1 [4] or of type-2 [5] are generalized versions of classical decision trees and combine their advantages with possibilities of modelling an uncertain or imprecise knowledge. These algorithms require that each continuous attribute have to be previously split into a number of fuzzy intervals.

In this paper, a new method to generate type-2 fuzzy partition is presented. The method takes into account an inner uncertainty that is contained in a modeled process. The paper is organized as follows: the next section briefly describes the type-2 fuzzy decision trees, section 3. depicts a proposed method to obtain type-2 fuzzy sets, section 4 illustrates experimental results, and the last section draws final conclusions.

2 Interval Type-2 Fuzzy Decision Trees

The interval type-2 fuzzy decision trees [5,6] are created by a recursive procedure from data set E and attribute set A . Every pattern contained in the data set is a

* This work was partly supported by Polish Ministry of Science and Higher Education (Habilitation Project N N516 372234 2008–2011, Polish-Singapore Research Project 2008–2010).

vector $\mathbf{x}_t = [x_1(t), \dots, x_{|A|}(t), y(t)]$ consisting both input values and a value of decision attribute. Every attribute $A^k \in A, k = 1, \dots, |A|$, contains $|A^k|$ different values a_m^k and describes some characteristic feature of objects from the data set. All values included in the attribute set are characterized by some interval type-2 fuzzy sets.

Let us remind that the interval fuzzy set of type-2 [7,8,9] defined in $X \subset \mathbb{R}$, is an ordered collection of pairs $\{x, \mu_{\tilde{A}}(x)\}$ where $x \in X$, and $\mu_{\tilde{A}}(x)$ is a fuzzy grade of membership defined on $[0, 1]$ interval as $\mu_{\tilde{A}}(x) = \int_{u \in [0,1]} 1/u$, where u is called a primary membership grade. To simplify the notation, the interval fuzzy membership grade can be represented as minimum and maximum points of its support: $\mu_{\tilde{A}}(x) = [\underline{\mu}_{\tilde{A}}(x), \overline{\mu}_{\tilde{A}}(x)]$.

In each step of recursion for the decision tree creating procedure, the data set is split, according to a chosen attribute A^k , into S fuzzy subsets, where $2 \leq S \leq |A^k|$. The choice of the attribute is performed by an algorithm based on the assumption that resulting data subsets have a higher degree of homogeneity than a complete data set. The detailed description of this algorithm can be found in paper [5].

2.1 Inference Process

Every decision tree can be considered as a set of decision rules, whose number is equal to the number of leaves in the tree. Obviously in case of the interval type-2 fuzzy decision tree, the type-2 fuzzy decision rule have to be under consideration, and can be presented in the following form:

$$R^o : \text{IF } \bigcap_{\tilde{a}_m^k \in \mathcal{P}^o} (x^k \text{ IS } \tilde{a}_m^k) \text{ THEN } y \text{ IS } \tilde{a}_o^D; \quad o = 1, \dots, |O|, \quad (1)$$

where $|O|$ denotes a number of leaves (rules), \tilde{a}_o^D — a value of the decision attribute assigned to leaf o , \mathcal{P}^o — a path from a root node to leaf o . A premise part of this rule is constituted by values of attributes, which can be found on path \mathcal{P}^o , for which this rule has been generated. A conclusion part contains the value of decision attribute \tilde{a}_o^D that have been assigned to this leaf by the procedure creating a tree structure.

At the beginning of the process of classification previously unseen patterns, an activation grade for each fuzzy decision rule has to be computed as the t -norm combination of all type-2 fuzzy sets from the premise part of rule (1):

$$\mu_{A^o}(\mathbf{x}) = \tilde{\text{T}}_{\tilde{a}_m^k \in \mathcal{P}^o} \mu_{\tilde{a}_m^k}(x_k) = \left[\text{T}_{\tilde{a}_m^k \in \mathcal{P}^o} \underline{\mu}_{\tilde{a}_m^k}(x_k), \text{T}_{\tilde{a}_m^k \in \mathcal{P}^o} \overline{\mu}_{\tilde{a}_m^k}(x_k) \right], \quad (2)$$

where: $\mu_{A^o}(\mathbf{x})$ is the activation grade of the rule o .

If we also assume that all values of the decision attribute are singletons, the activation grade and the conclusion grade are equal, so that:

$$\underline{\tau} = \mu_{A^o}(\mathbf{x}) * \underline{\mu}_{\tilde{a}_o^D}(y) = \underline{\mu}_{A^o}(\mathbf{x}), \quad (3)$$

$$\overline{\tau} = \overline{\mu}_{A^o}(\mathbf{x}) * \overline{\mu}_{\tilde{a}_o^D}(y) = \overline{\mu}_{A^o}(\mathbf{x}). \quad (4)$$

When we determine activated rules, we can take a decision, into which value of the decision attribute the pattern have to be assigned. In the simplest case, we can find the rule with the highest activation grade. However, a single pattern can activate more than one rule, possibly with the same consequent. Thus better results may be obtained if we use a total activation grade, that is defined for each value of the decision attribute as follows:

$$\mu_{\tilde{a}_m^D}(\mathbf{x}) = \underset{o:\mu_{\tilde{a}_o^D}=\mu_{\tilde{a}_m^D}}{\tilde{\mathbf{S}}} \mu_{A^o}(\mathbf{x}) = \left[\underset{o:\mu_{\tilde{a}_o^D}=\mu_{\tilde{a}_m^D}}{\mathbf{S}} \frac{\mu_{A^o}(\mathbf{x})}{\mu_{A^o}(\mathbf{x})}, \underset{o:\mu_{\tilde{a}_o^D}=\mu_{\tilde{a}_m^D}}{\mathbf{S}} \overline{\mu_{A^o}(\mathbf{x})} \right]. \quad (5)$$

In both considered cases, we have to compare interval values. This can be performed by one of the following method: MAX, AVG, MIN-MAX, and comparison of fuzzy intervals (CFI).

The MAX method searches for an interval value with the highest upper bound. The AVG method is very similar but it uses averages of intervals. The MIN-MAX method identifies intervals with the highest upper and lower bounds, separately. If both indicate the same interval, the corresponding value is chosen. If two different intervals are found, then the result is undetermined. The last method is based on an algorithm comparing fuzzy numbers, proposed in [10]. This algorithm determines a degree from $[0, 1]$ of satisfying an inequality between two fuzzy quantities A_1 and A_2 . The value 1 for this degree indicates a full certainty that A_1 is greater than A_2 , 0 indicates $A_1 < A_2$, and 0.5 is reserved for equal fuzzy quantities. In the simulations, we assume the correct classification if the fuzzy interval firing degree of a rule satisfies the inequality with all remaining rules with the inequality degree greater than 0.6. Otherwise, the system cannot make any certain decision.

3 Generation of Type-2 Fuzzy Partition

In this section, we would like to propose a new method for generating type-2 fuzzy partition. This is a two-phase method, in which the first phase is responsible for determining the parameters of type-1 fuzzy membership functions, and the second phase creates type-2 fuzzy membership functions.

3.1 Phase 1: Determining Parameters of Type-1 Fuzzy Membership Functions

In this phase, the parameters of type-1 fuzzy membership function are determined. For the purposes of this paper we use the asymmetric Gaussian membership function, which is defined by following formula:

$$a_gauss(x; m, \sigma_l, \sigma_r) = \begin{cases} \exp \left[-\left(\frac{x-m}{\sigma_l} \right)^2 \right] & \text{for } x < m, \\ \exp \left[-\left(\frac{x-m}{\sigma_r} \right)^2 \right] & \text{otherwise,} \end{cases} \quad (6)$$

where m, σ_l, σ_r are respectively: the center of the function, its left and right spreads.

We assume that this phase is performed using the FCM (Fuzzy C-Means) clustering algorithm [11]. Let the number of cluster centers obtained as a result of the FCM algorithm be denoted by C . Each cluster center is defined as a vector $\mathbf{v}_c = [m_1^c, \dots, m_{|A|}^c]$. The value m_c^k is also the center of the c^{th} membership function defined for the k^{th} attribute.

The left and right spreads of the asymmetrical Gaussian function can be computed independently for $E_{l_n}^k = \{x^k(t) : x_n^k(t) \leq m_n^k\}$ and $E_{r_n}^k = \{x^k(t) : m_n^k < x_n^k(t)\}$ with the following formula:

$$\sigma_{\delta_n}^k = \sqrt{\frac{\sum_{t=1}^{|E_{\delta_n}^k|} u_{kt} [x^k(t) - m_n^k]^2}{\sum_{t=1}^{|E_{\delta_n}^k|} u_{kt}}}, \quad (7)$$

where: $\delta = \{l, r\}$ signifies the left or right spread, u_{kt} is a fuzzy membership value computed by the FCM method.

3.2 Phase 2: Creating Type-2 Membership Function

The actual degree of membership assigned by the FCM algorithm and computed on the base of the membership functions specified in the previous phase will vary for real-world data. If we assume that training data were collected without a measurement error, this difference between membership grades is caused by an inner uncertainty of a modeled process. Therefore, we can expand upper and lower membership functions over training data, such that memberships of all data points are covered by the footprint of uncertainty of a type-2 fuzzy membership function, i.e. the area between the upper and the lower membership function. Consequently, the upper membership function can be a normal asymmetrical Gaussian function,

$$\mu_{A_n^k}(x_n) = a_gauss(x_n, m_n^k, \overline{\sigma}_{l_n}^k, \overline{\sigma}_{r_n}^k)$$

being a superior limit of the function family drawn through points $(x_n(t), u_{kt})$, i.e.,

$$\overline{\sigma}_{l_n}^k = \max_t \sigma_{l_t} : \{a_gauss(x_n(t), m_n^k, \sigma_{l_t}, \sigma_r) \mid x_n(t) \leq m_n^k\} \quad (8)$$

$$\overline{\sigma}_{l_n}^k = \max_t \frac{m_n^k - x_n(t)}{\sqrt{-\log u_{kt}}} \quad (9)$$

and

$$\overline{\sigma}_{r_n}^k = \max_t \sigma_{r_t} : \{a_gauss(x_n(t), m_n^k, \sigma_l, \sigma_{r_t}) \mid m_n^k < x_n(t)\} \quad (10)$$

$$\overline{\sigma}_{r_n}^k = \max_t \frac{x_n(t) - m_n^k}{\sqrt{-\log u_{kt}}} \quad (11)$$

The lower membership function can be a scaled (by h) asymmetrical Gaussian function,

$$\underline{\mu}_{A_n^k}(x_n) = \underline{h}_n^k a_gauss(x_n(t); m_n^k, \sigma_{l_n}^k, \sigma_{r_n}^k) \tag{12}$$

being an inferior limits of the function family drawn through points $(x_n(t), u_{kt})$, i.e.,

$$\underline{h}_n^k = \min_t h_t : \{h_t a_gauss(x_n(t); m_n^k, \sigma_{l_n}^k, \sigma_{r_n}^k)\} \tag{13}$$

$$\underline{h}_n^k = \min_t \frac{u_{kt}}{a_gauss(x_n(t); m_n^k, \sigma_{l_n}^k, \sigma_{r_n}^k)} \tag{14}$$

4 Experiments

To illustrate the capability of the proposed method, the Wisconsin Breast Cancer Dataset from UCI Machine Learning Repository [12] has been used. This dataset contains description of 699 cases from two classes. Benign class is assigned to 458 patterns and Malignant class to 241 patterns. Each pattern is characterized by 10 numerical attributes. The complete dataset consists 16 instances with missing values which have been removed.

The whole dataset has been split into two subsets used for training and testing. The training set has consisted of 547 and testing set — 136 patterns chosen randomly. The training set has been used to generate type-2 fuzzy partition as well as to build the decision tree. The testing set has been applied to determine the classification rates of the obtained decision tree.

The experiments have been also performed for a type-2 fuzzy partition method described in paper [6]. The results are summarized in Tables 1-2.

As we can see the interval fuzzy decision trees of type-2 constructed with fuzzy sets generated by the algorithm based on FCM have lower incorrect classification rates for the MIN-MAX and CFI inference methods than rates for the method based on uncertain fuzzy clustering. Unfortunately, these structures are also characterized by lower rates of correct classification. However we believe that

Table 1. Results obtained for the proposed method

	MAX	AVG	MIN-MAX	CFI
correct classification	0.93	0.96	0.84	0.84
incorrect classification	0.07	0.04	0.01	0.01
undecided	-	-	0.15	0.15

Table 2. Results obtained for the method described in paper [6]

	MAX	AVG	MIN-MAX	CFI
correct classification	0.96	0.96	0.96	0.96
incorrect classification	0.04	0.04	0.04	0.04
undecided	-	-	0	0

in case of medical diagnosis problems, it is better that system cannot give the answer than it gives the wrong answer.

5 Conclusion

In this paper, the new method for generation of type-2 fuzzy partition have been proposed. The method has been thought as a two stage process. In the first stage parameters for each asymmetrical Gaussian membership function have been determined. The second stage has generated type-2 fuzzy membership functions. The source of these interval type-2 membership functions has been regarded as an inner uncertainty contained in a modeled process. The obtained fuzzy partitions have been used to construct type-2 interval fuzzy decision trees.

References

1. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and regression trees. Wadsworth, Belmont (1984)
2. Quinlan, J.: Induction of decision trees. *Machine Learning* 1, 81–106 (1986)
3. Quinlan, J.: C4.5 Programs for Machine Learning. Morgan Kaufmann, San Francisco (1993)
4. Janikow, C.: Fuzzy decision trees: Issues and methods. *IEEE Trans Systems, Man, Cybernetics - Part B: Cybernetics* 28, 1–14 (1998)
5. Bartczuk, L., Rutkowska, D.: Fuzzy decision trees of type-2. In: *Some Aspects of Computer Science*. EXIT Academic Publishing House, Warsaw (2007) (in Polish)
6. Bartczuk, L., Rutkowska, D.: Type-2 fuzzy decision trees. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008*. LNCS (LNAI), vol. 5097, pp. 197–206. Springer, Heidelberg (2008)
7. Castillo, O., Melin, P.: Intelligent systems with interval type-2 fuzzy logic. *International Journal of Innovative Computing, Information and Control* 4(4), 771–783 (2008)
8. Liang, Q., Mendel, J.: Interval type-2 fuzzy logic systems: Theory and design. *IEEE Transactions on Fuzzy Systems* 8, 535–550 (2000)
9. Mendel, J., John, R.: Interval type-2 fuzzy logic systems made simple. *IEEE Transactions on Fuzzy Systems* 10, 622–639 (2002)
10. Dorohonceanu, B.: Comparing fuzzy numbers. *Dr. Dobb's Journal* 343, 38–45 (2002)
11. Bezdek, C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York (1981)
12. UCI, M.L.R.: <http://mllearn.ics.uci.edu/mlrepository.html>

Performance of Ontology-Based Semantic Similarities in Clustering

Montserrat Batet¹, Aida Valls¹, and Karina Gibert²

¹ Department of Computer Science and Mathematics
Universitat Rovira i Virgili, Av. Països Catalans 26, 43007 Tarragona, Spain
{montserrat.batet,aida.valls}@urv.cat

² Department of Statistics and Operations Research
Universitat Politècnica de Catalunya
Campus Nord, Ed.C5, c/Jordi Girona 1-3, E-08034 Barcelona, Spain
karina.gibert@upc.edu

Abstract. Clustering was usually applied on numerical and categorical information. However, textual information is acquiring an increasing importance with the appearance of methods for textual data mining. This paper proposes the use of classical clustering algorithms with a mixed function that combines numerical, categorical and semantic features. The content of the semantic features is extracted from textual data. This paper analyses and compares the behavior of different existing semantic similarity functions that use WordNet as background ontology. The different partitions obtained with the clustering algorithm are compared to human classifications in order to see which one approximates better the human reasoning. Moreover, the interpretability of the obtained clusters is discussed. The results show that those similarity measures that provide better results when compared using a standard benchmark also provide better and more interpretable partitions.

Keywords: Clustering, semantic similarity, ontologies.

1 Introduction

Clustering plays an important role in data mining. It is widely used for partitioning data into a certain number of homogeneous groups or clusters [1]. Traditionally, this data mining technique has been applied to numerical and categorical values. However, nowadays textual data mining is attracting more and more attention in order to exploit the information available in electronic texts or in the Web [2]. New variables, denoted as *semantic features*, can be used to describe the objects. The value of a semantic feature is a linguistic term that can be semantically interpreted using some additional knowledge, such as an ontology.

Developing clustering techniques for heterogeneous data bases including numerical and categorical features together with features representing conceptual descriptions of the objects is a new field of study.

According to this statement, we have designed and implemented a clustering method that can deal with numerical, categorical and semantic features to generate a hierarchical classification of a set of objects. The method calculates the contribution of each type of feature independently, depending on each type of value, and then the partial similarities are aggregated into a unique value.

In a previous paper, we compared the interpretability and quality of the clusters obtained if the semantic features were treated as categorical ones (where each word was treated as a simple modality) with respect to considering their meaning using a semantic similarity measure. The conclusion was that the consideration of the semantics of the concepts in those features improves the quality of the clustering because the clusters have a clearer conceptual interpretation [3].

However, in the area of computational linguistics, many approaches have been proposed to compute the semantic similarity between two concepts. Usually, this similarity computation is based on the estimation of the semantic evidence observed in some additional knowledge structure, in general an ontology [4,5]. The performance of these semantic similarity proposals has been evaluated in different studies [6,7,8] by comparing human evaluations of the similarity with the computerized results in a given set of word pairs [9].

Our hypothesis is that those similarity measures that provide best results comparing pairs of terms in a standard benchmark will provide more accurate clusters when they are used to compute similarities inside a clustering method.

In this paper, we will study the performance of ontology-based similarity measures when they are used in the classical Ward's clustering algorithm [10]. In our experiments, the ontology for assessing the semantic similarity is WordNet [11], which is a lexical database of English, where words are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual, taxonomical and lexical relations.

This paper is organized as follows: In section 2, a general view of how to compute semantic similarity is presented. Section 3 is focused on semantic similarity functions based on the exploitation of the taxonomic structure of the ontology. The distance function that integrates numerical, categorical and semantic features is presented in section 4. In section 5, an evaluation of the clustering results using different semantic similarities is done in order to prove our hypothesis. The last section gives the conclusions of this paper and outlines the future work.

2 Measuring Semantic Similarity

The methods for computing the semantic similarity between a pair of terms can be divided according to the type of knowledge exploited to perform the assessment. Some trends consider the taxonomical structure of the ontology and the distribution of terms in a corpus. These measures are based on the *Information Content* (IC) of a concept, which measures the amount of information provided by a given term from its probability of appearance in a corpus. This approach assumes that infrequent words are more informative than frequent ones. Based on this idea, Resnik [6] proposed an approach in which the computation of the

similarity between a pair of concepts was measured as the amount of information they share in common, which was the IC of their *Least Common Subsumer* (LCS) in a given taxonomy. Several extensions to Resnik’s work have been proposed [7][8]. These IC-based measures can be affected by the availability of a background corpus and their coverage with respect to the evaluated terms [8].

Other methods exploit the subsumption hierarchy (taxonomy) of concepts of an ontology [12][4][5][13]. These measures compute the semantic similarity between concepts as the path length (i.e the number of links) between concepts in the taxonomy. The main advantage of this kind of measures is that they only use an ontology as background knowledge, and, no corpus with domain data is needed. In this paper, we have centered our study on this type of measures.

3 Semantic Similarity Measures Based on the Taxonomy

The simplest way to measure the similarity between a pair of concepts c_1 and c_2 considers the minimum *path length* between these concepts, computed as the minimum number of links connecting them (Rada [12]) in the taxonomy (1).

$$pL(c_1, c_2) = \min \# \text{ of is-a edges connecting } c_1 \text{ and } c_2 \quad (1)$$

Some variations of this measure have been developed. Wu and Palmer [4] proposed a measure based on the depth of concepts in the ontology hierarchy. In Eq. 2, N_1 and N_2 are the number of is-a links from c_1 and c_2 to the *LCS* of both terms, and N_3 is the number of is-a links from the *LCS* to the root (2).

$$sim_{w\&p}(c_1, c_2) = \frac{2 * N_3}{N_1 + N_2 + 2 * N_3} \quad (2)$$

Otherwise, Leacock and Chodorow [5] proposed a measure that depends on the number of nodes N_p from c_1 to c_2 and the depth D of the taxonomy (3).

$$sim_{l\&c}(c_1, c_2) = -\log N_p / 2D \quad (3)$$

Notice that those path length-based measures only consider the minimum path between a pair of concepts, omitting the rest of the taxonomical knowledge available in the ontology. For complex taxonomies with thousands of interrelated concepts in multiple hierarchies this kind of measures wastes a great amount of knowledge.

A measure that considers the whole taxonomical hierarchy involving the evaluated concepts was defined in [13]: the Super-Concept based Distance (SCD). This measure is based on the ratio between non-common superconcepts (upper concepts of a concept) and common superconcepts of a pair of concepts, where $A(c_i)$ is the set of ancestors or superconcepts of c_i in the taxonomy, including itself (eq. 4).

$$sim(c_1, c_2) = \sqrt{\frac{|A(c_1) \cup A(c_2)| - |A(c_1) \cap A(c_2)|}{|A(c_1) \cup A(c_2)|}} \quad (4)$$

4 Combining Numerical, Categorical and Semantic Features

In clustering, the input is usually a data matrix in which the set of individuals $\mathcal{I} = \{1, \dots, n\}$ are in rows and the K features $X_1 \dots X_K$ that describe the individuals are in columns. Cells contain the value (x_{ik}) taken by individual $i \in \mathcal{I}$ for feature X_k .

The combination of numerical, categorical and semantic features is done by means of a family of measures where the distance between i and i' is computed by applying an specific distance for each type of feature X_k , where $\zeta = \{k : X_k \text{ is a numerical feature, } k = 1 : K\}$, $\mathcal{Q} = \{k : X_k \text{ is a categorical feature, } k = 1 : K\}$, and $\mathcal{S} = \{k : X_k \text{ is a semantic feature, } k = 1 : K\}$. Then, the distance is defined in (5) as a mixed distance function between individuals i and i' as:

$$d_{(\alpha,\beta,\gamma)}^2(i, i') = \alpha d_{\zeta}^2(i, i') + \beta d_{\mathcal{Q}}^2(i, i') + \gamma d_{\mathcal{S}}^2(i, i'), (\alpha, \beta, \gamma) \in [0, 1]^3, \alpha + \beta + \gamma = 1 \tag{5}$$

In our tests, $d_{\zeta}^2(i, i')$ is the normalized Euclidean distance for numerical features, $d_{\mathcal{Q}}^2(i, i')$ is the χ^2 metrics for categorical values and $d_{\mathcal{S}}^2(i, i')$ is a semantic similarity measure for semantic features in \mathcal{S} (see section 3). Therefore:

$$d_{(\alpha,\beta,\gamma)}^2(i, i') = \alpha \sum_{k \in \zeta} \frac{(x_{ik} - x_{i'k})^2}{s_k^2} + \frac{\beta}{n_{\mathcal{Q}}} \sum_{k \in \mathcal{Q}} d_k^2(i, i') + \frac{\gamma}{n_{\mathcal{S}}} \sum_{k \in \mathcal{S}} ds_k^2(i, i') \tag{6}$$

where s_k^2 is the variance of the feature X_k , $n_{\mathcal{Q}} = \text{card}(\mathcal{Q})$, $n_{\mathcal{S}} = \text{card}(\mathcal{S})$ and $d_k^2(i, i')$ is the contribution of categorical feature X_k according to the χ^2 definition, and $ds_k^2(i, i')$ is the contribution of semantic feature X_k . In (5) each component is weighted according to the importance of each type of features into the data matrix. The weighting constants (α, β, γ) can be taken as functions of the features characteristics. In particular, we propose to calculate the values for these constants guaranteeing that $(\alpha, \beta, \gamma) \in [0, 1]^3$ and $\alpha + \beta + \gamma = 1$ as:

$$\alpha = \frac{\frac{n_{\zeta}}{d_{\zeta}^2 \text{max}^*}}{\frac{n_{\zeta}}{d_{\zeta}^2 \text{max}^*} + \frac{n_{\mathcal{Q}}}{d_{\mathcal{Q}}^2 \text{max}^*} + \frac{n_{\mathcal{S}}}{d_{\mathcal{S}}^2 \text{max}^*}} \quad \beta = \frac{\frac{n_{\mathcal{Q}}}{d_{\mathcal{Q}}^2 \text{max}^*}}{\frac{n_{\zeta}}{d_{\zeta}^2 \text{max}^*} + \frac{n_{\mathcal{Q}}}{d_{\mathcal{Q}}^2 \text{max}^*} + \frac{n_{\mathcal{S}}}{d_{\mathcal{S}}^2 \text{max}^*}} \quad \gamma = \frac{\frac{n_{\mathcal{S}}}{d_{\mathcal{S}}^2 \text{max}^*}}{\frac{n_{\zeta}}{d_{\zeta}^2 \text{max}^*} + \frac{n_{\mathcal{Q}}}{d_{\mathcal{Q}}^2 \text{max}^*} + \frac{n_{\mathcal{S}}}{d_{\mathcal{S}}^2 \text{max}^*}}$$

where $n_{\zeta} = \text{card}(\zeta)$, $n_{\mathcal{Q}} = \text{card}(\mathcal{Q})$, $n_{\mathcal{S}} = \text{card}(\mathcal{S})$ and $d_{\zeta}^2 \text{max}^*$, $d_{\mathcal{Q}}^2 \text{max}^*$, $d_{\mathcal{S}}^2 \text{max}^*$ are the truncated maximums of the different subdistances.

The calculation of the weighting constants in this manner guarantees (1) that the tree components have the same influence in the calculation of $d^2(i, i')$, because they are proportional to the maximum value they can present; (2) they are robust to outliers because considers truncated maximums; and (3), it is given more importance to those type of features that mainly describe the objects by defining the constants proportionally to the number of features they represent.

5 Evaluation

In this section, the performance of the semantic similarity measures presented in Sect. 3 when used in a clustering process is studied. The partitions obtained

with each different measure are compared with partitions manually done by a group of people in order to study which one produces better results.

The performance of similarity measures is usually evaluated through a benchmark that consists of a list of word pairs ranked by a group of people. The agreement of the computational results and human judgments is assessed by looking at how well the computed ratings correlate with human ratings for the same data set. Resnik [6] evaluated different similarity measures by calculating the correlation between their similarity scores on 28 word pairs with assessments made by human subjects. This experiment replicated a previous analysis done by Miller and Charles [9] by giving a group of experts the same set of noun pairs. Resnik computed how well the rating of the experts in his evaluation correlated with Miller and Charles ratings. The average correlation was 0.884. This value is considered the upper bound to that one could expect from a machine computation on the same task [6]. We have used Resnik's benchmark and WordNet(3.0) to evaluate the measures presented in Sect 3. The correlation of each measure against the average human ratings for each pair of words is given in Table 1.

Table 1. Correlations for path length measures against human judgments

Similarity method	Correlation
Path	0.670
WP	0.804
LC	0.829
SCD	0.839

In Table 1, it can be seen that the Path Length measure offers a limited performance with the lowest correlation value (0.67). Other path length based measures that incorporate more ontological information such as those defined by Wu and Palmer (WP) and Leacock and Chodorow (LC) improve the results considerably (0.804 and 0.829, respectively). The measure that considers the whole subsumer's hierarchy, Super-concept Based Distance (SCD), has the best performance compared against the others, obtaining a significant increase in the correlation value. This indicates that considering both the amount of common and non-common information between a pair of concepts can result in a more accurate estimation of the semantic similarity.

Next, we want to prove that those semantic similarities that obtained the best results in a general benchmark will also have the best results when they are used to generate a partition of a set of objects with a traditional clustering algorithm. An implementation of the Ward's hierarchical clustering method including the presented mixed distance (Sect. 4) has been applied to a dataset of touristic city destinations. The data matrix contains 23 cities from all over the world. Each city is represented with a vector of 9 features extracted from Wikipedia: two numerical features (population and land area), two categorical features (continent and city ranking) and five semantic features (country, language, geographical situation, major city interest and geographical interest). The same WordNet version has been used. The clusters are displayed in Table 2.

Table 2. Clustering results using different semantic similarity functions

Sem. Partition		Cities in each cluster
Sim.		
Path	Class1	Paris, Montreal, Sydney, Los Angeles
	Class2	NY, Washington
	Class3	Caracas, Lleida, Tarragona, Córdoba, P. de Mallorca, Habana, Roma
	Class4	Interlaken, Tignes, Monterosa-Aosta, Andorra la Vella
	Class5	Chamonix, Madrid, Santa Cruz de Tenerife
	Class6	Funchal, Ponta Delgada, Barcelona
WP	Class1	Paris, Washington
	Class2	Montreal, Los Angeles, NY, Sydney
	Class3	Roma, Córdoba, Habana, Caracas
	Class4	S.C.Tenerife, Madrid, P.Mallorca, Funchal, Ponta Delgada, Barcelona
	Class5	Lleida, Tarragona, Andorra la Vella, Interlaken, Chamonix, Tignes, Monterosa-Aosta
LC	Class1	Paris, Washington, Montreal, Los Angeles, NY, Sydney
	Class2	Andorra la Vella, Interlaken, Chamonix, Tignes, Monterosa-Aosta
	Class3	Tarragona, Barcelona, Funchal, Ponta Delgada
	Class4	Roma, Habana, P.Mallorca, Caracas, Lleida, Córdoba, Madrid, S.C. Tenerife
SCD	Class1	Paris, Washington
	Class2	Montreal, Los Angeles, NY, Sydney
	Class3	Andorra la Vella, Interlaken, Chamonix, Tignes, Monterosa-Aosta
	Class4	Habana, Caracas, Roma
	Class5	Lleida, Córdoba, Madrid, Palma de Mallorca, Santa Cruz de Tenerife, Madrid, Tarragona, Barcelona, Funchal, Ponta Delgada

Table 3. Distance between human partitions and computerized partitions

Similarity method	Average d_{Part}
Path	0.560
WP	0.456
LC	0.441
SCD	0.32

Analogous to the usual way of evaluating semantic similarities, the set of cities was evaluated by a group of 4 subjects who partitioned them in different clusters of similar destinations. The clusters obtained were compared against the human classification using the distance between partitions defined in [14]. Table 3 summarizes the distances between the human classifications and the partitions obtained applying the clustering method with different semantic similarity functions. Since the human classifications were not identical, an average distance of each semantic similarity with respect to the 4 human subjects is given.

Table 3 shows that the partition obtained using the Path Length measure is the least accurate with respect to the human evaluations with a distance of 0.56. The WP and LC measures clearly outperform the poor results obtained

with Path Length. In fact, LC offer the best results of those measures based on the path length (0.44). As it happened in the experiment proposed by Resnik, the best results are obtained by SCD, with a distance of 0.32. In conclusion, the results obtained using those measures that have higher correlations using the Resnik's benchmark also have the best performance in the clustering process.

From the point of view of interpretability (Table 2), Path Length provides some classes difficult to understand, putting together Chamonix and Santa Cruz the Tenerife despite they are not the same type of touristic destination. The experiment done using the WP similarity provides more interpretable classes. However, Tarragona and Lleida are included in a cluster with cities with ski resorts, although to ski is not possible in those cities. The experiment done using LC is able to detect correctly the class of skiing cities. However, Class4 in LC is quite heterogeneous. Finally, using the SCD the interpretation of clusters looks more coherent. Cities are grouped as country capitals; state capitals from North America or Australia placed in islands or near the coast where the spoken language is English; European cities placed at mountains where the main attraction is ski; country capitals from Latin cultures with religious architecture; and Spanish and Portuguese cities, not located in mountains, speaking romance languages, with religious monuments or touristic attractions.

6 Conclusions

In this paper we have studied the influence of using different semantic similarity measures in the results of clustering. Considering the wide range of available semantic similarity approaches, we have centred our study on those measures that are based on the taxonomical exploitation of the ontology. These semantic similarity measures can be integrated into a mixed distance that is capable of combining the contribution of a set of numerical, categorical and semantically interpretable features to compute the distance between individuals in a clustering process. The case of touristic city destinations has been considered.

The evaluation of the presented semantic similarities using a standard benchmark, and the evaluation of the obtained partitions using those similarities with respect to human judgements, have shown that those similarities that correlate better with human ratings in a standard benchmark, also provide more accurate and refined clusters. This is an interesting result because it indicates that simple tests based on correlations between pairs of words can be performed to evaluate the similarity measures before incorporating them into a more complex and time-consuming clustering algorithm. In addition, the inclusion of semantic knowledge into the clustering process leads to more interpretable partitions with stronger semantic relations between the elements of the same cluster.

In our future work, we plan to evaluate this semantic similarity measures using domain ontologies specifically built to describe a particular domain, such as tourism or medicine. In addition, other types of semantic similarities could also be evaluated, such as the ones based on the information content.

Acknowledgments. This work is supported by the DAMASK Spanish project (Data mining algorithms with semantic knowledge, TIN2009-11005), the Spanish Government (PlanE) and the ARES project(CSD2007-00004). Montserrat Batet has a research grant provided by Universitat Rovira i Virgili.

References

1. Xu, R., Wunsch, D.I.: Survey of clustering algorithms. *IEEE Transactions on Neural Networks* 16(3), 645–678 (2005)
2. Hotho, A., Maedche, A., Staab, S.: Ontology-based text document clustering. *Künstliche Intelligenz* 4, 48–54 (2002)
3. Batet, M., Valls, A., Gibert, K.: Improving classical clustering with ontologies. In: *Proceedings of the 4th World conference of the IASC, Japan*, pp. 137–146 (2008)
4. Wu, Z., Palmer, M.: Verb semantics and lexical selection. In: *Proc. 32nd annual Meeting of the Association for Computational Linguistics, USA*, pp. 133–138 (1994)
5. Leacock, C., Chodorow, M.: Combining local context and wordnet similarity for word sense identification. In: *Fellbaum, C. (ed.) WordNet: An electronic lexical database*, pp. 265–283. MIT Press, Cambridge (1998)
6. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: *Proc. of 14th Int’l Joint Conf. on Artificial Intelligence*, pp. 448–453 (1995)
7. Lin, D.: An information-theoretic definition of similarity. In: *Proc. of the 15th International Conference on Machine Learning, Madison, USA*, pp. 296–304 (1998)
8. Jiang, J., Conrath, D.: Semantic similarity based on corpus statistics and lexical taxonomy. In: *Int’l Conf. on Research in Computational Linguistics*, pp. 19–33 (1997)
9. Miller, G., Charles, W.: Contextual correlates of semantic similarity. *Language and Cognitive Processes* 6(1), 1–28 (1991)
10. Ward, J.H.: Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association* 58, 236–244 (1963)
11. Fellbaum, C.: *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge (1998) More information, <http://www.cogsci.princeton.edu/~wn/>
12. Rada, R., Mili, H., Bichnell, E., Blettner, M.: Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man and Cybernetics* 9(1), 17–30 (1989)
13. Batet, M., Sánchez, D., Valls, A., Gibert, K.: Ontology-based semantic similarity in the biomedical domain. In: *12th Conf. on Artificial Intelligence in Medicine. WS. Intelligent Data Analysis in Biomedicine and Pharmacology, Italy*, pp. 41–46 (2009)
14. López de Mántaras, R.: A distance-based attribute selection measure for decision tree induction. *Machine learning* 6, 81–92 (1991)

Information Theory vs. Correlation Based Feature Ranking Methods in Application to Metallurgical Problem Solving

Marcin Blachnik¹, Adam Bukowiec¹, Mirosław Kordos², and Jacek Biesiada¹

¹ Silesian University of Technology, Electrotechnology Department,
Katowice, Krasinskiego 8, Poland

² University of Bielsko-Biała, Department of Mathematics and Computer Science,
Bielsko-Biała, Willowa 2, Poland

Abstract. Feature selection is a typical stage of building any classification or regression model. There are several approaches to it, however one of the fastest is based on determining the relevance of each feature independently by calculating ranking values. In this paper we provide empirical comparison of four different ranking criteria that belong to two different groups *information theory* and *correlation metrics*. The comparison is performed on the empirical datasets obtained while building a model used for predicting mass of chemical compounds necessary to obtain steel of predefined quality.

1 Introduction

One of the most popular metallurgical processes of steel production is based on melting steel scraps in an electric arc furnace (EAF) [1]. This process can be divided into three stages. At the first stage steel scraps are melted in an EAF. At the second, which takes place in the ladle arc furnace (LHF), other elements are added in order to fine-tune the composition of the steel to meet the specification and the customers requirements. The last stage is casting. Aluminum, silicon and manganese are the most common deoxidizers used at the LHF stage. Therefore during the tapping some elements are added into the metal stream casting from the EAF to LHF furnace, others are placed on the bottom of the ladle, and the rest of compounds is added during the LHF procedure.

In the LHF process one of the challenging problems is the prediction of the amount of compounds (aluminum, silicon, manganese and others) that are necessary to meet steel specification. In practice the refining process is often suspended while verifying chemical properties of the steel. This part of that process is very expensive requiring turning on and off the furnace and the arc. Therefore reducing the number of chemical tests may decrease the costs by reducing time and energy consumption. Another goal for steel making engineers is the reduction of the amount of steel add-ins which often are very expensive.

Currently this problem is solved by some simple theoretically defined linear equation, and the experience of the engineer controlling that process. The existing solution calculates the amount of chemical compounds just considering the information of the given steel specification. In practice these parameters depend on melting time of the

EAF process (longer melting determines higher reduction of chemical elements), time of melting during LHF process, the steel specification, steel scrap types used for melting, etc. To consider all these parameters a new model has to be built. Our preliminary estimation showed that linear model is not suitable for that problem, so a more advanced nonlinear model had to be considered. Our choice was one of the most popular and widely used Support Vector Regression (SVR) algorithm, with gaussian kernel functions. Another important stage of data mining process is feature selection which may simplify the model, and often increase model accuracy. In our experiments we decided to test the quality of different ranking based feature selection methods. It is a very simple and fast feature selection approach. The choice of this method is motivated by the results of NIPS'2003 Feature Selection Challenge [5] where one of the best results were obtained using the mentioned ranking based approach. However we face the problem of selecting the appropriate ranking coefficient. For that purpose we have compared two families of ranking coefficients based on information theory - two metrics such as *Information Gain* (IGR-index), and *Mantaras distance* (D_{ML}), and two correlation matrices: simple Pearsons linear correlation coefficient, and Spearman's rank correlation.

Both of these problems - building the model for predicting amount of chemical elements necessary to obtain the correct steel grade and the comparison of both feature ranking families are covered in this paper.

Next section briefly describes feature selection, in detail describing ranking based methods, with four different ranking criterions. Section (3) provides phases of metallurgical problem modeling, while section (4) presents results of feature selection for all listed ranking methods. Last section concludes the paper.

2 Feature Selection Methods

Feature selection is an important problem in data mining tasks. Selecting the best subset of n features out of a full set of m is an NP-hard problem. Already many different search strategies have been developed for solving that problem [5], however in real application most of them can not be applied because of computational complexity. There are four types of methods used in feature selection:

- Embedded methods - where feature selection is embedded into the prediction algorithm, like in decision trees
- Filter method - where feature selection is done independently of the classification model. As presented in fig. (1) the filter selects a feature subset using evaluation function defined as some statistical criteria like *Kullback-Leibler divergence* or *mutual information*. The advantage of this approach is its speed, being one of the fastest methods, and taking feature subsets independently of the classifier. However, its drawback is that the feature subset is not optimally selected for a particular prediction model.
- Wrapper approach - [7] where the evaluation function used to determine the quality of the feature subset is defined as the accuracy of a specific classifier, which is also used as a predictive model. A block diagram of the that approach is presented in fig. (2). The most important advantage of that approach is the prediction accuracy

of the final model, unfortunately this occurs at a higher computational complexity needed to train the classifier for each searching step.

- Frappers approach - is a combination of filters and wrappers, where hyperparameters of the feature filter are tuned by the accuracy of the classifier. A scheme of this approach is represented in fig. (II). This approach is a compromise between filters and wrappers with advantages and disadvantages of both approaches.

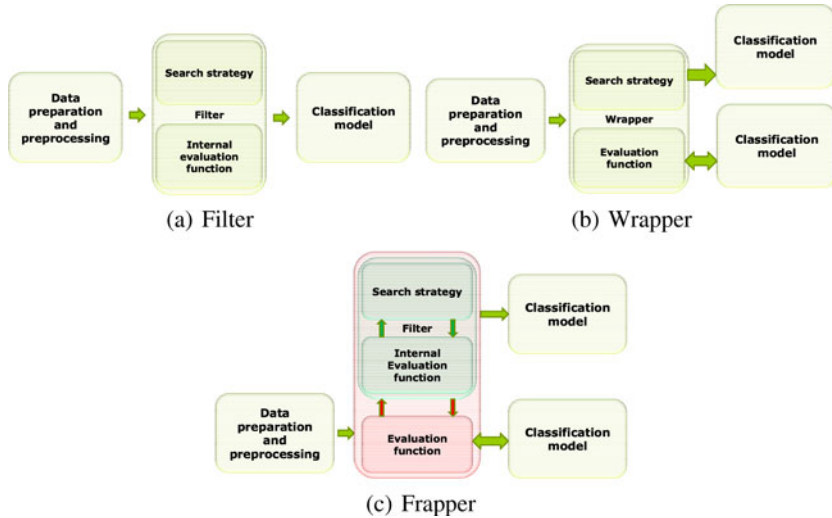


Fig. 1. Three types of feature selection methods

2.1 Ranking Based Feature Selection

One of the simplest and fastest group of methods dedicated to feature selection are ranking based algorithms. In this approach, which belongs to feature filters, each feature is independently assigned a coefficient $J_i = f(\mathbf{f}_i, \mathbf{y})$ that describes its abilities to predict values of output variable \mathbf{y} given just single input variable \mathbf{f}_i . Having such a coefficient features are sorted from the most relevant (with the highest coefficient) to the least relevant (with the lowest coefficients value). Then from initial m features, first n features are selected, and for this subset the prediction model is build. The sketch of this algorithm is presented in fig. (II).

As described above this is the fastest algorithm while its computational complexity is linear with respect to the number of features $O(m)$. The only problem of the algorithm is the determination of the correct number of the selected features (n). This part can be done using wrappers approach, where the quality of the selected number of features is determined using a prediction algorithm. In ranking based methods, one problem still remains; what kind of relevance index $J(\cdot)$ should be used. In classification problems a comparison between different ranking algorithms can be found (ex. in [2]) where authors where unable to determine the best relevance index, and they concluded that the quality of the filter strongly depends on the analyzed dataset. Based on such conclusions

Algorithm 1. Feature selection based on feature ranking

```

Require:  $\mathbf{f}$  {Initial feature set}
Require:  $\mathbf{y}$  {Output variable}
Require:  $J(\cdot)$  {Ranking function}
  for  $i = 1 \dots m$  do
     $a_i \leftarrow J(f_i, \mathbf{y})$  {Calculate rank value}
  end for
 $\mathbf{f} \leftarrow \text{sort}(\mathbf{f}, \mathbf{a})$  {Sort features according to  $J()$ }
 $acc \leftarrow 0$ 
 $\mathbf{f}^a \leftarrow \emptyset$ 
for  $j = 1 \dots m$  do
   $\mathbf{f}^a = \mathbf{f}^a \cup f_j$  {Add new feature  $\mathbf{f}^a$  to current subset}
   $tacc \leftarrow \text{ocení}(\mathbf{f}^a)$  { estimate subset quality  $\mathbf{f}^a$  }
  if  $tacc > acc$  then
     $acc \leftarrow tacc$  {If new subset is better then the previous one }
     $\mathbf{f}' \leftarrow \mathbf{f}^a$  {store current subset}
  end if
end for
return  $\mathbf{f}'$ 

```

while building a model to predict the amount of necessary steel compounds we did a comparison between different rankers comparing two families of indexes - correlation based [9] and based on information theory [2].

Pearson Correlation. The simplest method determining J_i value for regression problems is Pearsons linear correlation coefficient. The value of J is calculated according to the formula

$$J_P(\mathbf{f}_j, \mathbf{y}) = \left| \frac{\sum_{i=1}^k (\mathbf{y}_i - \text{mean}(\mathbf{y})) (\mathbf{f}_{i,j} - \text{mean}(\mathbf{f}_j))^2}{\sigma_{\mathbf{f}_j} \sigma_{\mathbf{y}}} \right| \quad (1)$$

where:

- $\sigma_{\mathbf{f}_j}$ and $\sigma_{\mathbf{y}}$ are standard deviation of variables \mathbf{f}_i and \mathbf{y} .
- $\text{mean}(\mathbf{x})$ is a function returning mean value of given vector \mathbf{x}

This well known and popular coefficient is able to find only linear dependence between two random variables.

Spearman's Rank Correlation Coefficient. A more advanced correlation coefficient is Spearman's rank coefficient, which is able to find nonlinear correlations between two random variables. This metric is based on replacing original values of variables by associated rank values \mathbf{r} , and calculating earlier defined Pearsons linear correlation coefficient for variables replaced by their ranges.

$$J_S(\mathbf{f}_j, \mathbf{y}) = J_P(\mathbf{r}_{\mathbf{f}_j}, \mathbf{r}_{\mathbf{y}}) \quad (2)$$

Where:

- r_{f_j} - rank values associated to f_j variable
- r_y - rank values associated to the y variable

Appropriate ranks are given by ascending sorting variable values and assigning to each value number equal to the position in the sorted list. In case of ties, when certain value (a) appear q -times, where $q > 1$ (more then ones) $a = [v_i, v_{i+1}, v_{i+q}]$, the rank assigned with each of $v_i \cdot v_{i+q}$ values is equal mean rank

$$\forall_{e=i, \dots, i+q} v_e = \frac{1}{q} \sum_{z=i}^{i+q} (r_z) \tag{3}$$

This solution is able to find monotonic nonlinear correlation between features.

Information Gain Ratio. *Information Gain Ratio* (IGR) is a metric that belongs to information theory coefficients. It is based on Shanon entropy where:

$$\begin{aligned} H(\mathbf{x}) &= - \sum_{i=1}^c p(x_i) \lg_2 p(x_i) \\ H(\mathbf{x}, \mathbf{y}) &= - \sum_{i,j=1}^{n,m} p(x_i, y_j) \lg_2 p(x_i, y_j) \\ H(\mathbf{x}|\mathbf{y}) &= H(\mathbf{x}, \mathbf{y}) - H(\mathbf{x}) \end{aligned} \tag{4}$$

- $p(x_i)$ - is the probability of x_i

used to define *mutual information*:

$$MI(\mathbf{x}, \mathbf{y}) = -H(\mathbf{x}, \mathbf{y}) + H(\mathbf{x}) + H(\mathbf{y}) \tag{5}$$

The final formula determining is defined as:

$$IGR(\mathbf{x}, \mathbf{y}) = \frac{MI(\mathbf{x}, \mathbf{y})}{H(\mathbf{x})} \tag{6}$$

The IGR metric describes the amount of information about variable y providing by variable x . The IGR value is equal to the *Information Gain* normalized by the entropy of variable x .

Mantaras Distance Ranking. Mantaras distance D_{ML} is a matric defined as

$$D_{ML}(\mathbf{x}, \mathbf{y}) = H(\mathbf{x}|\mathbf{y}) + H(\mathbf{y}|\mathbf{x}) \tag{7}$$

the advantage of D_{ML} metric is the ability to preserve all distance axioms.

3 Dataset and Preprocessing Steps

The dataset used to build the prediction model consists of historical data obtained from one of polish steel mills. The data describes refining and fine tuning steel parameters during the ladle arc furnace stage of melting. The problem is described as predicting the amounts of the three most important chemical elements (carbon, manganese and silicon) necessary to assure appropriate steel quality based on 41 features. The amounts of these three elements were calculated from the amounts of additives that were added into the steel, since each additive contains a constant percentage of given elements. The input variables are the weight of steel tapped into the LHF furnace, amount of energy needed to melt the steel (in the EAF furnace), amount of oxygen utilized during the EAF stage, results of chemical analysis before discharging the EAF furnace, and results of chemical analysis of the steel during the LHF process. The histograms for one steel grade called S235JRG2 are presented below.

3.1 Amount of Carbon Analysis

The histogram of output variable before any preprocessing is presented in fig. (2). Analysis of this histogram points out that predicting the amount of Carbon over 100kg is impossible due to the low number of training samples. To avoid the problem of unstable behavior of the training process caused by the small number of training samples over 100kg, the outlier analysis was performed based on interquartil range. Also to avoid the problem of dominating 0 value, when no Carbon was added, all samples with carbon = 0 were removed, the histogram of the output variable that remained in the dataset is presented in fig. (2). A regression model was built for that data. The process of predicting 0 carbon was further performed based on a binary classification problem, where class C_{-1} was related to "no carbon", and class C_1 to "add carbon", while the mass of necessary carbon was delivered by the regression model. In this paper we present results only for regression problems.

3.2 Amount of Manganese Analysis

The analysis of histograms of the output variable for Manganese (fig.(2)) suggests the existence of two clusters. One for the amount of Manganese below 200kg and the second one for Manganese amount over 200 kg. After the outlier analysis that was shrank to the values between 200 and 500 kg.

3.3 Amount of Silicon Analysis

A similar problem of a dominating histogram bar for 0 value that we faced during Carbon analysis appeared also during Silicon analysis. The histogram before and after removing dominating 0 are presented in figures (2c,d) respectively. Also in this problem classification model was built to predict either "no Silicon" or "add Silicon", and also only the results of the regression algorithm are presented.

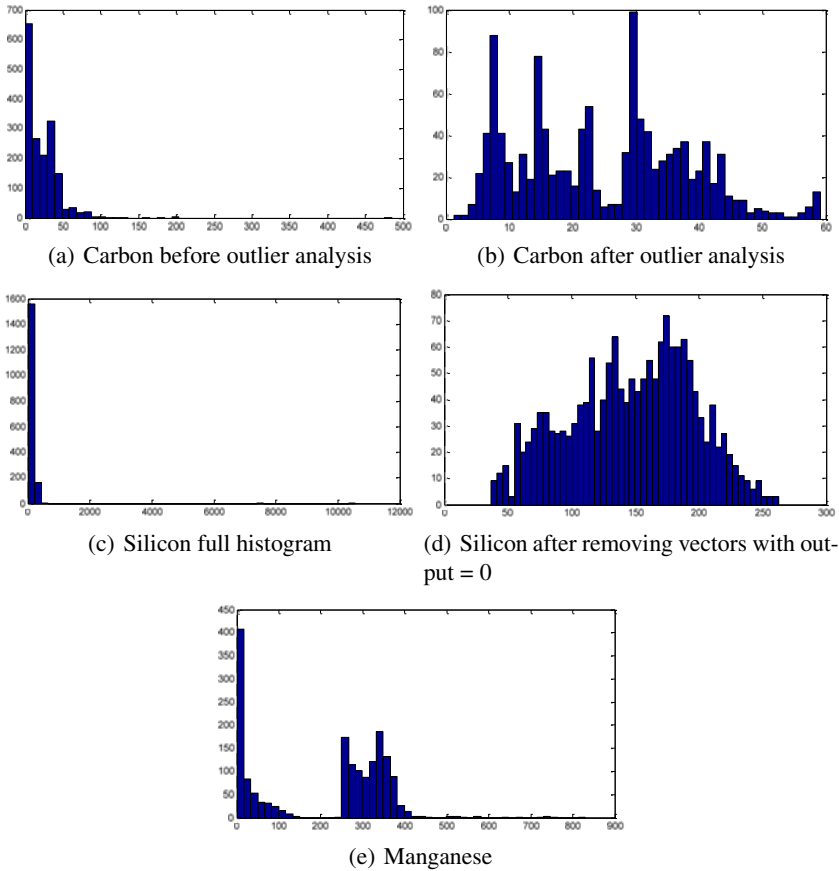
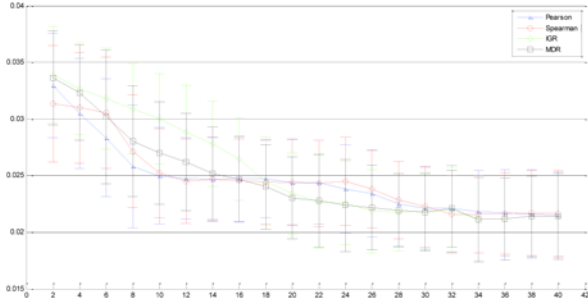


Fig. 2. Histograms of the output variable for chemical elements

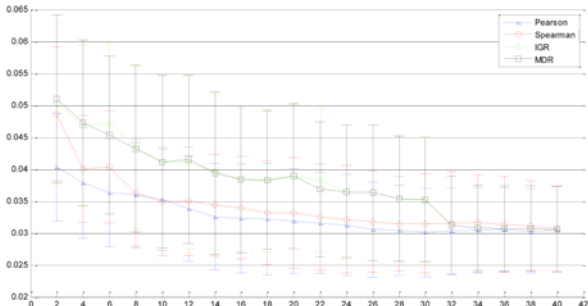
4 Comparison of Ranking Criteria

As suggested in the introduction, SVM for regression (SVR) [3] with Gaussian kernel and ϵ -insensitive cost function was selected to be used for solving all regression problems. Unfortunately training SVR algorithm requires searching for optimal hyperparameters in the cubic space. The hyperparameters of the SVR model are margin softness (C - value), kernel parameter (γ) and ϵ in the ϵ -insensitive cost function. This problem was solved by greed search algorithm, where $C = [12832128]$, $\gamma = [0.50.711.31.5]$, and $\epsilon = [0.10.010.001]$.

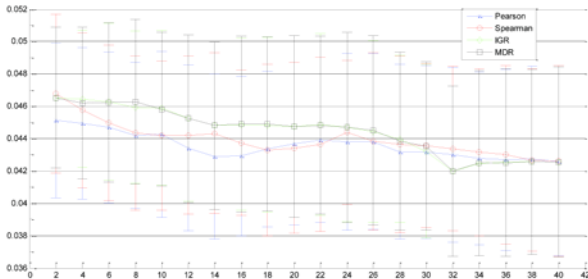
For each number of features ordered by the ranking, SVR with all possible set of parameters was trained. Results presented in the figures were obtained with a 10-fold cross validation (CV) that was repeated 3 times (3×10 CV). The best of all obtained results for a given number of features were selected and presented in figure (3) (smallest MSE). All calculations were performed using Matlab toolbox for data mining called Spider [8], extended by the Infosel++ feature selection library created by our group [6].



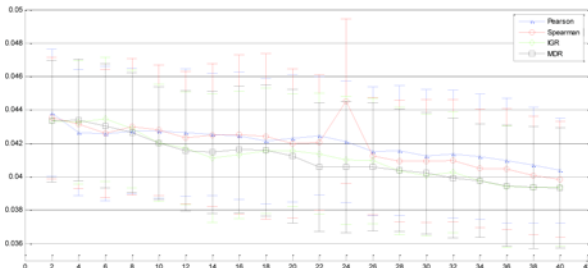
(a) Manganese Type A



(b) Manganese Type B



(c) Carbon



(d) Silicon

Fig. 3. MSE and its standard deviation in the function of selected features

5 Conclusions

As suggested in the introduction, this paper covered two types of problems: building a statistical model for metallurgical problems and comparing two groups of ranking criteria used for feature selection. After outlining the problem and discussing different types of feature selection and an in-depth analysis of ranking based feature selection we have discussed the steps necessary to take before building regression models. While building the model we have found the problem of small number of samples for some ranges of output variable. Because of that we decided to remove such ranges performing outlier analysis. This helped us obtain better accuracy of our model. Similarly much better results were obtained splitting the output variable into two groups like for Manganese problem.

We have also separately considered the problem of feature selection based on two types of ranking criteria. The first of them are information theory metrics such as *Information Gain Ratio* and *Mantaras distance*. The second group consists of two correlation metrics: Pearson's linear correlation and Spearman's rank correlation. The obtained results were surprising since in almost all cases the number of all relevant features was almost equal to the whole number of features in the dataset. For Manganese (B) and Carbon both *correlation based criteria* produced better results than *information theory metrics*. However, the best average results for Carbon dataset were obtained for 32 features using both information theory metrics, while for Manganese all features were required. In case of silicon dataset it can be seen that in the beginning the best results are provided by the linear correlation metric while in the second part for higher number of features information theory metrics started to be more accurate, allowing to obtain the best results for 34 features. Analysis of Silicon problem shows that both *information theory* indexes allowed to obtain better results in the whole analyzed feature space. Concluding *information theory* based ranking seems to perform better than the correlation measures, however not for all cases (ex. Manganese (B)). Obtained results allow for another interesting observation that the results obtained by Spearman's rank correlation were usually poorer than that of the simple linear correlation index.

Acknowledgement. This work was funded by the Polish Committee for Scientific Research grant 2007-2010 No.: N N519 1506 33.

References

1. Lis, T.: Współczesne metody otrzymywania stali. Wydawnictwo Politechniki Śląskiej, Gliwice (2000)
2. Duch, W., Wiczcerek, T., Biesiada, J., Blachnik, M.: Comparison of feature ranking methods based on information entropy. In: Proc. of International Joint Conference on Neural Networks (IJCNN). IEEE Press, Budapest (2004)
3. Chang, C.-C., Lin, C.-J.: LIBSVM: a library for support vector machines (2001), <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
4. Duch, W.: Filter methods. In: Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L. (eds.) Feature extraction, foundations and applications, pp. 89–118. Springer, Heidelberg (2006)
5. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L. (eds.): Feature extraction, foundations and applications. Springer, Berlin (2006)

6. Kachel, A., Blachnik, B., Duch, W., Biesiada, J.: Infosel++: Information Based Feature Selection C++ Library, <http://metet.polsl.pl/jbiesiada/infosel> (in preparation)
7. Kohavi, R., John, G.: Wrappers for Feature Subset Selection. *Artificial Intelligence journal*, special issue on relevance 97(1-2), 273–324 (1997)
8. Weston, J., Elisseeff, A., BakIr, G., Sinz, F.: Spider 1.71 (2006), <http://www.kyb.tuebingen.mpg.de/bs/people/spider/>
9. Hollander, M., Wolfe, D.A.: *Nonparametric statistical methods*. Wiley, Chichester (1973)

Generic Model for Experimenting and Using a Family of Classifiers Systems: Description and Basic Applications

Cédric Buche and Pierre De Loor

UEB / ENIB / LISyC
European center for virtual reality
Technopôle Brest-Iroise F-29280 Plouzané, France
{buche, deloor}@enib.fr

Abstract. Classifiers systems are tools adapted to learn interactions between autonomous agents and their environments. However, there are many kinds of classifiers systems which differ in subtle technical ways. This article presents a generic model (called GEMEAU) that is common to the major kinds of classifiers systems. GEMEAU was developed for different simple applications which are also described.

1 Introduction

Defining the behavior of autonomous artificial entities is faced with the problem of selecting a model able to account for the link between perceptions and actions in an efficient manner. There are a great number of proposed solutions to this issue. However, they require detailed descriptions which are difficult to achieve, either because they require a definition based on *a priori* rules and symbols [28], or because they are subject to configuration difficulties and behavioral indeterminism [17]. Another solution would be to define the entities' initial approximate behavior, which would then adapt according to its environment. This solution is implemented by classifiers systems. It uses a set of competing rules and incorporates learning processes by choosing and improving these rules. A great deal of literature exists on the subject [3,12,6,13,11]. A number of authors have put forward different variations of the approach, each offering different mechanisms adapted to specific problems. Our objective is to be able to test and advance these mechanisms without difficulty, consequently we are interested in designing and implementing a generic model.

This article is organized as following: first we present the general mechanisms of classifiers system. We then go on to present a generic model, called GEMEAU¹, which integrates these mechanisms, and with which we can easily test different versions. Next we explain how we applied this model to different types of applications: multiplexers and *woods* environments.

¹ **GEMEAU: G**ENERIC **M**ODEL for **E**xperimenting **A**ND **U**sing a family of classifiers systems.

2 Classifiers Systems

2.1 Principles

A classifiers system is an adaptive system that learns to perform the best action given its input. The system manage a combination of "condition-action" rules called classifiers, pondered by quality parameters. The system perceives its environment (usually a vector of numerical values), deduces the applicable rules, carries out an action as a result of these rules. Technically, when a particular input occurs, classifiers whose conditions are satisfied by that input are selected and the system chooses one of the possible actions (from selected rules) according to the quality parameters. The system is able to receive a reward from the environment which it then uses to modify the rules or their quality parameters.

Through experimentation, classifiers system can therefore be used to learn the association between conditions and actions, thus maximizing credit intake. In order to avoid a combinatorial explosion of the quantity of rules, they are generalized; they apply to different perceptions of the environment. Mechanisms which allow the creation, enrichment (specialization/generalization), or destruction of these rules must therefore be used. Evolutionary algorithms are often used to do this, even though other heuristic approaches are available. The qualities of the rules are modified dynamically through reinforcement learning, and the rules themselves are modified by genetic algorithms.

2.2 Formalization

In this section we propose the incremental and generic formalization of classifiers systems, and gradually introduce learning mechanisms.

The global structure of a classifier system, is a 7-uplet $(I_i, [P], [M], [A], Matching, Selection, I_o)$:

- I_i is the interface input due to which each *Perception* within the environment corresponds to a binary code.
- $[P]$, (*population*), is the set of the system's classifiers, coded by a succession of n bits². The generalizing representations contain $\#$ symbols which correspond to an indeterminate value. A rule is a (C, A) pair with $C \cup A \in \{0, 1, \#\}^n$ where :
 - C : the condition for application of the rule.
 - A : the action(s) associated with the application of the rule.

Let us take the example of a robot with four 'all-or-nothing' sensors and one action. The input interface converts the state of the sensors into a binary value and the output interface triggers the action depending on the action's bit value. Thus, a $\{011\#, 1\}$ rule means that the rule is applicable if the first sensor is inactive and the two following sensors active. The state of the fourth sensor has no influence, and applying the rule triggers the action.

- $[M] \subseteq [P]$ is the set of classifiers of which the condition element pairs with the perceived environmental information during a selection cycle. This is known as *Match-set*.

² Even if certain systems work with other alphabets [9][17][5].

- $[A] \subseteq [M]$ is the set of classifiers representing the selected action. This is known as *Action-set*.
- *Matching* is the mechanism which makes the transition from $[P]$ to $[M]$ possible. This is generally achieved using a matching rule between C and the information derived from I_i . This rule is able to interpret the generalization symbols that make up the classifier conditions.
- *Selection* is the mechanism which makes the transition from $[M]$ to $[A]$ possible. Depending on the details of the different versions of classifiers systems, it is able to determine the desired action.
- I_o is the output interface through which the activated *Action* corresponds to a binary code.

Learning occurs due to an evaluation of the quality of the rules represented by one or a number of additional parameters. The definition of a classifier is thus extended to a $R = (C, A, f)$ triplet where f characterizes its quality. Learning developed by Rewarding the rules, by altering their quality using reinforcement learning algorithms and by Generating rules using evolutionary and heuristic covering algorithms. The dynamics of learning classifiers systems are therefore based on the following cycle: Perception / Matching / Generation (*covering*) / Selection / Action / Reward / Generation (*evolutionary algorithm*).

Selection is guided by the quality of the rules, which are grouped together depending on their $[A]$ element. Often, a 'wheel of fortune' mechanism³ is applied, which means that each package has a probability proportional to its capacity to be selected. The credit assignment (Reward) mechanism distributes credit to the rules that have contributed to its acquisition. It increases the quality of the rules triggered prior to the acquisition of the credit and decreases that of the others. Its definition affects the relevant length of a series of actions: i.e. the number of rules in a sequence considered necessary in order to achieve a certain goal. The generation mechanism must both minimize the number of rules and conserve those which assist in achieving credit. A good rule is therefore a high-quality generalizing rule (relative to the others). The two generation (*rules discovery*) mechanisms used are *covering* (creation of rules when no classifiers match the perception of the environment) and *evolutionary algorithms*.

3 GEMEAU

Classical classifiers systems (ZCS, XCS, ACS, Hierarchical ...) [15][16][24] go some way to finding optimal solutions in Markovian or non-Markovian environments. Nevertheless, as Sanza notes [10], the improvements and supplementary systems are suitable only for specific cases and none of the models are able to supply an overall solution for all of the problems (XCS is only effective if the credits are discrete and of a fixed quantity; ACS is only useful if each action leads to a modification in the perception of the world ...).

³ The wheel of fortune mechanism consists of picking elements randomly, so that their probability of being chosen is proportional to their selectivity.

There are, therefore, a great number of classifiers systems [14]. Developing and testing a variety of such systems take time and is not easy. Using the structure and dynamics analysis conducted previously we were able to come up with a generic background for a whole family of classifiers systems. Our architecture claims to be generic, in the sense that it can be used to implement ZCS and XCS systems (ZCS, XCS, ZCSM, XCSM).

3.1 Architecture

The architecture is displayed in Fig. 1 as a UML classes diagram. The system is called GEMEAU. It is based around two components: *interface with the environment* and *system*.

The interface with the environment determines the interactions between the system and environment, both of which are common to different classifiers systems. In our model, the different interfaces are implemented using three categories: *CS_I*, *CS_O* and *CS_R* (respectively entry interface, output interface and credit). Communication between the interfaces and the environment takes place in the form of messages, enabling the classifiers system to have an implementation in parallel to the environment.

The System defines the elements and the mechanisms of our classifiers system in concrete terms. Let us consider the following elements:

- A classifier (*CS_Classifier*) is made up of several parts: condition (*CS_Condition*), action (*CS_Action*) and configuration (*CS_Parameter*);
- The sets $[P]$, $[M]$, $[A]$ and $[A]_{-1}$ are lists of *CS_ClassifierList*-type classifiers.

We put forward the following mechanisms:

- The **Matching** mechanism, with which the classifiers that match the information coming from the environment can be extracted. It is included in the *CS_ClassifierList* by the *match()* method;
- The **Generation** mechanism by *covering*, which creates rules according to the content of $[M]$ after **Matching**. It is included in the *CS_ClassifierList* by the *cover()* method which can be configured (notably for the number of #);
- The general (*CS_System*) method represents the workings of a given cycle (*step()* method);
- The **Selection** mechanisms of the winning (*CS_SelectorAlgo*) actions (which must be able to differ according to the desired learning);
- The **Reward** mechanism, (*CS_AOAlgo*), modifying the classifiers' configuration.
- The **Generation** genetic algorithm, (*CS_GeneticAlgo*), where different operators must be specified, i.e. crossing or mutation.

3.2 Use

GEMEAU can be specialized in order to obtain a ZCS (Fig. 1). Through inheritance, we can define:

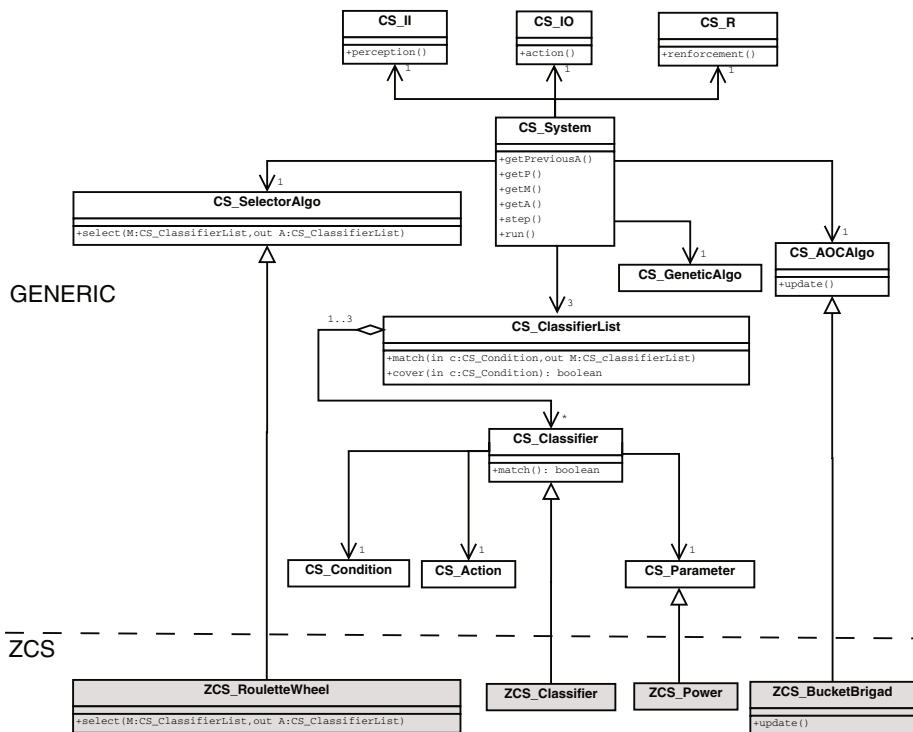


Fig. 1. UML diagram of GEMEAU augmented by a ZCS using inheritance

- The Rules (*ZCS_Classifier* derived from *CS_Classifier*) having a configuration force (*ZCS_Power* derived from *CS_Parameter*);
- The 'wheel of fortune' type Selection mechanism *ZCS_RouletteWheel* derived from *CS_SelectorAlgo*;
- The *Bucket Brigade*-type Reward mechanism (*ZCS_BucketBrigad* derived from *CS_AOCAalgo*);
- The Generation genetic algorithm (*ZCS_GeneticAlgo* derived from *ZCS_GeneticAlgo*) notably specifying that the selective value is the rule's strength.

The rest of the system uses the pre-existing default mechanisms such as *covering*. We implemented the XCS classifier system using the same techniques. In order to do so, we must redefine *CS_Parameter*.

We can also easily add memory to ZCS in order to obtain a ZCSM [3]. In that case, we must increase the result of perception using an internal classifier system register which is modified by part of the last action to have been carried out. Using GEMEAU, the *step()* method can be redefined simply by inheriting from the *CS_System*. We implemented the XCSM classifier system using these same principles.

By using these specialization and extension mechanisms we were able to use our architecture to implement and test ZCS and XCS family classifiers systems (ZCS, ZCSM,

XCS, XCSM). Our perspectives are based on the implementation of supplementary traditional anticipatory or hierarchical systems (ACS: anticipation [12] and ALECSYS: hierarchy [4]). Implementing these systems could well be less simple than for the family of ZCS and XCS, and the architecture may need to be modified.

3.3 Validation

One simple and frequently used evaluation environment is a multiplexer. Let us consider a multiplexer with an input of 6 bits $a_0, a_1, d_0, d_1, d_2, d_3$ and an *output* of one bit. a_0 and a_1 correspond to address bits. The multiplexer equation is $output = \overline{a_0}.\overline{a_1}.d_0 + \overline{a_0}.a_1.d_1 + a_0.\overline{a_1}.d_2 + a_0.a_1.d_3$. The output will be either the value of d_0, d_1, d_2 or d_3 , depending on the address. The aim is to find this multiplexing function.

Using GEMEAU, we must simply determine the detectors and effectors that interface with the environment plus the reinforcement to be distributed, and then instantiate a *CS_System* (Fig. 2). The conditions and actions of the classifiers here correspond to the multiplexer's input and output respectively. The classifier system is rewarded when the rule selected corresponds to the multiplexing function.

```

/* Environnement */
env = new EnvironmentMultiplexer(nbEntries,nbOut);
detector = new CS_II_Boolean(env);
effector = new CS_IO_Boolean(env);
reinforcement = new CS_R(env);
/* System */
system = new CS_System();
system->setDetector(detector);
system->setEffector(effector);
system->setReinforcement(reinforcement);
system->run();

```

Fig. 2. Use of GEMEAU for a multiplexer-type environment

Another advantageous evaluation environment is the *woods* environment. It corresponds to a graphical representation based on an infinitely repeating pattern. It represents a forest and the aim of a situated agent is to find food. Within this forest there are insurmountable obstacles (trees) and areas of unrestricted movement. The perception of the agent corresponds to a representation of the 8 squares surrounding the occupied position. The simplest of these environments is *woods1* (Fig. 3a). This is a deterministic Markovian environment⁴. The *woods100* (Fig. 3b), however, is non-Markovian. Indeed the optimum displacement from square number 2 is to the right although for square number 5 it is to the left even though these two squares are perceived identically.

The system learns by alternating between an iteration in exploration mode (selecting the action using the 'wheel of fortune' mechanism) and an iteration in exploitation mode (choosing the best action). The curves only take into account the results in exploitation mode, dealing with the average of the last ten iterations.

GEMEAU can deal with this two classical examples, it converges for the multiplexer (Fig. 4a) and for *woods1* (Fig. 4b). For the multiplexer, we achieve a 100% success

⁴ There are no perceptions values corresponding to different states of the agent.

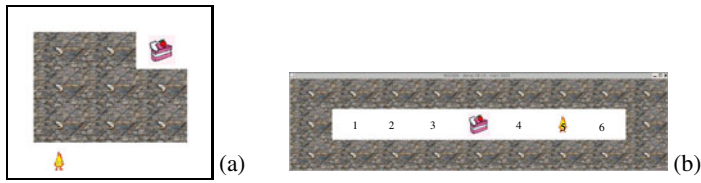


Fig. 3. Markovian woods1 (a) and non-Markovian woods100 (b) environments

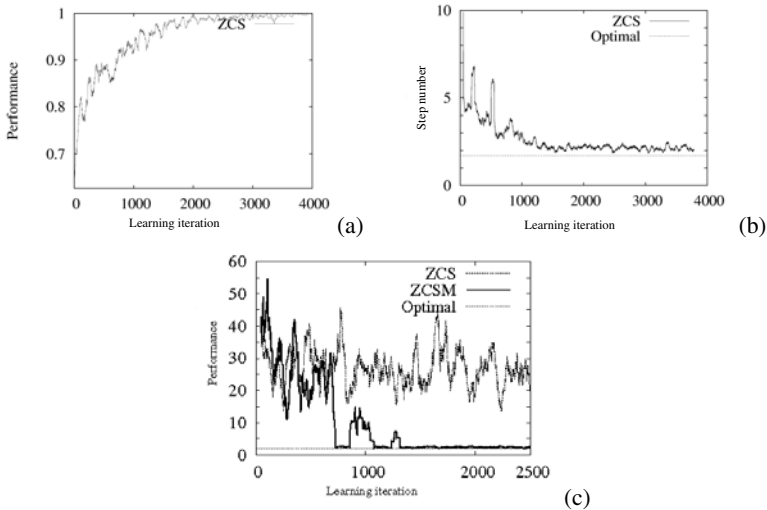


Fig. 4. ZCS multiplexer learning (a) and in woods1 (b). As of a sufficient number of iterations, our system conducts the multiplexing function and obtains the minimum number of movements in the case of woods1. ZCS and ZCSM learning in woods100 (c).

rate. For *woods1*, we achieve solutions similar to the minimum number of movements. The results are conclusive: in both cases we reported performances similar to those described in the results of [15]. Furthermore GEMEAU allows the rapid evaluation of derived classifier systems. We compared the ZCS and ZCSM results in the *woods100* non-Markovian environment (Fig. 4c). Our results rediscover the ZCS' difficulties obtaining optimal rules in non-Markovian environments. They also confirm that our architecture can be used to extend the capacities of ZCS to non-Markovian environments.

4 Conclusion

Having described existing classifiers systems, we illustrated a more general classifiers system which groups together the traditional systems. We put forward our model called GEMEAU enabling traditional systems and their variants to be both modelled and extended. This implementation is flexible enough to be used for a variety of problems as it proposes an interface between the environment and the classifier system (input/output/reinforcement). It has been used to test many types of classifiers systems and

different conceptual hypotheses quickly, as well as to obtain significant comparative results. Among other things, these tests showed us the interest of being able to access a library of classifiers systems with which we should be able to define a methodology for choosing learning algorithms based on certain stages of the tests.

References

1. Brooks, R.: Elephants don't play chess. *Robotics and Autonomous Systems* 6(1&2), 3–15 (1990)
2. Carver, M., Lesser, V.: The evolution of blackboard control architectures. Tech. Rep. UMC-92-071, Department of Computer Science, University Massachusetts (1992)
3. Cliff, D., Ross, S.: Adding Temporary Memory to ZCS. Tech. Rep. CSRP347, School of Cognitive and Computing Sciences, University of Sussex (1995)
4. Dorigo, M.: Alecsys and the AutoMouse: Learning to Control a Real Robot by Distributed Classifier Systems. *Machine Learning* 19, 209–240 (1995)
5. Heguy, O., Sanza, C., Berro, A., Duthen, Y.: GXCS: A generic classifier system and its application in a real time cooperative behavior simulations. In: *International Symposium and School on Advanced Distributed System* (2002)
6. Lanzi, P., Wilson, S.: Optimal classifier system performance in non-Markov environments. Tech. Rep. 99.36 (1999)
7. Maes, P.: The dynamics of action selection. In: *Proceedings of the international Joint Conference on Artificial Intelligence* (1989)
8. Mateas, M.: An Oz-centric review of interactive drama and believable agents. In: Veloso, M.M., Wooldridge, M.J. (eds.) *Artificial Intelligence Today. LNCS (LNAI)*, vol. 1600, pp. 297–328. Springer, Heidelberg (1999)
9. Matteucci, M.: Fuzzy learning classifier system: Issues and architecture. Tech. Rep. 99.71 (1999)
10. Sanza, C., Heguy, O., Duthen, Y.: Evolution and cooperation of virtual entities with classifier systems. In: *Eurographic Workshop on Computer Animation and Simulation* (2001)
11. Sigaud, O., Wilson, W.: Learning classifier systems: A survey. *Journal of Soft Computing* 11(11), 1065–1078 (2007)
12. Stolzmann, W.: Anticipatory classifier systems. In: *Third Annual Genetic Programming Conference*, pp. 658–664. Morgan Kaufmann, San Francisco (1998)
13. Tomlinson, A., Bull, L.: A zeroth level corporate classifier system. In: *Second International Workshop on Learning Classifier Systems*. Springer, Heidelberg (1999)
14. Urbanowicz, R., Moore, J.: Learning classifier systems: A complete introduction, review, and roadmap. *Journal of Artificial Evolution and Applications*, 25 (2009)
15. Wilson, W.: ZCS: A zeroth level classifier system. *Evolutionary Computation* 2(1), 1–18 (1994)
16. Wilson, W.: Classifier Fitness Based on Accuracy. *Evolutionary Computation* 3(2), 149–175 (1995)
17. Wilson, W.: Get real! XCS with continuous-valued inputs. In: Lanzi, P.L., Stolzmann, W., Wilson, S.W. (eds.) *IWLCS 1999. LNCS (LNAI)*, vol. 1813, pp. 209–219. Springer, Heidelberg (2000)

Neural Pattern Recognition with Self-organizing Maps for Efficient Processing of Forex Market Data Streams

Piotr Ciskowski* and Marek Zaton

Institute of Computer Engineering, Control and Robotics,
Wroclaw University of Technology

Abstract. The paper addresses the problem of using Japanese candlestick methodology to analyze stock or forex market data by neural nets. Self organizing maps are presented as tools for providing maps of known candlestick formations. They may be used to visualize these patterns, and as inputs for more complex trading decision systems. In that case their role is preprocessing, coding and pre-classification of price data. An example of a profitable system based on this method is presented. Simplicity and efficiency of training and network simulating algorithms is emphasized in the context of processing streams of market data.

1 Introduction

Neural networks have been widely used in solving problems from financial domain ([1]). Among such tasks as credit assessment, fraud detection, option pricing etc., a wide spectrum of applications is devoted to price timeseries analysis and forecasting. Solutions found in literature use various types of neural nets to forecast future values of the analyzed instrument or to provide a good trading decision. An example of methodology of building trading systems using neural nets may be found in [4]. Neural nets have also been used for pattern classification of stock market data, some methods even used Japanese candlesticks (e.g. [3]), however in not so straight and clear form as our method.

Our approach does not consider forecasting itself, we do not use neural nets as "black boxes" to predict future from a set of present and past market data (prices, indicators etc.). We aim at dividing the decision process into functional blocks or stages, for which different specialized types of neural nets may be used as assistance. The idea is close to modeling intellectual activity of humans, in this case technical analysts, for whom the visual and geometrical analysis of patterns on charts is only one specific (often unintentional) activity in the whole decision process, which either precedes other activities, influences them, or provides a context for them.

In this paper we focus on only one such activity - the analysis of single candles and small groups of candles on the chart. Particularly, self organizing maps will

* This work is supported by KBN grant in the years 2006-2009.

be used as tools for recognizing patterns in price timeseries and for coding the recognized formations as inputs for further steps of decision making. Their role in the whole decision process may be considered as preprocessing, compression, or initial classification of candle data. Along that task, self organizing nets will provide us with clear and coherent maps of formations for the analyzed instrument, preserving the density, topology and similarity of candle formations on a particular chart. Further in the paper we will present a trading system, based on a feedforward neural net, taking signals from the above mentioned self organizing maps. The design and performance of that net is not the main scope of our work, however. We would like to focus on the Japanese candlestick methodology, self-organizing maps as a tool for recognizing patterns, and joining these two techniques for efficient decision tools. We will operate on the largest and most fluent, unregulated market for exchanging derivatives on currencies, commodities, metals, stocks and indices - the "forex" market, while the proposed solution may be applicable to all markets.

2 Neural Nets for Processing Price Data Streams

We are looking at the problem of stock/forex market pattern recognition also from the viewpoint of processing intensive data streams. Indeed the price data, here called the quotes, arrive at the trading platform in the form of an intensive stream. The prices on the most popular and fluent instruments (EURUSD and other major currency pairs, most popular commodities, or stock indices) change every few seconds or even a few times in each second. The methods of analyzing these data must be simple and fast. For that reason the candlestick analysis is generally a right tool, as it performs analysis of past candles only once at the beginning of each candle (for the rest of the interval of the current candle it waits until the candle is eventually formed). That may be 1 day, 1 hour, but also 1 minute, depending on the trader's strategy.

Additionally one trading platform may work on many instruments and intervals, for all of which a fast response is needed. Moreover, mobile trading platforms are gaining more and more popularity. For this kind of applications fast and simple algorithms, consuming less computational power, are the key points to successful use. Self organizing maps, described in this paper, provide such efficient tools, they may also be used as a preprocessing stage for simplifying further steps of other decision algorithms. In this case, along with pre-classification of single candles or short formations, they provide data reduction - the four prices of each candle are reduced to a pattern position on the map.

3 Japanese Candles for Analyzing Price Data

Two main ways of analyzing markets are: fundamental and technical analysis. The latter relies only on the charts - geometrical, visual, and recently also computational analysis of the history of prices. Its principles fit exactly into the idea of neural net training. Technical analysis assumes that: - price reflects all

available information, - history repeats itself, - prices move in trends. For technical analysts the patterns recognized in the past are a valuable clue to predict probable scenarios of market moves in the future. These scenarios are built on psychological analysis of traders and on statistical analysis of past data. Although technical analysis is to some extent subjective to the person performing it, it is much more appropriate for neural automation than fundamental analysis. In addition, historical training data are easy to collect.

For many years technical analysts used only the Western techniques to analyze stock and forex markets. These methods were based on traditional charts (linear or bar), a set of measures called technical indicators, theories such as Elliot waves or Fibonacci levels, and formations drawn by prices on the two mentioned kinds of charts. In 1990s the Eastern methodology of Japanese candles was introduced and popularized among Western investors, mostly due to Steve Nison ([2]). It is based on a special kind of a chart.

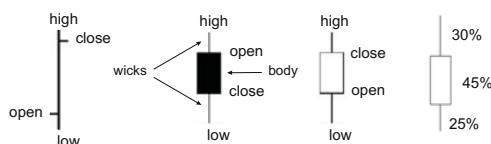


Fig. 1. The structure of a Japanese candle (both middle), compared to a bar (left). Percentage notation of one candle's structure supplied to the SOM inputs (right).

Let us briefly compare the three types of charts used to visualize the market. The simplest linear chart shows only the closing prices for each interval, providing an image of longer market moves, without showing the structure of single candles. Bar and candlestick charts visualize the open, high and low price of each interval. The candle chart is more precise and capable, due to adding color to each bar - white for bullish and black for bearish. The structure of bar and candle is compared in fig. 1. The latter shows the market's "struggle", condition and "mood" during each interval. One-, two- and three-candle formations, built upon psychological background (often called the "crowd psychology"), may signal important moments, such as turns, beginnings and ends of impulse and corrective waves etc. A few examples of trading signals based on candle formation analysis are presented in fig. 2. This illustrative example, although performed "a posteriori" (with the knowledge of each formation's consequences), shows that candle formations occur not incidentally in specific moments in time. In reality, we do not know the right hand side of the chart while analyzing the current candle. We study the charts not to predict the future, but to provide us with possible scenarios of market behavior, to chose the most probable of them, while still be prepared (with trade management techniques) for the other ones. The analysis should be based on various methods and indicators. Pattern recognition of Japanese candle formations should be one of them. A reliable method for automatic recognition and classification of these formations would provide us with

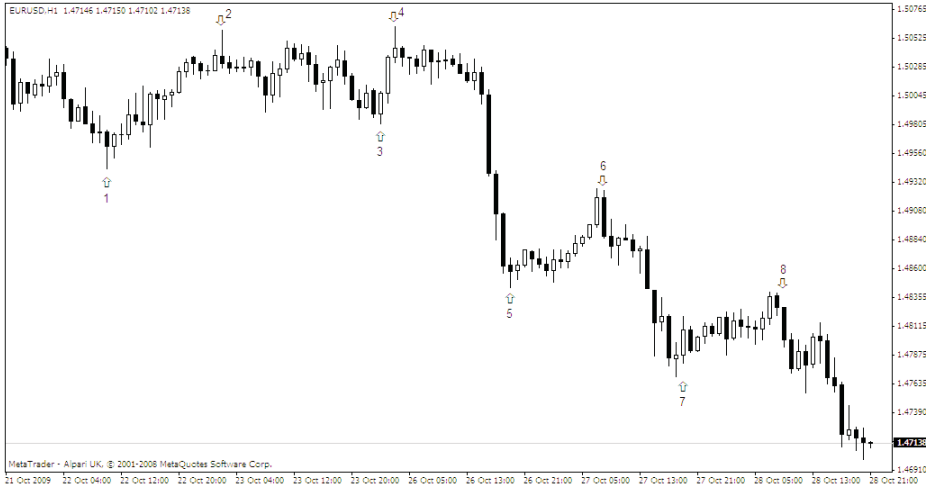


Fig. 2. Candle chart, with example of trading signals based on Japanese candlestick formations

a strong confirmation of other signals and important clues for optimal decisions. Such a tool will be provided by self organizing maps.

The tradition of Japanese candles is long and the number of formations that have been recognized is large. There are about a dozen major formations - well known continuation and reversal patterns, among them: hammer, shooting star, dark cloud, morning star (no. 1, 4, 8, 7 in fig. 2). The names of many of them are very illustrative and poetic. For an in depth geometrical, psychological and tactical analysis of patterns, see [2].

All candle formations may be analyzed not sooner than after their last candle is formed, that is at the beginning of the next candle. It is also taken as a rule that candle formations should be confirmed by other signals or should confirm signals coming from other techniques (e.g. Fibonacci levels or Elliot waves). Generally speaking, the more signals occur at some time, the higher the probability of a specific price move.

4 Neural Pattern Recognition of Japanese Candles

Most applications of neural nets to analyze market timeseries are based on Western methods - that is on different indicators calculated on price data. These data are easy to calculate or to obtain from trading platforms. Traders and researchers use neural nets in their attempts to find dependencies between values of some factors (e.g. indicators) and future prices or proper trading decisions. The analysis of Japanese candles is more geometrical and visual, closer to modeling human perception and intuition rather than strict functional dependency. Most technical indicators are delayed in time as based on moving averages. The analysis of

candle formations is focused on current market behavior - mostly on the previous candle right after it finishes forming its shape. The emphasis is put not only on closing or average values for each timeframe, but also on the structure of the market movement during each period.

We chose self organizing maps (SOMs) for the task of recognizing and coding Japanese candle formations. First of all, these nets learn in unsupervised manner, so no human recognition will be needed to classify formations while preparing training data. The preprocessing of training data will include only a special way of scaling. Secondly, these nets discover clusters of data points in training data. They imitate the density and topology of these data. Therefore they will adapt only to the formations present on charts for the selected currency and timeframe. It is even possible that they will discover new patterns, not described as formations yet, characteristic only for the given instrument. Similar formations will be placed closely to each other on the map. After training the map and labeling its neurons, the winners' positions will clearly indicate the recognized pattern and will be easy to code for further steps of the decision process. Another reason for using SOMs is that their training and functioning algorithms are very simple, providing good performance on streams of data.

5 Self Organizing Maps for Candle Pattern Recognition

We use self organizing maps in their traditional form - a layer of linear neurons with connections to all inputs, arranged on a plane. The neighborhood function uses hexagonal neighborhood topology. When an example is shown to the net during training, the winner - determined by the maximum output value - and its neighbors adapt their weights moving them closer to the inputs. Therefore the neurons neighboring each other in the map learn similar patterns.

An important issue is how the way candle data are applied to the SOM. The basic data of each candle are: O - open, C - close, H - highest, and L - lowest price during the candle's interval. These four values define the candle's position on a chart and its shape - the proportions of its body, upper and lower wicks. We have decided to describe the candle's shape in percents - the length of its body and both wicks as parts of the whole candle's length, as shown in fig. 11 (right). The sign of the body's length defines its direction - positive for bullish and negative for bearish candle. Additionally the price movement between the last two bars should be given on net's inputs to indicate the difference between two adjacent candle's positions. We have also used SOMs for discovering patterns in two- and three- candle formations. In that case two and three candle windows were supplied on the nets' inputs using the same notation.

All neural nets presented in this paper were implemented using MATLAB Neural Network Toolbox. Self Organizing Feature Map from that package was trained in a standard unsupervised way. For one-candle formations nets of 16, 25 and 36 neurons in the layer were trained, for two-candle formations - nets of 49, 81 and 100 neurons, for three-candle formations - nets of 100 and 225 neurons. In all cases the smallest structures provided the best performance. After training

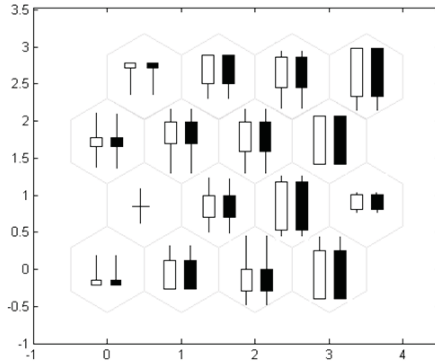


Fig. 3. Topology of patterns recognized by the 1-candle SOM

the maps were labeled. The sets of known formations were supplied on nets' inputs. The map of the 1-candle SOM is presented in fig. 3. The net grouped similar patterns on the map placing formations of market's hesitation (e.g. the "doji" and "spinning tops") in the middle of it, bullish patterns in the top part of the map (ideal "hammer" in the upper left corner), while bearish patterns in the lower part of the map (ideal "shooting star" in the lower left corner). The left hand side of the map contains patterns with longer wicks, while the right hand side - candles built mostly by their bodies.

6 A Sample Trading System

The maps described above provide us with coherent visual maps of candlestick formations. They also allow us to classify and code the analyzed window of timeseries for further steps of decision algorithms. The system presented in this section uses the outputs of SOMs analyzing 1-, 2- and 3-candle windows, translated to the direction of the recognized pattern (-1 for bearish, +1 for bullish and 0 for neutral), along with the value of a short term exponentially weighted moving average of the price, as inputs to a feedforward net learning trading decisions. These decision were set arbitrarily by a human trader using a trend following strategy and visual analysis of the chart. The analyst "knew the future", that is saw both the left and right hand side of the analyzed time point on the chart, therefore his decisions were correct by definition. The multilayer perceptron network is very simple - built of 4 sigmoid units in the hidden layer and one linear in the output layer. The structure of the neural decision part of the trading system is presented in fig. 4.

First, the three SOMs were trained to recognize patterns, as described earlier. Then the feedforward net was trained on the same period of time with 300 desired trading signals, with standard backpropagation algorithm. 70% of data was used for training, while 15% for validating and 15% for testing. The net

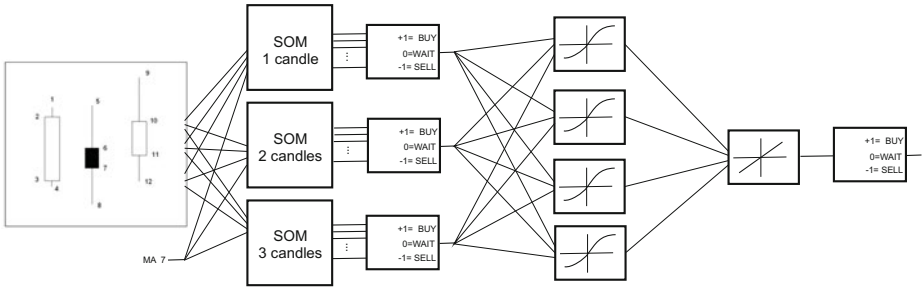


Fig. 4. The structure of the neural decision part of the trading system



Fig. 5. Results of system backtest on EURUSD - account equity

Table 1. The results of trading system's backtesting on historical data

backtest period	13.10.2006 - 11.06.2009			
	trades	profit	in one trade	
profit	25	90 901.50 USD	max profit	16 058.00 USD
loss	159	-51 842.00 USD	max loss	2 454.00 USD
all	184	39 059.50 USD	average profit	3 606.06 USD
total profit		390.59 %	average loss	326.05 USD

reached $1.02913 \cdot 10^{-13}$ value for the mean squared error and the correlation coefficient of 0.999999. These values illustrate only the performance of the neural decision subsystem. However, good decisions (entry points for transactions) may be worthless if used with wrong strategy of securing an open trade and its profit. For a full view of system's profitability on the market, it should be tested on-line on real market and its performance should be measured with such characteristics as: gross and net profit, profit factor, number and value of profit and loss transactions, maximal drawdown etc. In our case, we used simple but strict money and risk management rules, so that their influence did not dominate the role of the neural decision part.

The system was tested on historical data (EURUSD, 1 hour interval) in Meta-trader's strategy tester. The results of the test are presented in fig. 5 and in table 1. Testing period of 2.5 years covered bullish and bearish markets, and

periods of stagnation. Starting account balance was 10000 USD, with the leverage of 1:100. Money and risk management strategy defined the following rules: 1 open trade at a time (stop and reverse system), maximum of 30% of capital involved in margin for one trade and maximum of 2% of capital exposed to risk in one trade. The total profit achieved was 390%. The results show both the effectiveness of the system and the role of proper money and risk management. The number of profit trades was 6 time smaller than the number of loss trades, while the value of the former exceeded the value of the latter by 1.5. This was due to the fact that loss trades were closed quickly at stop losses, while profit trades were kept for a long time. The system presented stable and monotone performance, without large drawdowns in capital. The proportion of profit to loss trades does not deprecate the effectiveness of neural decision module. It was able to point out the most important moments for long lasting and profitable trades, while the loss trades happened mostly due to delayed market reaction to some formations (followed by a so-called "second chance"). In that cases good trades open too early may be closed on strict stop losses due to market hesitation. More sophisticated strategies including re-entries and multiplying long profitable trades may provide further improvement to the simple strategy presented here.

7 Conclusions

The problem of efficient processing of forex market data is addressed in the paper. It was shown that the Japanese candles, appropriately coded, provide a superb method of presenting timeseries data to neural nets. Self organizing maps provide clear coherent maps of candlestick formations for the analyzed instrument and timeframe. The classification performed by the maps is simple and efficient and may be used as input for the final trading decision algorithm. An example of a profitable trading system based on that methodology was presented and analyzed.

References

1. McNelis, P.D.: *Neural Networks in Finance: Gaining predictive edge in the market*. Elsevier, Amsterdam (2005)
2. Nison, S.: *Japanese Candlestick Charting Techniques*, 2nd edn. Prentice Hall Press, Englewood Cliffs (2001)
3. Li, S.-T., Kuo, S.-C.: Knowledge discovery in financial investment for forecasting and trading strategy through wavelet-based SOM networks. *Expert Systems with Applications* 32(2), 935–951 (2008)
4. Vanstone, B., Finnie, G.: An empirical methodology for developing stockmarket trading systems using artificial neural networks. *Expert Systems with Applications* 36(3), 6668–6680 (2009)

Measures for Comparing Association Rule Sets

Damian Dudek

Department of Software Development and Internet Technologies,
The University of Information Technology and Management “Copernicus”,
ul. Inowroclawska 56, 53-648 Wroclaw, Poland
ddudek@wsiz.wroc.pl

Abstract. Most experimental methods for evaluating algorithms of association rule mining are based solely on quantitative measures such as correlation between minimum support, number of rules or frequent itemsets and data processing time. In this paper we present new measures for comparing association rule sets. We show that observing rule overlapping, support and confidence in two compared rule sets helps evaluate algorithm quality or measure uniformity of source datasets.

Keywords: association rules, evaluation methods, comparing rule sets, data set uniformity, data source stability.

1 Introduction

Methods for *association rule mining* (ARM) have gained great interest in the field of machine learning as useful tools for analyzing databases and building models in various application domains. Since *association rules* (AR) were introduced by Agrawal et al. [1], there have been a truly huge body of research on frameworks and algorithms of association discovery and maintenance, including [1]: Apriori [2], Borders [3], estDec [5], EDUA [12] and RMAIN [6]. Experimental evaluation is a commonly accepted method for comparing different ARM methods with each other. For this purpose a range of measures are used, including: execution time, I/O cost, memory usage, size of candidate itemsets, number of frequent itemsets or rules with respect to minimum support, and number of analysis passes through data. These measures provide good insight into efficiency of the compared methods. However, none of them give us any idea of qualitative results of ARM, namely, what rules or frequent itemsets are returned and what are their statistical measures (e.g. support, confidence). Alas, such questions are commonly omitted in comparative studies of ARM methods. Although this approach can be justified for accurate algorithms using similar mining procedures (e.g. *Apriori-gen*), it is not acceptable for algorithms with different rule discovery techniques and, even more importantly, for approximate methods with trade-off between efficiency and precision. While to date there have been many measures of assessing performance (see the list above), hardly any ones have

¹ For an extensive and thorough overview of ARM methods see [6].

been proposed for computing accuracy of AR mining or maintenance. In order to fill this gap, in this paper new measures for comparing sets of association rules are introduced. They can be used to evaluate ARM precision, compare results of rule mining or maintenance for different algorithms, and investigate important properties of source data.

The remainder of the paper is organized as follows. The formal model of association rules is presented briefly in section 2. In the next section basic measures for comparing sets of AR are introduced. Then, in section 4 we present more complex methods, which are based upon these foundations and can be applied for: (i) qualitative evaluation of incremental ARM methods, (ii) testing uniformity of a dataset, and (iii) tracking stability of a sequential data source.

2 The Model of Association Rules

In this section the classical model of association rules is presented, which is necessary for further definitions of rule comparison measures. The provided formalism is based on the models by Agrawal et al. [1,2] and Goethals [7].

An *association rule* $X \Rightarrow Y$ represents dependency between two sets of attributes: the *antecedent* X and *consequent* Y . Within the shopping cart domain a rule *cheese* \wedge *eggs* \wedge *tomatoes* \Rightarrow *potatoes* means that customers, who put cheese, eggs and potatoes together into their shopping carts, *often* buy potatoes as well. More formally, consider a set of attributes $U = \{I_1, I_2, \dots, I_n\}$ representing the universe of discourse. Each attribute $I_j \in U$, for $j \in [1; n]$, $n \in \mathbb{N}$ is assigned a domain $\{0, 1\}$. A *transaction* is a vector $t = (tid, i_1, i_2, \dots, i_m)$, where $tid \in \mathbb{N}$ is an identifier, and $i_k \in \{0, 1\}$, for $k \in [1; m]$, is value of an attribute I_k in a transaction t , denoted by $I_k(t)$. A *transaction database* D is a set of transactions t . We say that a transaction t *satisfies* a set of attributes $X \subseteq U$, denoted by $t \vdash X$, iff each attribute $I_j \in X$, $j \in [1; |X|] \cap \mathbb{N}$ takes value $I_j(t) = 1$. The *frequency* of an attribute set X in a transaction database D is the fraction $freq(X, D) = |\{t : t \vdash X\}|/|D|$. A *frequent itemset* is a set of attributes $X \subseteq U$, such that $freq(X, D) \geq \sigma$, where $\sigma \in [0; 1]$ is the *minimal support threshold*. An *association rule* is an expression $X \Rightarrow Y$, where $X \subset U$, $Y \subset U$ and $X \cap Y = \emptyset$. The *support* $sup(r, D)$ of a rule $r : X \Rightarrow Y$ in a transaction database D is equal to $freq(X \cup Y, D)$. The support tells how representative a given rule is in a considered dataset. The *confidence* $con(r, D)$ of a rule $r : X \Rightarrow Y$ in a transaction database D is defined as the fraction: $con(X, D) = sup(r, D)/freq(X, D)$. The *con* measure represents reliability of a given rule. We denote by $R(D, \sigma, \gamma)$ a set of all the association rules r , such that $sup(r, D) \geq \sigma$ and $con(r, D) \geq \gamma$, where D is a transaction database, $\sigma \in [0; 1]$ is a minimum support threshold and $\gamma \in [0; 1]$ is a minimum confidence threshold.

Some researchers point at limited usefulness of the confidence measure as it ignores the frequency of the rule consequent. In order to amend this drawback other measures have been proposed. The *interest* (also called *lift*) measure of an association rule $r : X \Rightarrow Y$ within a transaction database D is defined as the fraction [10]: $interest(r, D) = freq(X \cup Y, D)/(freq(X, D) \cdot freq(Y, D)) =$

$con(r, D)/freq(Y, D)$. Values of the interest range from 0 up. The *match* is another measure of effectiveness of association rules, introduced by Wei et al. [11], defined as the fraction: $match(r, D) = (freq(X \cup Y, D) - freq(X, D) \cdot freq(Y, D)) / (freq(X, D)(1 - freq(X, D)))$. Its values range from -1 to 1 .

3 Basic Measures for Comparing Rule Sets

In this chapter we propose measures for comparing rule sets with respect to general overlapping of attribute sets, support, and confidence. We do not deal with the rule interest and match as they can be derived from more elementary measures, mentioned above. First we introduce the *rule_{overlap}* measure, which shows the degree of intersection for two rule sets.

Definition 1. *The rule overlapping ratio of rule sets $R_1 \equiv R(D_1, \sigma_1, \gamma_1)$ and $R_2 \equiv R(D_2, \sigma_2, \gamma_2)$ is defined by the following equation [6]:*

$$rule_{overlap}(R_1, R_2) = \frac{|R_1 \cap R_2|}{|R_1 \cup R_2|}.$$

The *rule_{overlap}* is very useful for making general comparisons of rule sets and investigating properties of datasets or data sources (see section 4). However, often more accurate measures are necessary for thorough testing of rule set similarity. The *sup_{diff}* shows average rule support difference between two rule sets.

Definition 2. *The average support difference of rule sets $R_1 \equiv R(D_1, \sigma_1, \gamma_1)$ and $R_2 \equiv R(D_2, \sigma_2, \gamma_2)$ is defined by the following equation [6]:*

$$sup_{diff}(R_1, R_2) = \frac{\sum_{r \in R_1 \cap R_2} |sup(r, D_1) - sup(r, D_2)| + |R_1 \cup R_2| - |R_1 \cap R_2|}{|R_1 \cup R_2|}.$$

The *sup_{diff}* measure is quite similar to the *average support error* of a set of itemsets R_2 with respect to a set of itemsets R_1 , denoted by $ASE(R_2|R_1)$, which was proposed by Chang and Lee [5]. However, *sup_{diff}* is a symmetric measure, i.e. $sup_{diff}(R_1, R_2) = sup_{diff}(R_2, R_1)$ for all sets R_1 and R_2 , while $ASE(R_2|R_1) = ASE(R_1|R_2)$ only if $|R_1| = |R_2|$. Moreover, *sup_{diff}* is more restrictive as it always maximizes the support difference up to 1 for rules, that do not overlap with each other, while ASE treats non-overlapping itemsets more gently, summing up their support, which is usually less than 1.

Definition 3. *The average confidence difference of rule sets $R_1 \equiv R(D_1, \sigma_1, \gamma_1)$ and $R_2 \equiv R(D_2, \sigma_2, \gamma_2)$ is defined by the following equation [6]:*

$$con_{diff}(R_1, R_2) = \frac{\sum_{r \in R_1 \cap R_2} |con(r, D_1) - con(r, D_2)| + |R_1 \cup R_2| - |R_1 \cap R_2|}{|R_1 \cup R_2|}.$$

Example. Below we demonstrate how the proposed measures work. Assume the following rule sets $R_1 \equiv R(D_1, \sigma_1, \gamma_1)$ and $R_2 \equiv R(D_2, \sigma_2, \gamma_2)$.

R_1	r_1 : $milk \wedge raisins \wedge yogurt \Rightarrow cereals$	$sup(r_1, D_1) = 0.05$; $con(r_1, D_1) = 0.62$
	r_2 : $candies \wedge chocolate \wedge nuts \Rightarrow wafers$	$sup(r_2, D_1) = 0.08$; $con(r_2, D_1) = 0.54$
	r_3 : $bread \wedge butter \Rightarrow cheese$	$sup(r_3, D_1) = 0.06$; $con(r_3, D_1) = 0.78$
	r_4 : $apples \wedge bananas \wedge cereals \Rightarrow sugar$	$sup(r_4, D_1) = 0.10$; $con(r_4, D_1) = 0.59$
	r_5 : $flour \wedge groats \wedge salt \Rightarrow rice$	$sup(r_5, D_1) = 0.04$; $con(r_5, D_1) = 0.83$
	r_6 : $cherries \wedge strawberries \Rightarrow raspberries$	$sup(r_6, D_1) = 0.12$; $con(r_6, D_1) = 0.67$

R_2	r_2 : $candies \wedge chocolate \wedge nuts \Rightarrow wafers$	$sup(r_2, D_2) = 0.15$; $con(r_2, D_2) = 0.68$
	r_5 : $flour \wedge groats \wedge salt \Rightarrow rice$	$sup(r_5, D_2) = 0.06$; $con(r_5, D_2) = 0.91$
	r_7 : $bananas \wedge grapefruits \Rightarrow lemons$	$sup(r_7, D_2) = 0.07$; $con(r_7, D_2) = 0.79$
	r_8 : $cucumbers \wedge dill \wedge garlic \Rightarrow salt$	$sup(r_8, D_2) = 0.11$; $con(r_8, D_2) = 0.56$

There are 2 overlapping rules (r_2, r_5) and 8 unique rules $(r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8)$. Hence, $rule_{overlap}(R_1, R_2) = 2/8 = 0.25$, $sup_{diff}(R_1, R_2) = (|0.08 - 0.15| + |0.04 - 0.06| + 8 - 2)/8 = (0.07 + 0.02 + 6)/8 \approx 0.76$, and $con_{diff}(R_1, R_2) = (|0.54 - 0.68| + |0.83 - 0.91| + 8 - 2)/8 = (0.14 + 0.08 + 6)/8 \approx 0.78$.

4 Applications of the Rule Comparison Measures

There are at least three significant applications of the proposed measures for association rule sets comparison: (i) qualitative evaluation of incremental ARM algorithms, especially the approximate ones, (ii) analyzing rule distribution uniformity within data sets, and (iii) testing stability of ARM data sources.

4.1 Qualitative Evaluation of Incremental ARM Methods

As mentioned in the Introduction, it can be not enough to use common quantitative measures of efficiency for thorough experimental evaluation or comparison of ARM algorithms (especially incremental ones). The measures presented in this paper can be useful for testing accuracy of rule mining and maintenance. For this purpose we propose the following experiment strategy². (1) Select an AR mining or maintenance method as the reference point for comparisons with the examined algorithm. (2) Choose a test dataset for experiments. Within the dataset select the initial partition of transactions, which are to be processed in the first run and plan subsequent increments of input data for the examined algorithm (Fig. 1): additions of transactions (further partitions of the test dataset) or deletions (if applies to the investigated algorithm). (3) Perform AR mining using the examined algorithm for the initial partition and then mining or maintenance runs for subsequent data increments. After every run store separately the resulting AR set. (4) Perform similar AR mining or maintenance runs using

² Experimental studies using this method can be seen in the previous work [6].

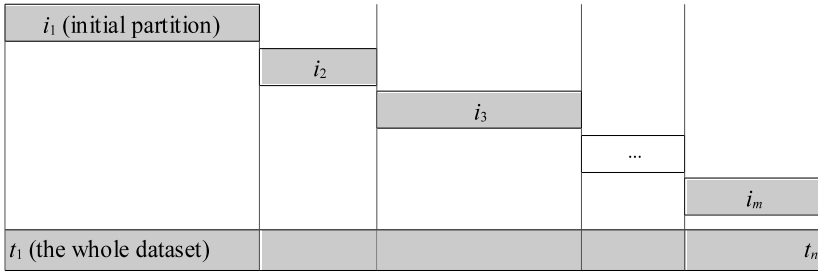


Fig. 1. Performing AR mining (or maintenance) for subsequent data partitions

the reference algorithm for the same data. Store separately the resulting AR sets.
 (5) Compare the corresponding AR sets using: $rule_{overlap}$, sup_{diff} , and con_{diff} .
 (6) Visualize the results for subsequent increments using tables or data charts.

4.2 Measuring Uniformity of a Dataset

In some applications of ARM it is not enough to find a set of rules satisfying given criteria, but it is also important to know distribution of these rules within a tested dataset. If transactions, which support an association rule, are distributed uniformly along a dataset, we can conclude that the rule is stable and valid as related to the whole long-term data. In the shopping cart scenario we would expect quite uniform distribution of rules in collected transactions of a grocery shop, where many products are bought on a daily basis, disregarding the time of the year. The situation can change significantly for a sporting store, which is subject to season fluctuations. Then most association rules are not distributed uniformly and they have more local meaning – they can be found only in some partitions of long-term data. Why does the dataset uniformity matter? Classical association rules (together with their support and confidence) can turn out to be unreliable, if discovered in strongly not uniform data as they may represent patterns, which are local or repeatable in long-term cycles. Hence, a better choice for analyzing such datasets can be mining partitions of data separately or using sequential rules (out of the scope of this paper).

For the purpose of testing uniformity of rule distribution within datasets, below we propose the *rule comparison matrix* M_{RC} and the *RDU* measure.

Definition 4. Let a transaction database $D = t_1, t_2, \dots, t_n$, $n \in \mathbb{N}$, be given. Let σ and γ be defined as before. Let the block size $b_s \in [1; |D|] \cap \mathbb{N}$, such that $|D| \bmod b_s = 0$, be given. The rule comparison matrix M_{RC} for the database D with respect to b_s , σ , and γ is defined as follows:

$$M_{RC}(D, \sigma, \gamma, b_s) = \begin{pmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & \ddots & \vdots \\ a_{k1} & \cdots & a_{kk} \end{pmatrix}, a_{ij} = rule_{overlap}(R(p_i, \sigma, \gamma), R(p_j, \sigma, \gamma)),$$

where $p_l = \{t_{d+1}, t_{d+2}, \dots, t_{d+b_s}\} \subseteq D$, $d = b_s(l - 1)$, $l \in [1; k] \cap \mathbb{N}$, $k = |D|/b_s$.

Notice, that M_{RC} is a symmetric square matrix $k \times k$, so we do not have to compute $rule_{overlap}$ for all k^2 pairs of partitions, but essentially only for elements below or above the matrix diagonal, while all values at the diagonal are equal 1.

Definition 5. Let a transaction database D and the rule comparison matrix $M_{RC}(D, \sigma, \gamma, b_s) = (a_{ij})$ for $i, j \in [1; |D|/b_s] \cap \mathbb{N}$ be given. The rule distribution uniformity (RDU) within D with respect to M_{RC} is defined as follows:

$$RDU(D, M_{RC}) = \frac{1}{m} \sum_{i>j} a_{ij}, \text{ where } m = \frac{|D|}{2b_s} \left(\frac{|D|}{b_s} - 1 \right).$$

Experimental example. In order to illustrate the M_{RC} matrix and the RDU measure, two datasets were analyzed under the *APS Incremental Learning* test bed application [6]. The tests were run on the x86 machine (892.5 MHz CPU, 752 MB RAM, NTFS, OS MS Windows Server 2003).

The first dataset was a synthetic T10.I4.D100K dataset (first 20,000 transactions), which was prepared using the IBM generator program [9] and downloaded from the repository of the University of Helsinki [8]. The $M_{RC}(D_1, \sigma_1, \gamma_1, b_s)$ matrix for the T10.I4.D100K dataset (Fig. 2, left) was elaborated using the following parameter values: $\sigma_1 = 0.01$; $\gamma_1 = 0.50$; $b_s = 2000$. Color saturation is related to values of elements. We can see that under these settings the dataset is rather irregular as rules found in different transaction partitions have little in common. The rule distribution uniformity $RDU(D_1, M_{RC})$ is only about 0.11.

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
p_1	1.00	0.46	0.33	0.03	0.03	0.50	0.01	0.03	0.03	0.02
p_2	0.46	1.00	0.46	0.01	0.09	0.56	0.01	0.05	0.09	0.02
p_3	0.33	0.46	1.00	0.02	0.09	0.37	0.01	0.04	0.09	0.01
p_4	0.03	0.01	0.02	1.00	0.06	0.01	0.00	0.02	0.05	0.00
p_5	0.03	0.09	0.09	0.06	1.00	0.01	0.00	0.12	0.80	0.05
p_6	0.50	0.56	0.37	0.01	0.01	1.00	0.01	0.01	0.01	0.02
p_7	0.01	0.01	0.01	0.00	0.00	0.01	1.00	0.06	0.00	0.00
p_8	0.03	0.05	0.04	0.02	0.12	0.01	0.06	1.00	0.12	0.01
p_9	0.03	0.09	0.09	0.05	0.80	0.01	0.00	0.12	1.00	0.04
p_{10}	0.02	0.02	0.01	0.00	0.05	0.02	0.00	0.01	0.04	1.00

IBM T10.I4.D100K

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
p_1	1.00	0.43	0.25	0.16	0.13	0.22	0.27	0.17	0.20	0.18
p_2	0.43	1.00	0.39	0.18	0.14	0.32	0.31	0.21	0.24	0.18
p_3	0.25	0.39	1.00	0.17	0.15	0.22	0.29	0.19	0.18	0.16
p_4	0.16	0.18	0.17	1.00	0.58	0.16	0.19	0.13	0.11	0.10
p_5	0.13	0.14	0.15	0.58	1.00	0.14	0.16	0.14	0.09	0.08
p_6	0.22	0.32	0.22	0.16	0.14	1.00	0.61	0.19	0.19	0.14
p_7	0.27	0.31	0.29	0.19	0.16	0.61	1.00	0.23	0.21	0.16
p_8	0.17	0.21	0.19	0.13	0.14	0.19	0.23	1.00	0.27	0.12
p_9	0.20	0.24	0.18	0.11	0.09	0.19	0.21	0.27	1.00	0.29
p_{10}	0.18	0.18	0.16	0.10	0.08	0.14	0.16	0.12	0.29	1.00

Brijs Retail

Fig. 2. The rule comparison matrices M_{RC} for the first 20,000 transactions of the datasets: synthetic IBM T10.I4.D100K (left) and real Brijs Retail (right)

Another series of experiments were conducted using the real shopping cart data from a Belgian supermarket (prepared by Tom Brijs [4]), containing 88,162 transactions and referring to 16,470 unique products. As before, the first 20,000 transactions were used. This time the $M_{RC}(D_2, \sigma_2, \gamma_2, b_s)$ matrix was generated for the parameter values: $\sigma_2 = 0.02$; $\gamma_2 = 0.06$; $b_s = 2000$. Apparently, with $RDU(D_2, M_{RC})$ reaching about 0.21, rule distribution within the Brijs Retail dataset is more uniform (Fig. 2, right), than in T10.I4.D100K.

Computing RDU for Large Datasets Using Sampling. Determining the M_{RC} matrix and the RDU measure for very large sets of transactions, especially for fine granularity (small block size), can be computationally intensive, often beyond an acceptable level. Then sampling techniques can be used to approximate RDU . A simple method we propose in this paper is decreasing the dimension of the M_{RC} through selecting data partitions (blocks), within which rules are to be discovered and compared. For this purpose we build a set of sampling partitions $S_p = \{p_i \subseteq D : |p_i| = b_s, i \in [1; |D|/b_s] \cap \mathbb{N}\}$. For instance, consider a database D containing 100,000 transactions and divided into 100 partitions p_1, p_2, \dots, p_{100} each of the size $b_s = 1000$. We can choose to sample every tenth partition beginning at p_1 , hence $S_p = \{p_1, p_{11}, p_{21}, \dots, p_{91}\}$, the M_{RC} will have the dimension of 10×10 and $rule_{overlap}$ will be effectively computed for 45 pairs of partitions, instead of 4950 pairs for the initial matrix 100×100 (without sampling). Of course, while computing RDU using the M_{RC} matrix with sampling partitions, we need to have the database size reduced to $|D'| = b_s|S_p|$.

4.3 Tracking Stability of a Sequential Data Source

The M_{RC} matrix and the RDU measure are useful for investigating uniformity of datasets, which are whole available at the moment of analysis. However, in many applications transactions come incrementally from sequential data sources and thus can not be processed using the M_{RC} matrix. In that case, instead of computing mutual similarity of all data partitions, we propose using the *cumulative rule stability (CRS)* measure, which compares the previous data block with the last increment and computes average $rule_{overlap}$ for adjacent partitions with respect to the whole examined sequence of transactions. The CRS measure is intended for iterative calculations as subsequent data portions arrive.

Definition 6. Let D, σ and γ be defined as before. Let two sets of transactions be given: the previous partition $p_i = \{t_1, t_2, \dots, t_m\} \subset D$ and the recent partition $p_{i+1} = \{t_{m+1}, t_{m+2}, \dots, t_n\} \subset D, i, m, n \in \mathbb{N}, i \geq 1$. The cumulative rule stability (CRS) of D for the i -th iteration is defined as follows:

$$CRS_i(D, \sigma, \gamma) = \begin{cases} rule_{overlap}(R(p_i, \sigma, \gamma), R(p_{i+1}, \sigma, \gamma)), & \text{if } i = 1, \\ \frac{1}{i}((i - 1)CRS_{i-1}(D, \sigma, \gamma) + rule_{overlap}(R(p_i, \sigma, \gamma), R(p_{i+1}, \sigma, \gamma))), & \text{if } i > 1. \end{cases}$$

Notice, that in Definition 6 the transaction database D is a concept used for defining format and properties of transactions, which arrive continuously in partitions. It can not be considered a standard dataset as in previous definitions.

Example. Using the experimental results for the Brijs dataset (Fig. 2, right), we compute iteratively the $CRS_i(D, \sigma, \gamma)$ for $i = 1, 2, \dots, 9$, moving through the 9 subsequent pairs of adjacent partitions $(p_1, p_2), \dots, (p_9, p_{10})$, and we get the values, respectively: 0.43, 0.41, 0.33, 0.39, 0.34, 0.39, 0.36, 0.35, 0.35.

5 Conclusions

The important contribution of this work to the research on ARM is proposing methods for qualitative evaluation of mining results in addition to common efficiency measures (execution time etc.). The introduced measures can be useful for evaluating precision of incremental ARM methods (especially approximate ones), investigating distribution of association rules along a dataset and continuously tracking stability of a sequential data source. They can help explain why an ARM algorithm performs better for one dataset or worse for another one, and decide, whether or not the classical AR model is appropriate for processing given data. Currently, the proposed measures enable comparing mining results, which are discovered under the same thresholds of minimal support and confidence. An interesting issue for further research is developing measures for comparing AR sets, which are found using different parameter values.

Acknowledgments. This research was partially funded by the European Union and the Polish Ministry of Science and Higher Education under the project POKL.04.01.01-00-295/08-00.

References

1. Agrawal, R., Imielinski, T., Swami, A.: Mining association rules between sets of items in large databases. In: Proceedings of ACM SIGMOD Conference on Management of Data (SIGMOD 1993), pp. 207–216 (1993)
2. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules. In: Proc. of the Twentieth International Conf. on Very Large Databases, Santiago, Chile (1994)
3. Aumann, Y., Feldman, R., Lipshtat, O., Manilla, H.: Borders: An Efficient Algorithm for Association Generation in Dynamic Databases. *Journal of Intelligent Information Systems* 21, 61–73 (1999)
4. Brijs, T.: Retail Market Basket Analysis: A Quantitative Modelling Approach. PhD Thesis, Department of Applied Economic Sciences, Universiteit Hasselt (Diepenbeek, Belgium) (2002)
5. Chang, J.H., Lee, W.S.: Finding Recent Frequent Itemsets Adaptively over Online Data Streams. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 487–492 (2003)
6. Dudek, D.: RMAIN: Association rules maintenance without reruns through data. *Information Sciences* 179, 4123–4139 (2009)
7. Goethals, B.: Efficient Frequent Pattern Mining. PhD thesis, Transnational University of Limburg, Diepenbeek, Belgium (2002)
8. Goethals, B.: Frequent Itemset Mining Implementations Repository, University of Helsinki (2004), <http://fimi.cs.helsinki.fi>
9. IBM Almaden Research Center: Dataset generator, <http://www.almaden.ibm.com>
10. Silverstein, C., Brin, S., Motwani, R.: Beyond Market Baskets: Generalizing Association Rules to Dependence Rules. *Data Mining and Knowl. Discov.* 2, 39–68 (1998)
11. Wei, J.-M., Yi, W.-G., Wang, M.-Y.: Novel measurement for mining effective association rules. *Knowledge-Based Systems* 19, 739–743 (2006)
12. Zhang, S., Zhang, J., Zhang, C.: EDUA: An efficient algorithm for dynamic database mining. *Information Sciences* 177, 2756–2767 (2007)

Distributed Data Mining Methodology for Clustering and Classification Model

Marcin Gorawski and Ewa Pluciennik-Psota

Silesian University of Technology,
Institute of Computer Science,
Akademicka 16,44-100 Gliwice, Poland
{Marcin.Gorawski,Ewa.Pluciennik}@polsl.pl

Abstract. Distributed computing and data mining are nowadays almost ubiquitous. Authors propose methodology of distributed data mining by combining local analytical models (built in parallel in nodes of a distributed computer system) into a global one without necessity to construct distributed version of data mining algorithm. Different combining strategies for clustering and classification are proposed and their verification methods as well. Proposed solutions were tested with data sets coming from UCI Machine Learning Repository.

1 Introduction

Data mining is now essential part of data processing for most of scientific and commercial organizations. Distributed data processing becomes not only a method of improving performance but a necessity. Distributed data mining can be conducted in a few ways [1]. Most popular are meta-learning [2] and distributed versions of local data mining algorithms [3]. We propose methodology of combining local analytical models (built in parallel in nodes of a distributed computer system) into a global one without necessity of construction distributed version of data mining algorithm and with compact global model form.

Basic assumptions for proposed solution are (i) a complete horizontal data fragmentation and (ii) a model form understood by a human being. Building global data model consists of two stages. In the first one local models are built in a parallel manner. Second one consists of combining these models into a global data picture. We propose some combining methods for local clustering and classification models. In case of clustering we propose techniques for combining clusters represented by attributes' histograms. In case of classification we propose techniques for process individual classification rules.

2 Distributed Data Mining Methodology

In [4] we have proposed distributed data mining models which can be divided into three steps: choosing and implementing a local data mining algorithm, selecting a data mining model quality measure and working out combining strategy and

its quality measure. We have proposed two kinds of verifying values. First one is a quality of model built using all data with local algorithm (control model). It is assumed that the model built this way is the most accurate [5] so it can be stated that a global model with approximate accuracy is very good. On the other hand we need to pinpoint some additional quality values which can help to set the lower limit of the quality. We call these values supplementary control values (SCV).

As a main SCV we have chosen average quality of local models. In case of classification we also consider a quality of global model built as a meta-classifier based on local model voting. If it comes to clustering we have chosen quality of the summary global model (model constructed as a set of all clusters from local models) as lower limit of the global model quality.

For classification models as a quality value we use the accuracy (the main assumption is of course the same testing set for all models). As for clustering we have decided to use a category utility measure [6] as models quality measure.

2.1 Combining Strategies for Clustering

Combining local clustering models boils down to finding a way of local clusters connection (from different models) which has maximal quality measure. Local clusters are built on different partition of the data set. This is different solution then clusters ensemble where we have to deal with different partitions of the same set [7].

In our case instead of checking all clusters combinations (from different local models) and choose the best one ($(n!)^{m-1}$ possible combinations of n clusters from m local models) we have decided to choose clusters to join on the basis of some distance (or similarity) measure. In our solution we join clusters according to the following schema:

1. Choose primary model (PM) from local models.
2. For all clusters from primary model find the closest cluster (from remaining models) and join them together.
3. Repeat step 2 until all local models' clusters are joined.

In this schema the most important problem is to choose primary model which in fact determines a number of clusters in a global model. If we have to deal with situation where all local models have the same number of clusters we can randomly choose a primary model. Number of clusters in the global one is the same as in local models. This is the simplest case which can result from data nature or local clustering algorithm (when algorithm has a parameter determining number of clusters).

If local clustering algorithm hasn't got such parameter we have to deal with a different number of clusters in local models. Therefore number of clusters in the global model depends on which local model we decide to choose as a primary model. We did not consider any intervention if it comes to the number of clusters in the global model such as averaging number of clusters from local models. So in fact we need to consider three cases while joining two local models (at given

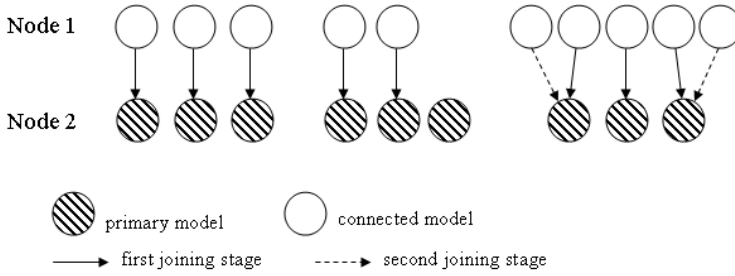


Fig. 1. Joining clusters cases

moment we join two models: primary and so called connected model) as shown in Fig. 1. In the first one we join one to one. In the second one some clusters in the primary model are not joined. In the third one some clusters in the connected model are not joined. In this case we need to conduct second stage of joining (all clusters in the connected model should be joined). Considering described above situations we have formulated the following strategies of choosing primary model:

1. For the same number of clusters in local models we choose randomly primary model- equivalent random strategy with two variants EBM (equal best match) and EFM (equal first match). In EFM strategy for currently processed cluster from connected model we can search for the closest cluster from primary model apart from those clusters that are already joined for current connected model. In EBM strategy first we check all distances between clusters from primary and connected models and then join the closest ones.
2. For different number of clusters in local models we use following strategies:
 - MAX - as PM we choose local model with the maximal number of clusters,
 - MIN - as PM we choose local model with the minimal number of clusters,
 - CTR - as PM we choose local model with the central number of clusters,
 - MF - as PM we choose local model with the most frequent number of clusters.

MF strategy is a hybrid strategy because it combines cases from CTR, MIN, MAX strategies.

If it comes to form of a cluster description (cluster label) we have decided to use histograms of attributes used as model dimensions. For now we have assumed discreet values of attributes. Histogram description is simple, compact and universal. Histogram description can be obtained always regardless of cluster description generated by local clustering algorithm. For histogram cluster representation we can also always evaluate a category utility. We have defined the cluster representation as follows: For n -dimensional space with attributes A_1 to A_n taking values from sets, adequately a_1, \dots, a_n cluster Cl has a form:

$$\begin{aligned}
 Cl &= \{H(A_i)\}, \text{ where } i \in (1..n) \\
 H(A_i) &= H^j(A_i), \\
 \text{where } H^j(A_i) &= |Cl_{A_i=a_{ij}}| * 100 / |Cl_{A_i}|, j \in (1..|a_i|)
 \end{aligned} \tag{1}$$

Histogram for a given attribute $H(A_i)$ represents percentage of attribute values $H^j(A_i)$ for object belonging to the cluster.

Another thing we needed to consider is a cluster distance measure. The simplest measure is L1 defined as a sum of differences between particular histogram buckets [8] and Euclidean measure where those differences are powered. As a distance measure for clusters described by attribute histograms we have decided to choose the following three variants:

$$\begin{aligned} dist(Cl_1, Cl_2) &= \sum_{i=1}^n dist(H_1(A_i), H_2(A_i)) \\ dist(Cl_1, Cl_2) &= \overset{avg}{i} dist(H_1(A_i), H_2(A_i)) \\ dist(Cl_1, Cl_2) &= \overset{avg}{i} w_i * dist(H_1(A_i), H_2(A_i)) \end{aligned} \quad (2)$$

The first one is a sum of distances between particular attribute histograms. The second one is the average of these distances and the third one is a weighted average. For histogram distances we have used L1 and E_{cl} (Euclidean distance) measures.

2.2 Combining Strategies for Classification

We have assumed data model in form of a rule set. We have defined three types of rules which can appear in a global classification model. The first ones are conflicting rules. Their coverage can have common parts and conclusions are different. The second ones are subrules - rules which coverage is a subset of coverage of another rule with the same conclusion. The third ones are friendly rules - rules which coverage can have common parts with rules with the same conclusion. Rule's conclusion is a value of a class to which object that meets premise condition will be classified. These kinds of rules have to be taken into consideration during combining local models' process and applying global model. Full algorithms and definitions can be found in [9].

3 Tests

For tests we have used three sets from the UCI Machine Learning Repository [10]. *Adult* set (census data) has two valued class and 48 842 records. *Votes* set (congressional voting results) has two valued class and 435 records. *Cars* set (cars' features like price, security level, etc.) has four valued class and 1728 records. Every set was divided into a test set and a training set (used for a global control model). Then training set was divided into three sets with similar sizes (7000, 13000, 12561 number of records for *adult* set; 3 x 100 for *votes* set and 3 x 400 for *cars* set) used for local models. We have also built global control models for both classification and clustering and also global summary models for clustering and global models based on voting for classification.

All created models for classification were then tested with the test sets. Local classification models were built using the ID3 algorithm [11]. For local clustering models we have used non-parameter COBWEB algorithm [12] and O-Cluster algorithm [13] which needs maximal number of clusters as one of the parameters.

For O-Cluster algorithm we have also used another test set *cen_big* (two valued class, record number: 199 523, version of *adult* set) for additional performance test. Then local models were joined using all possible combination of strategies with clusters' distances and histogram distances described above (2.1). In weighted clusters distance measure we have used $w_i = 1/|a_i|$.

3.1 Clustering Tests Results

Table 1 and 2 contain collation of the best (max column) and the worst (min column) category utility for combining clusters' strategies with respect to the control model, average of the local models and summary model category utility. Values greater than 100 denote that a given model category utility is better than (for example) a control model category utility (values equal to 100 mean the same value of category utility for both models).

Clustering Tests for COBWEB as Local Clustering Algorithm. Local cluster models and control model built using COBWEB algorithm had two clusters for *adult* and *votes* set. For *cars* set control model had 3 clusters and local models had 3, 3 and 4 clusters.

Table 1. Comparison of combining clusters' strategies for COBWEB algorithm

Set	Category Utility percentage with respect to					
	control model		local models average		summary model	
	min	max	min	max	min	max
adult	99,92	99,92	100,05	100,05	299,69	299,69
votes	97,83	97,83	98,25	98,25	285,63	285,63
cars	83,39	99,50	87,74	104,69	149,07	177,88

As we can see from Table 1 for *adult* and *votes* sets all tested strategies have created the same global model - min and max values are the same. This situation is not surprising considering very small number of clusters and additionally the same for all local models. In case of *cars* set these values are different. The best strategies for *cars* set were MIN, CTR, MF. Created models have the same category utility value 0,2211. The worst global model was created with MAX strategy - category utility value 0,1853. Clusters' distance measure has no influence on model quality. The most important conclusion is that in all cases we have obtained global models with category utility oscillated around control model category utility (from 97,83% to 99,92%). Another thing worth mentioning is that all global models were much better than the summary model which category utility we have used as minimum for global model.

Table 2. Comparison of combining clusters' strategies for O-Cluster algorithm

Set	Category Utility percentage with respect to					
	control model		local models		summary model	
	min	max	min	max	min	max
adult	88,24	111,23	87,78	110,64	261,13	329,15
votes	74,02	107,87	61,50	89,62	178,62	260,30
cars	50,55	77,37	47,31	72,42	88,05	134,77
cen_big	91,15	99,76	92,99	101,77	279,42	305,81

Clustering tests for O-Cluster as local clustering algorithm. For *adult* set local models had 12, 13, 14 clusters and control model had 13 clusters (maximal number of clusters parameter value was set to 15). For *votes* set local models had 3 clusters and control model had 4 clusters (maximal number of clusters set to 10). For *cars* set local models had 5, 6, 6 clusters and control model had 6 clusters (maximal number of clusters set to 10). For *cen_big* set local models had 14, 15, 14 clusters and control model had 15 clusters (maximal number of clusters set to 15). For bigger number of clusters values in min and max column are different. This results from a larger number of possible combinations of clusters and more diversified global models. As we can see in Table 2 we have bigger dispersion in global models category utility. The best strategy for *adult* set was MIN - category utility value 0,1050. The worst global model was created with MF strategy with weighted average clusters' distance measure - category utility value 0,0833. The best strategy for *votes* set was CTR - category utility value 0,9946; the worst MIN strategy - category utility value 0,6925. The best strategy for *cars* set was MIN - category utility value 0,1128; the worst CTR strategy - category utility value 0,0737. The best strategy for *cen_big* set was CTR strategy with sum and average clusters' distance measure - category utility value 0,1263; the worst MAX strategy with sum and average clusters' distance measure - category utility value 0,1154. Global models category utility oscillated around control model category utility from 77,37% to 107,87%. Similarly to COBWEB algorithm, all global models were much better than the summary model which category utility we have used as minimum for global model.

3.2 Classification Tests Results

Detailed classification test results can be found in [4]. Best global model built using proposed strategies have accuracy comparable or better than voting models with respect to the control model. On the average, combining strategies for all sets had a slightly worse accuracy than voting models except for the *cars* set. Maximal accuracies for strategies oscillate around voting model accuracy. Maximal accuracies for strategies in relation to the control model were slightly worse than the voting model accuracy for *adult* set, in case of remaining sets they were equal or better. Global models' accuracies range from 86% to 101% of the control model accuracy.

3.3 Performance Tests Results

We have compared control model building time with time needed for parallel local model building and combining local models into global one. For classification distributed processing was more than three-fold quicker than non distributed one for the *adult* set and two-fold for the *cars* set. In case of the *votes* set distributed model building last 60% of the time needed for control model building (non distributed).

In case of clustering for COBWEB local algorithm we have achieved 3 to 8-fold acceleration. In case of O-Cluster as local algorithm results were much worse. For *adult*, *votes* and *cars* sets global models were built from 4,5 to 1,5 slower than the control model. Only for *cen_big* set we have achieved some acceleration (global model was built in 80% of time needed for control model). There are three main reasons for such differences. O-Cluster is a very fast algorithm designed for huge data sets - in some cases time needed to build control and local models was the same. We have used Oracle Java native implementation of O-Cluster, combining strategies were implemented in PL/SQL which is a much slower solution. COBWEB is an incremental algorithm for building cluster tree with single objects in nodes. The bigger the tree (as consecutive objects are processed) the more time is needed for processing. So time gain when set is divided into smaller parts is lucrative.

4 Summary

In this article we have presented new methods for creating distributed clustering and classification models. Distributed data mining scheme with global models quality verification was described.

During tests' procedures we have compared accuracy of global models with accuracy of control models, summary models (for clustering), voting models (for classification) and local models' average accuracy. We have also examined time needed for building and combining models. Obtained results are promising. Proposed methods improve model building efficiency without the significant loss of accuracy. Of course our solution required further examination. In case of classification we need to examine other classification's algorithms for local model building and conflicting rules modifications. For clustering we plan tests with different clusters representation (in form of rules). More tests with different nodes number and naturally distributed data are needed.

References

1. Chan, P., Prodromidis, A., Stolfo, G.: Meta-learning in distributed data mining systems: Issues and approaches. *Advances of Distributed Data Mining*. AAAI Press, Menlo Park (2000)
2. Guo, Y., Reuger, S.M., Sutiwaraphun, J., Forbes-Millot, J.: Meta-learning for parallel data mining. In: *Proceedings of the 7th Parallel Computing Workshop* (1997)

3. Caragea, D., Silvescu, A., Honavar, V.: Invited Paper. a Framework for Learning from Distributed Data Using Sufficient Statistics and its Application to Learning Decision Trees. *International Journal of Hybrid Intelligent Systems* 1(2), 80–89 (2004)
4. Gorawski, M., Pluciennik, E.: Distributed Data Mining Methodology with Classification Model Example. In: *1st International Conference on Computational Collective Intelligence - Semantic Web, Social Networks & Multiagent Systems, ICCCI, Wroclaw, Poland, October 5-7 (2009)*
5. Grossman, R., Turinsky, A.: A Framework for Finding Distributed Data Mining Strategies That Are Intermediate Between Centralized Strategies and In-Place Strategies. In: *Proceedings of Workshop on Distributed and Parallel Knowledge Discovery at KDD 2000*, pp. 1–7 (2000)
6. Theodorakis, M., Vlachos, A., Kalambovikis, T.Z.: Using Hierarchical Clustering to Enhance Classification Accuracy. In: *Proceedings of 3rd Hellenic Conference in Artificial Intelligence, Samos (2004)*
7. Topchy, A., Jain, A.K., Punch, W.: Clustering ensembles: models of consensus and weak partitions. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27(12), 1866–1881 (2005); Digital Object Identifier: 10.1109/TPAMI.2005.237
8. Petrakis, Y., Koloniari, G., Pitoura, E.: On Using Histograms as Routing Indexes in Peer-to-Peer System. In: Ng, W.S., Ooi, B.-C., Ouksel, A.M., Sartori, C. (eds.) *DBISP2P 2004. LNCS*, vol. 3367, pp. 16–30. Springer, Heidelberg (2005)
9. Gorawski, M., Pluciennik, E.: Analytical Models Combining Methodology with Classification Model Example. In: *First International Conference on Information Technology, Gdansk (2008)*, ISBN:978-1-4244-2244-9, http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4621623,
10. Asuncion, A., Newman, D.J.: *UCI Machine Learning Repository*. University of California, School of Information and Computer Science, Irvine (2007), <http://www.ics.uci.edu/~mllearn/MLRepository.html>
11. Quinlan, R.: Induction of Decision Trees. *Machine Learning* 1, 81–106 (1986)
12. Fisher, D.H.: Knowledge acquisition via incremental conceptual clustering. *Journal Machine Learning* 2(2), 139–172 (1987)
13. Milenova, B.L., Campos, M.M.: O-Cluster: Scalable Clustering of Large High Dimensional Data Sets. In: *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002)*, p. 290 (2002)

Task Management in Advanced Computational Intelligence System

Krzysztof Grąbczewski and Norbert Jankowski

Department of Informatics, Nicolaus Copernicus University, Toruń, Poland
{kg,norbert}@is.umk.pl
<http://www.is.umk.pl>

Abstract. Computational intelligence (CI) comes up with more and more sophisticated, hierarchical learning machines. Running advanced techniques, including meta-learning, requires general data mining systems, capable of efficient management of very complex machines. Requirements for running complex learning tasks, within such systems, are significantly different than those of running processes by operating systems. We address major requirements that should be met by CI systems and present corresponding solutions tested and implemented in our system. The main focus are the aspects of task spooling and multitasking.

1 Introduction

Thanks to continuous growth of computing power, that could be observed during recent decades, we can build more and more sophisticated software for solving more and more serious problems. Even home computers are now equipped with multi-core processors, so it is not too difficult to gather a number of quite powerful CPUs for scientific explorations. Naturally, it is still (and will always be) very important not to waste computing resources, so we need data mining systems capable of taking full advantage of available power. Because hardware development goes toward more parallel computations, computational intelligence (CI) systems must follow this direction and efficiently solve multitasking problems at kernel level. Advanced CI applications can easily be parallelized, because solving contemporary problems always requires testing many complex machines. How complex can the models be and how many computational experiments must be performed to obtain satisfactory solutions, can be seen by examples of recent data mining contests, e.g. the NIPS Feature Selection Challenge [65], different meta-learning enterprises [12,8,4] and miscellaneous ensemble machines approaches. Therefore, we need versatile tools for easy manipulation of learning machines. Intemi—the system, we have recently developed [37] is a general data mining tool, designed with emphasis on efficiency and eligibility for meta-learning.

In this paper, we present some kernel-level solutions of Intemi, which facilitate building variety of machines in simple and efficient way. After short introduction to basic architecture of the system (section 2), we present Intemi task management system. Major prerequisites can be found in section 3, then we describe Intemi task life cycle (section 4) and finally, the task spooling module of the system (section 5).

2 Intemi Basics

Intemi handles CI algorithms in a unified way based on the concept of *machines*. A general view of machine is presented in figure 1. The abstract concept of machine encircles not only learning machines but also all other algorithms related to data management (data loaders, data transformations, tests etc.). Machines exchange information by means of their inputs and outputs. Before a machine may be created, its process parameters and input bindings must be specified. After creation, machine process must be started, to prepare the machine for further services provided by outputs and possibly deposit adequate results in *results repository*. The process may be designed for arbitrary goals, from simple data loading to advanced data mining. Each machine is allowed to create other machines called its *submachines* or *child machines*. Machine trees, representing parent-child relations, and graphs of input–output interconnections are managed within *projects*.

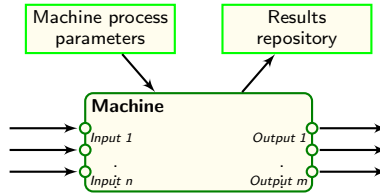


Fig. 1. The abstract view of machine

An example of complex machine is presented in figure 2. It is an instance of *Repeater*—a machine that, a specified number of times, repeats a test for subsequent collections of outputs provided by a distributor machine. The information about the number of repetitions, the test and the distributor is provided within *Repeater* machine configuration. The *schemes* included in the figure are machines responsible just for creation of a number of child machines in appropriate configuration. In the figures, child machines are depicted as boxes drawn within the area of their parents.

Advanced Intemi projects may produce large numbers of different machines. To handle them efficiently while preserving ease of use, the project must provide flexible tools. One of them is the task spooling system, presented in this article.

When dealing with large number of machines, it is quite probable, that the same machine is requested many times. To prevent from running the same machine repeatedly, Intemi is equipped with machine unification mechanism and a cache system. The problem has been solved at system level, because it is definitely more versatile than any machine-level solution and makes machine development easier. Task management module must also be aware of unification possibilities, to ensure that two identical machines will never be run—in such cases, the same machine must be assigned to all equivalent requests.

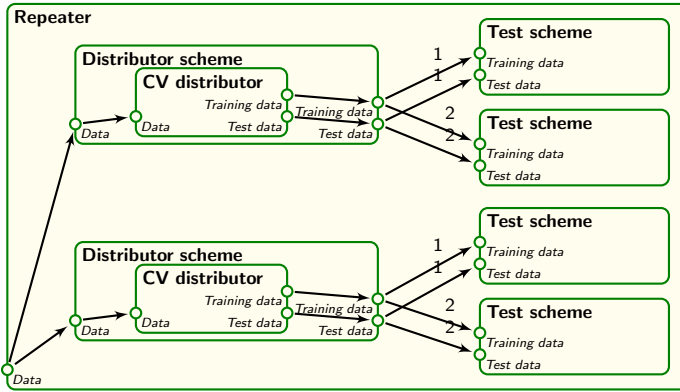


Fig. 2. Run time view of Repeater machine configured to perform twice 2-fold cross-validation (CV). Test schemes are simplified for clearer view—in fact each one contains four submachines as in figure 3.

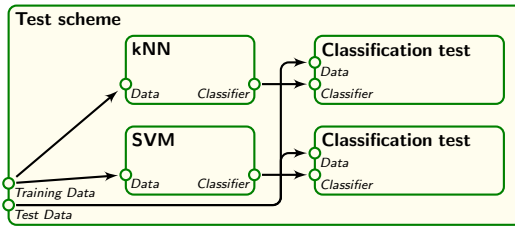


Fig. 3. A test scheme example

3 CI Task Management Prerequisites

Large scale CI computations require parallelization. It can be easily done, because in most projects, many machines may run independently. On the other side, we can not run all the requested machines instantly, because it could quickly overfill available memory. Therefore, we need to control the number of subprocesses running in parallel.

Task management in a CI system has its own specificity, quite different from task management inside operating systems (OS). Although in both problems, priorities are assigned to tasks, the differences are fundamental. For example, in operating systems we need a fair scheduler, i.e. a system of equal CPU allocation to all process of the same priority, while in advanced CI, memory saving is more important, because huge amounts of machines running at the same time would paralyze the system. Therefore, CI systems should rather care for finishing tasks as soon as possible and start new ones only if no further parallelization of currently running tasks is possible.

Another significant difference between CI and OS task management is that in CI we would like the system to reflect task dependencies in an automated way. Here, by *dependencies* we mean the input-output relation. We can create a task with inputs bound to outputs of a machine that is not started yet or is currently running, so that the new task can not be started until some other tasks are finished. Task synchronization in OS is a completely different problem.

The most important requirements for a CI task management system are:

- efficient exploitation of CPU resources i.e. taking advantage of as much CPU power as possible,
- protection against system memory exhaustion,
- automated resolving of dependencies between tasks,
- support for unification system in preventing from running equivalent tasks more than once.

The system, we have implemented within Intemi, satisfies all these conditions.

4 Task Life Cycle

Intemi machine request life follows one of many possible paths, according to the flowchart presented in figure 4. The flowchart blocks represent different states of the request, while dashed lines encircle the areas corresponding to particular system modules.

Each machine request is first analyzed to determine the machine contexts providing inputs to the requested machine. As it can be seen in figures 2 and 3, an input may be bound in a number of ways including binding to parent's input, to sibling's output (thus an output of a machine that does not exist at the time of configuration), etc. Therefore, the abstract information provided at machine configuration time, must be transformed into outputs specification containing references to machine instances. After that, the request is passed to the *input readiness control* module, where, if necessary, it waits for processes of other machines (the machines providing inputs) to finish. The inputs readiness guard keeps the collection of awaiting machine requests up to date, thanks to receiving (in real time) the information about each new machine request and about all the changes in machine (outputs) readiness. When all the inputs of a particular machine request are ready, it is examined by *machine cache*, whether the machine is already available because of earlier calculations. If the unification is not possible, the machine request is pushed to the *task spooler* and finally it can be fulfilled by a *task running* thread. The life of a request may be aborted at any time by parent machine or by system user. This may influence the flows of other requests for the same machine.

It needs to be stressed that the whole machine life cycle is managed completely automatically. From the user point of view, only the start and the end of the path, machine goes through, must be taken care of, i.e. the user orders a machine providing its configuration and inputs bindings, and then just waits for the machine (or for a collection of submachines) to be ready for further analysis.

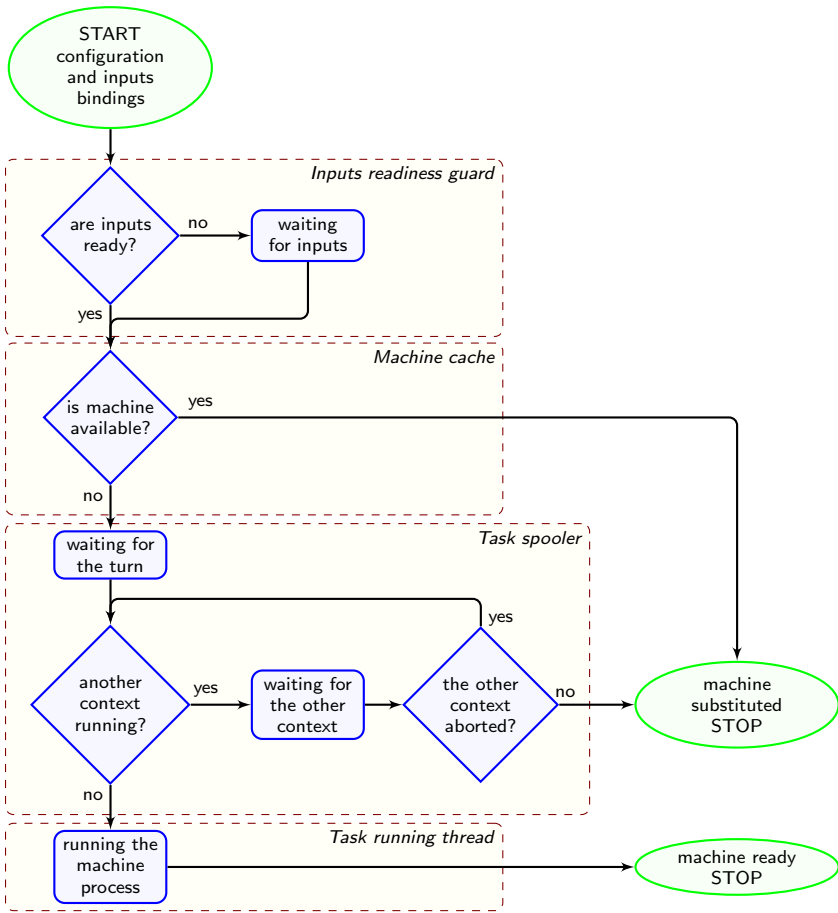


Fig. 4. Machine request life cycle

5 Task Spooling

In most computer systems dealing with tasks, structures of queues are quite successful in task ordering and preventing from running too large number of tasks in parallel. As signaled in section 3, in the case of CI projects, fundamental requirements are different, which makes solutions based on first-in-first-out (FIFO) methodology not appropriate.

In a CI project, it is not desirable to run all the subtasks in parallel and grant each of them with the same amount of CPU time. Instead, we prefer finishing the subtasks already running, over starting completely new ones. To address this issue, we have based Intemi task spooling system on a *tree structure with ordered nodes*. The tree nodes correspond to machines (running or scheduled for running) and the subnode relation is the submachine relation. The order of child machines reflects priorities assigned to the submachines at request time.

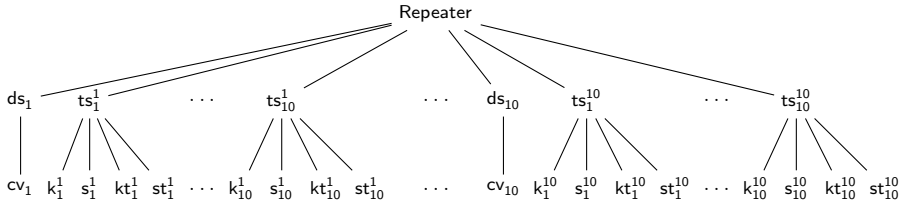


Fig. 5. Repeater submachine tree performing 10×10 -fold CV of tests defined in figure 3. Notation: ds – distributor scheme, ts – test scheme, cv – CV distributor, k – kNN, kt – classification test for kNN, s – SVM, st – classification test for SVM.

In the case of equal priorities, the time of request determines the order. When a machine is popped from the spooling tree, it is searched with hill climbing technique (i.e. depth first search respecting the order of child nodes).

To observe the advantages of Intemi spooling system structure in comparison to standard queue, let’s analyze the progress of calculating 10 repetitions of 10-fold CV to compare classification accuracy of kNN and SVM algorithms. Such configuration results in a repeater machine as presented in figures 2 and 3, but with 10 distributor schemes instead of 2 and 10 test schemes per distributor scheme, in place of 2. The resulting machine hierarchy is sketched in figure 5. The repeater machine creates 10 distributor schemes and 100 test schemes. Each distributor scheme creates one cross-validation distributor and each test scheme requests 4 child machines: kNN, classification test of kNN, SVM and classification test of SVM.

To avoid randomness of the process due to parallel calculation, we assume that all the tasks are calculated by a task manager with one running thread. The request for the repeater machine pushes the root node of the tree (of figure 5) into the spooler. When the request is popped out, the repeater process is run and puts all the repeater children to the queue: the first distribution scheme (ds_1), 10 test schemes (ts_1^1, \dots, ds_{10}^1) bound to ds_1 outputs, the second distribution scheme (ds_2) and so on. Thus, 110 machine requests go to the queue. After that, the repeater starts waiting for its children and the task manager calls for next task to run (ds_1 is popped). The distributor scheme requests the CV distributor machine (cv_1) and starts waiting until cv_1 is ready.

When a standard queue is used as the spooler, there are 109 requests in the queue before cv_1 , so it will be run after all the 109 preceding requests are popped, run and start waiting after pushing all their requests for children to the spooler. It means that when cv_1 gets its turn to run, 111 threads are in waiting mode (the repeater machine and all 110 of its children) and all the 410 machines of the third level are in the queue. So, the task manager controls 112 task threads. It costs a lot: the operating system must deal with many waiting threads and all the started machines occupy memory.

With Intemi spooling system based on tree with ordered nodes, the history of machine requests and pops is quite different. Only the begin is similar, because

the repeater machine is popped, run and it requests its 110 children. Then, ds_1 is popped out and run. It pushes cv_1 to the spooler and starts waiting. Next pop from the spooler returns not ts_1^1 as in the case of standard queue, but cv_1 , because the ds_1 branch is favored over all the other children of the repeater. When cv_1 is finished, ds_1 can be finished too, and ts_1^1 is run. It requests its 4 children, which are finished before ts_2^1 is started, thanks to the ordered tree based spooling system. As a result, only two waiting machine processes and one running may be observed at the same time, so the task manager controls only 3 threads. This is because the machines are popped from the spooler in the following order:

```
Repeater, ds1, cv1, ts11, k11, s11, kt11, st11, ..., ts101, k101, s101, kt101, st101, ...,
ds10, cv10, ts110, k110, s110, kt110, st110, ..., ts1010, k1010, s1010, kt1010, st1010,
```

while in the case of a standard FIFO the order is:

```
Repeater, ds1, ts11, ..., ts101, ..., ds10, ts110, ..., ts1010, cv1, k11, s11, kt11, st11,
..., k101, s101, kt101, st101, ..., cv10, k110, s110, kt110, st110, ..., k1010, s1010, kt1010, st1010.
```

Since, thanks to the spooling system, Intemi keeps just three running machines at a time, both memory and CPU time are saved significantly. In one of the example projects, peak memory usage was about **30 MB**, while with standard queue it was over **160 MB**. Also the overall time used by the projects was significantly reduced (around **15%**).

The idea of machine unification is that a machine must not be constructed twice. To make it possible, the task spooler must also be involved in unification, because the same machine request may occur in the spooler twice (two requests for the same machines but in different contexts, with different priorities). The spooling system may not allow for popping the same request twice, so when a request is popped, all its other instances must be blocked until the request is handled. When the task is finished successfully, all other requests are provided with the model, otherwise, the remaining requests stay in the spooler with proper priorities, to be started again in adequate time. The functionality of popping next task to run, is expressed algorithmically by the following meta-code:

```
1 Task GetNextTask() {
2   foreach (Task t in waiting_for_another_context) {
3     if (the other context aborted) return t;
4     if (the other context is ready)
5       { t.Status = Substituted; continue; }
6   }
7   while (true) {
8     if (spooler.IsEmpty) return null;
9     Task t = spooler.Pop();
10    if (machine of t is finished by another context)
11      { t.Status = Substituted; continue; }
12    if (machine of t is running within another context)
13      { waiting_for_another_context.Add(t); continue; }
14    return t;
15  }
16 }
```

The tasks, that receive their turn while other tasks requesting the same machine are running, are moved to the `waiting_for_another_context` collection, which is served in a privileged manner. It is processed in lines 2-6, before the main loop, processing the spooler (the tasks that have not got their turn yet), coded in lines 7-15.

6 Summary

Although task management module stays in background, it constitutes a very important part of any CI system. As we have presented, proper organization of task life cycle and task spooling may significantly improve performance and possibilities of the system. Naturally, task management must seamlessly cooperate with other modules of the system, e.g. with machine unification. System design respecting all aspects of machine learning processes, results in high efficiency and reliability of the system. Intemi task management approach has yielded many advantages including reduction of the number of simultaneously started machine processes. It facilitates computing more complex machines and reduces the time of computations. The scale of improvements has exceeded our expectations.

References

1. Brazdil, P., Soares, C., Da Costa, J.P.: Ranking learning algorithms: Using IBL and meta-learning on accuracy and time results. *Machine Learning* 50(3), 251–277 (2003)
2. Fürnkranz, J., Petrak, J.: An evaluation of landmarking variants. In: Giraud-Carrier, C., Lavra, N., Moyle, S., Kavsek, B. (eds.) *Proceedings of the ECML/PKDD Workshop on Integrating Aspects of Data Mining, Decision Support and Meta-Learning* (2001)
3. Grąbczewski, K., Jankowski, N.: Meta-learning architecture for knowledge representation and management in computational intelligence. *International Journal of Information Technology and Intelligent Computing* 2(2), 27 (2007)
4. Grąbczewski, K., Jankowski, N.: Meta-learning with machine generators and complexity controlled exploration. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008. LNCS (LNAI)*, vol. 5097, pp. 545–555. Springer, Heidelberg (2008)
5. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L.: *Feature extraction, foundations and applications*. Springer, Heidelberg (2006)
6. Guyon, I.: *Nips 2003 workshop on feature extraction* (December 2003), <http://www.clopinet.com/isabelle/Projects/NIPS2003>
7. Jankowski, N., Grąbczewski, K.: Learning machines information distribution system with example applications. In: *Computer Recognition Systems 2. Advances in Soft Computing*, pp. 205–215. Springer, Heidelberg (2007)
8. Kalousis, A., Hilario, M.: Model selection via meta-learning: a comparative study, pp. 406–413 (2000)

Combining the Results in Pairwise Classification Using Dempster-Shafer Theory: A Comparison of Two Approaches

Marcin Gromisz¹ and Sławomir Zadrozny²

¹ Doctoral Studies (SRI PAS), ul. Newelska 6, 01-447 Warsaw, Poland
Marcin.Gromisz@fuw.edu.pl

² Systems Research Institute PAS, ul. Newelska 6, 01-447 Warsaw, Poland
Sławomir.Zadrozny@ibspan.waw.pl
http://www.ibspan.waw.pl/~zadrozny/index_eng.htm

Abstract. The paper deals with the multi-class (polychotomous) classification problem solved using an ensemble of binary classifiers. The use of the Dempster-Shafer theory to combine the results of binary classifiers is studied. Two approaches are proposed and compared. Some experimental results are provided.

1 Introduction

The use of classifiers ensembles may improve the performance of the classification system [1]. It is also a popular technique used to solve multi-class classification problems using binary classifiers. Furthermore, binary classifiers may be chosen due to their known effectiveness, what is the case, e.g., of the support vector machines (SVM), or due to the lower cost of their training. Here we are dealing with an approach to the decomposition of a multi-class problem into a series of binary problems which is referred to as the *pairwise classification*.

This work has been inspired by the high-energy particles physics research [2], where the huge number of multidimensional experimental observations have to be classified. A huge training data set is needed if a multi-class classifier has to be created directly. The use of the pairwise classification technique mitigates this problem. The main issue addressed in the paper, is how to combine the results obtained using each of the binary classifiers. We propose to apply the Dempster-Shafer theory (DST) for that purposes and compare two possible approaches.

The paper is organized as follows. In Sect. 2 some relevant concepts are briefly reminded. Sect. 3 presents two aforementioned approaches. Sect. 4 reports the results of computational experiments and compares the approaches.

2 Preliminaries and Related Work

Pairwise Classification. The multi-class single-label classification problem is considered: there is a set of L classes $\mathcal{C} = \{c_1, c_2, \dots, c_L\}$ and a set of objects $X = \{x_i\}_{i \in I}$, characterized by a set of features, with known class assignment. The aim is to construct a classifier which will properly assign classes to objects.

Many classifiers are best suited for the binary case, i.e., where $\mathcal{C} = \{c_1, c_2\}$ [3]. To apply them directly in the multi-class case one may turn the original problem into a series of the binary problems and construct an *ensemble* of binary classifiers (*dichotomizers* [4]). There are two popular ways to do that [3]. In the “one-against-all” approach L dichotomizers T_{c_i} are created, one for each class $c_i \in \mathcal{C}$, such that T_{c_i} solves a binary classification problem in which one class is c_i and the another class comprises the objects of all other classes $c_j \in \mathcal{C}$, $j \neq i$. In the “one-against-one” approach (aka *pairwise classification* or *round robin*) $L(L-1)/2$ dichotomizers T_{c_i, c_j} are created, one for each pair of classes $c_i, c_j \in \mathcal{C}$, such that T_{c_i, c_j} solves a binary classification problem with two classes c_i and c_j .

We are interested in how to combine the output of dichotomizers in the “one-against-one” approach. A popular technique of *voting* works as follows: if a dichotomizer T_{c_i, c_j} assigns the class c_k , $k \in \{i, j\}$, to an object x then it is treated as a *vote* for this class, and the class with the highest number of votes is finally assigned to x . However, a classifier T_{c_i, c_j} is trained to distinguish only between classes c_i and c_j , thus its vote for an object from a different class should be taken with care. A way to overcome this difficulty may consist in an explicit modeling of the uncertainty related to the output of a dichotomizer.

Basics of the Dempster-Shafer Theory of Evidence (DST). This theory [5] provides means for a formal representation and processing of the evidence. Assuming that the evidence concerns the class to which a given object belongs, the DST may be summarized as follows. The set \mathcal{C} is an exhaustive list of the mutually exclusive classes, which in the DST is called as *the frame of discernment*. The evidence concerning the belongingness of an object x to a class is represented by *the basic probability assignment (bpa)* m^x , i.e., a function:

$$m^x : 2^{\mathcal{C}} \longrightarrow [0, 1], \quad \sum_{C \subseteq \mathcal{C}} m^x(C) = 1, \quad m(\emptyset) = 0 . \quad (1)$$

A subset $A \subseteq \mathcal{C}$ such that $m^x(A) > 0$ is referred to as the *focal element* of m^x . A bpa expresses the probability that the class of an object x belongs to $C \subseteq \mathcal{C}$. The functions Bel^x and Pl^x , associated with a basic probability assignment m^x :

$$Bel^x(B) = \sum_{A \subseteq B} m^x(A), \quad Pl^x(B) = \sum_{A: A \cap B \neq \emptyset} m^x(A) , \quad (2)$$

represent the *total belief* and the *plausibility*, respectively, that the class of an object x belongs to a set $B \subseteq \mathcal{C}$; the former function is called *the belief function* and the latter *the plausibility function*. A special class of belief functions, referred to as the *simple support functions*, is distinguished, which correspond to the bpa of the type: $m(A) = \alpha$, $m(\mathcal{C}) = 1 - \alpha$, for a subset $A \subseteq \mathcal{C}$ and $\alpha \in (0, 1]$. Such belief functions (or equivalently, basic probability assignments) represent the body of evidence which supports to a degree α only one proper subset of the frame of discernment and the whole frame \mathcal{C} .

Two independent pieces of evidence represented by the basic probability assignments m_1 and m_2 , may be combined and yield a basic probability assignment

m obtained using the the *orthogonal sum*, denoted as $m = m_1 \oplus m_2$ and defined as follows: $m(A) = \frac{\sum_{D,E:D \cap E=A} m_1(D)m_2(E)}{\sum_{D,E:D \cap E \neq \emptyset} m_1(D)m_2(E)}$, $\forall A \subseteq \mathcal{C}$, $A \neq \emptyset$.

A pair of belief/plausibility functions, Bel and Pl may be interpreted as representing a set $\mathcal{P} = \{P_i\}_{i \in J}$ of classical probability distributions P_i on \mathcal{C} such that $Bel(A) \leq P_i(A) \leq Pl(A)$, $\forall i \in J$ and $\forall A \subseteq \mathcal{C}$.

The *pignistic possibility distribution* [6], $BetP$, is defined as: $BetP(c) = \sum_{\{A:c \in A\}} \frac{m(A)}{|A|}$. It is a classical probability distribution that may be treated as representing a basic probability assignment m .

The Ordered Weighted Averaging (OWA) Operator [7]. The OWA operator of dimension n , with a weight vector $W = [w_1, \dots, w_n]$, $\sum_{i=1}^n w_i = 1$, is a function O_W such that $O_W : [0, 1]^n \rightarrow [0, 1]$, $O_W(a_1, \dots, a_n) = \sum_{i=1}^n w_i b_i$, where b_i is i -th largest element among a_i 's.

For the weights vectors $W = [1, 0, \dots, 0]$, $W = [0, \dots, 0, 1]$ and $W = [\frac{1}{n}, \dots, \frac{1}{n}]$, one obtains the maximum, minimum and average operators, respectively.

Yager [7] defined two measures characterizing the OWA operator O_W of dimension n , the *ORness*: $ORness(O_W) = \frac{\sum_{i=1}^n (n-i)w_i}{n-1}$, and the *dispersion*: $disp(O_W) = -\sum_{\substack{i=1 \\ w_i \neq 0}}^n w_i \ln(w_i)$. The former measures the similarity of an OWA operator to the maximum operator.

Related Work. There are a few papers concerning the use of the DST in pairwise classification. Quost et al. [8] consider dichotomizers T_{c_i, c_j} producing the bpa for the frame of discernment $\{c_i, c_j\}$. This bpa is treated as resulting from the conditioning of an unknown bpa defined on the whole set \mathcal{C} . The authors propose a way to recover these unknown bpas for each dichotomizer and then combine them to obtain the overall bpa. Moreover, the authors propose to take into account the plausibility of the fact that given object x belongs to one of the classes c_i and c_j , which modifies the bpa produced by this dichotomizer. Finally, the pignistic probability distribution is derived from thus obtained bpa and the class with the highest probability is assigned to an object under consideration.

Burger et al. [9] propose an approach similar in the spirit to one of ours. They associate a basic probability assignment with each dichotomizer's T_{c_i, c_j} decision for an object x , taking into account the distance of this object from a separating hyperplane (dichotomizers are assumed to be SVMs). The actual values of bpa are related to the membership degrees of these distances to three fuzzy sets directly modeling: belongingness of the object to one of two classes c_i and c_j , and the "hesitance" of the dichotomizer to classify x , respectively. The set of focal elements of these bpa's is as in our "exclusive" approach (cf. Sect. 3) but this choice is neither discussed nor analyzed in depth.

3 Two Approaches

Our aim is to use the Dempster-Shafer theory to model the uncertainty inherent to the pairwise classification. Thus, we have a set of dichotomizers T_{c_i, c_j} and we first look for the representation of their output in terms of the DST.

In the *first approach* we adopt the following line of reasoning. The result obtained for an object x using each dichotomizer, $T_{c_i, c_j}(x)$, gives only a hint as to the possible belongingness of x to one of the classes c_i and c_j . However, x may actually belong to another class $c_k \in \mathcal{C}$, $k \notin \{i, j\}$. Thus, we adopt the *exclusive interpretation* of the output of a dichotomizer and assume that it does not provide evidence for x 's belongingness to a specific class but rather for its non-belongingness, i.e., it *excludes* its belongingness to each of two classes c_i and c_j , to a varying degree. In particular, we treat the output of a dichotomizer T_{c_i, c_j} for x as providing *two bodies of evidence*, denoted as “ $\neg i$ ” and “ $\neg j$ ”, respectively:

$$\begin{aligned} \text{“}\neg i\text{”} & \text{ supporting a hypothesis that } x \text{ does not belong to } c_i \ (\overline{\{c_i\}}) \\ \text{“}\neg j\text{”} & \text{ supporting a hypothesis that } x \text{ does not belong to } c_j \ (\{c_j\}) \end{aligned} \tag{3}$$

The output of a dichotomizer may be to some extent inconclusive as to the proper classification of x , what in the DST framework is expressed as a degree of support for the trivial hypothesis that x belongs to one of the classes in \mathcal{C} .

Two bodies of evidence (3), provided by a dichotomizer T_{c_i, c_j} , are represented by two basic probability assignments $m_{\neg i}^x(\cdot)$ and $m_{\neg j}^x(\cdot)$ (below $k = i$ or $k = j$):

$$m_{\neg k}^x(\overline{\{c_k\}}) = F_{\neg k}(x), \quad m_{\neg k}^x(\mathcal{C}) = 1 - F_{\neg k}(x), \quad m_{\neg k}^x(A) = 0 \quad \text{for other } A \subseteq \mathcal{C} \tag{4}$$

Finally, the output of a dichotomizer T_{c_i, c_j} is represented by a basic probability assignment $m_{T_{c_i, c_j}}$ defined as the orthogonal sum of the bpas $m_{\neg j}^x$ and $m_{\neg i}^x$ (4):

$$m_{T_{c_i, c_j}} = m_{\neg j}^x \oplus m_{\neg i}^x \tag{5}$$

with the following focal elements: $\overline{\{c_i\}}, \overline{\{c_j\}}, \overline{\{c_i, c_j\}}, \mathcal{C}$.

The form of the function $F_{\neg k}$ in (4) is determined using an empirically selected function of the output of a dichotomizer. In case the dichotomizer is an SVM, its output may be geometrically interpreted as the distance $\rho(x)$ of the vector x , representing an object to be classified, from the separating hyperplane. We propose to model $F_{\neg k}$ in (4) as logistic functions:

$$F_{\neg k}(x) = \frac{1}{1 + \exp[\beta_{\neg k}(\rho(x) - \delta_{\neg k})]} \tag{6}$$

where the parameters $\beta_{\neg k}$ and $\delta_{\neg k}$ are chosen to fit the normalized histograms of the empirical cumulative distribution function of the variable $\rho(x)$, computed for the the examples of class c_k in the training set. Figure 1 shows how the bpa is modeled, both for the first and the second approach proposed in this section.

Thus, each dichotomizer T_{c_i, c_j} is associated with a vector of its separating hyperplane w , and a set of parameters $\{\beta_{\neg i}, \delta_{\neg i}, \beta_{\neg j}, \delta_{\neg j}\}$ defining two bpas (6).

The overall output of the ensemble of dichotomizers is represented as the orthogonal sum m_{\oplus} of bpas (5):

$$m_{\oplus} = m_{T_1} \oplus m_{T_2} \oplus \dots \oplus m_{T_K} \tag{7}$$

where the set of dichotomizers T_{c_i, c_j} is denoted as $\{T_1, \dots, T_K\}$, $K = L(L-1)/2$.

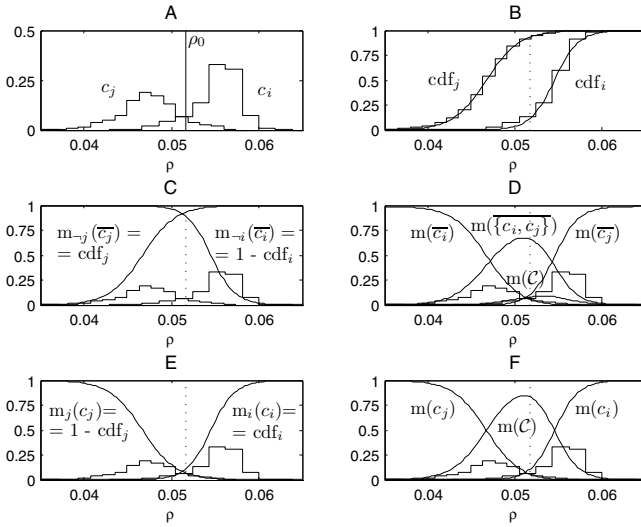


Fig. 1. Representing the output of a dichotomizer with a bpa: (A) empirical distribution of the distances from the separating hyperplane, ρ_0 , for the examples of the classes c_i and c_j present in the training dataset; (B) cumulative distribution functions (cdfs) for these distributions with logistic functions fitted; (C) bpa definition for the “exclusive” approach (cf. (4)), i.e., bpa for particular subsets of \mathcal{C} are defined as functions of the $\rho(x)$; (D) corresponding final “exclusive” bpa (cf. (5)); (E) bpa definition for the “inclusive” approach (cf. (9)); (F) corresponding final “inclusive” bpa. For better readability, the braces around single element sets are omitted; e.g., $m_{-j}(\overline{c_j})$ is shown instead of $m_{-j}(\{c_j\})$.

The *second approach* is based on the *inclusive interpretation* of the output of a dichotomizer T_{c_i, c_j} for an object x . Now, the output is interpreted as providing two bodies of evidence “supporting” each of the classes c_i and c_j rather than excluding them, as in the first, “exclusive” approach (cf. (3)):

$$\begin{aligned}
 \text{“}i\text{”} & \text{ supporting a hypothesis that } x \text{ does belong to } c_i \ (x \in \{c_i\}) \\
 \text{“}j\text{”} & \text{ supporting a hypothesis that } x \text{ does belong to } c_j \ (x \in \{c_j\})
 \end{aligned}
 \tag{8}$$

The degrees of support depend, as in the first approach, on the x ’s location with respect to the separating hyperplane defining T_{c_i, c_j} (cf., $\rho(x)$ and formula (6)). The induced basic probability assignments, m_k^x , $k \in \{i, j\}$, represent also some degree of ignorance, and take the following form:

$$m_k^x(\{c_k\}) = F_k(x), \quad m_k^x(\mathcal{C}) = 1 - F_k(x), \quad m_k^x(A) = 0 \quad \text{for other } A \subseteq \mathcal{C} \tag{9}$$

where $F_k(x) = \frac{1}{1 + \exp[\beta_k(\rho(x) - \delta_k)]}$, as in formula (6). Next, the bpas representing particular dichotomizers are combined to obtain $m_{T_{c_i, c_j}} = m_i^x \oplus m_j^x$ (cf. (5)); this time with the following focal elements: $\{c_i\}, \{c_j\}, \mathcal{C}$.

The overall output of the ensemble of “inclusive” dichotomizers is represented as for the “exclusive” approach, using the orthogonal sum m_{\oplus} (7) of bpas (9).

Thus, both in the “exclusive” and “inclusive” approaches we represent the output of an ensemble of dichotomizers with a bpa defined on \mathcal{C} (7). Then, it has to be decided which class should be assigned to an object x . The choice should be guided by the minimization of the cost of the misclassification, as in the Bayesian approach, where expected loss is minimized. We adopt the techniques proposed by Denoeux (10) and Yager (11), which adapt Bayesian approach in the context of the DST. Let the loss related to assigning class c_i to an object x , while its actual class is c_j , be denoted as $\lambda(c_i | c_j)$. In the classical Bayesian approach a classifier outputs for a given object x a probability distribution $P(\cdot | x)$ over the set of classes \mathcal{C} , and for each class $c_i \in \mathcal{C}$ the expected loss is computed as:

$$L(c_i | x) = \sum_{c_j \in \mathcal{C}} P(c_j | x) \lambda(c_i | c_j) . \tag{10}$$

If the output of a classifier is a bpa m_x the concept of expected value may be replaced with lower or upper expected value (12)(10) (cf. Sect. 2), leading to the following versions of the formula (10), respectively:

$$L_*(c_i | x) = \sum_{C \subseteq \mathcal{C}} m_x(C) \min_{c_j \in C} \lambda(c_i | c_j), \quad L^*(c_i | x) = \sum_{C \subseteq \mathcal{C}} m_x(C) \max_{c_j \in C} \lambda(c_i | c_j)$$

Alternatively, the pignistic probability distribution (cf. Sect. 2) corresponding to given bpa may be used in (10) instead of P . All these counterparts of the classical Bayesian approach (10) may be seen as specific cases of Yager’s approach (11), in which the OWA operator is used, and the expected loss is defined as:

$$L^{OWA}(c_i | x) = \sum_{\substack{C \subseteq \mathcal{C} \\ C = \{c_{j_1}, c_{j_2}, \dots, c_{j_m}\}}} m_x(C) O^W(\lambda(c_i | c_{j_1}), \lambda(c_i | c_{j_2}), \dots, \lambda(c_i | c_{j_m})) \tag{11}$$

where O_W denotes an OWA operator of dimension $n = |C|$. The weights of the OWA operators in (11) are set selecting a *degree of optimism* (the ORness degree) and maximizing the dispersion (11). The earlier mentioned other counterparts of the classical Bayesian approach may be obtained from (11) by using the OWA operators corresponding to the min, max and arithmetic mean operators.

We propose to use (11), in both the “exclusive” and “inclusive” approach, to compute the loss related to a given classification decision. Then, for an object x a class which minimizes the loss is assigned.

4 Computational Experiments

We have carried out a series of computational experiments in order to test and compare two approaches presented in Sect. 3. We compared the same ensemble of dichotomizers, implemented as SVMs with the RBF kernel. The output of

Table 1. The accuracy for particular approaches (in rows) and different groupings of classes (in columns) used for the Abalone dataset (in the parentheses the percentage of cases, where more than one class emerged as an output of a classifier, is shown)

Aggregation	Number of classes after grouping			
	#3	#4	#5	#6
voting	65.42 (0.29)	60.63 (0.19)	55.08 (1.72)	44.82 (4.21)
“exclusive”	65.42 (0)	60.63 (0)	53.73 (0)	42.62 (0)
“inclusive”	57.76 (0)	53.83 (0.10)	37.93 (3.45)	25.86 (0.67)

the dichotomizers is, however, interpreted in the following three different ways: using “exclusive” and “inclusive” approaches of Sect. 3, and using the voting.

The experiments have been carried out using the Abalone dataset of the UCI Machine Learning Repository [13]. The aim of the classification is to predict the age of an abalone (a kind of a sea snail) using its physical measurements. The number of classes in this dataset is 29. In [14] this dataset was used with grouping the classes into 3 superclasses. We have tested our approaches for various groupings of the original classes: into 3 classes (as in [14]: with classes 1-8, 9-10, and 11-29 combined), into 4 classes (classes 1-7, 8-9, 10-11 and 12-29), into 5 classes (classes 1-7, 8-9, 10, 11-12 and 13-29), and into 6 classes (classes 1-7, 8, 9, 10, 11-12, and 13-29). The dataset is divided into the training and testing parts as in [14], i.e., there are 3133 training examples and 1044 testing examples.

The results obtained in the experiments are summarized in Tab. 1. The assumed grouping of the original classes yields balanced superclasses, thus the accuracy of the classification given in Tab. 1 provides a reliable evaluation. Poor results obtained for the “inclusive” approach are a consequence of a too strong conclusion expressed by the bpa (9), concerning the output of a dichotomizer. It ignores the fact that an object under consideration may belong to a class different from, both, c_i and c_j , and thus the “preference” of a dichotomizer T_{c_i, c_j} , e.g., for c_i with respect to c_j , should not be translated to such a strong support for c_i . This experiment shows the advantage of the “exclusive” approach over the “inclusive” one.

Table 1 seems to suggest that the DST based approach does not bring an advantage in comparison to the simple voting technique. This may be explained by the fact that basic probability assignment values are correlated with the binary decision of an underlying dichotomizer, i.e., overall bpa indications directly correspond with the number of “votes” a given class obtains. However, a DST based approach provides a convenient framework for taking into account the cost of misclassification, what may be important for some classification tasks, cf., [2].

5 Concluding Remarks

We have discussed and compared two approaches to the pairwise classifiers aggregation using the Dempster-Shafer theory. The “exclusive” approach better

models the output of the dichotomizers, what was confirmed by the computational experiments. It seems to be also less dependent on the choice of the optimism degree used in the Yager's counterpart of the Bayesian loss function. This however requires further studies. It gives also comparable results to the most popular technique of voting and provides a convenient framework for the account for the misclassification cost.

Acknowledgments. This work reports the results of computational experiments carried out using the UCI Machine Learning Repository [13] and the LIBSVM software [15].

References

1. Kuncheva, L.I.: *Combining Pattern Classifiers: Methods and Algorithms*. Wiley Interscience, Hoboken (2004)
2. Gromisz, M.: On the application of an ensemble of classifiers for data preselection. In: Hołubiec, J. (ed.) *Systems Analysis for Financial and Management Applications*, vol. 11, EXIT, Warszawa (2009) (in Polish)
3. Fürnkranz, J.: Round robin classification. *J. of Mach. Learn. Res.* 2, 721–747 (2002)
4. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*. J. Wiley & Sons, Chichester (2001)
5. Shafer, G.: *A mathematical theory of evidence*. Princeton University Press, Princeton (1976)
6. Smets, P., Kennes, R.: The transferable belief model. *Artif. Intell.* 66(2), 191–234 (1994)
7. Yager, R.: On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Trans. on Syst., Man and Cybern.* 18, 183–190 (1988)
8. Quost, B., Denoeux, T., Masson, M.H.: Pairwise classifier combination using belief functions. *Pattern Recognition Letters* 28(5), 644–653 (2007)
9. Burger, T., Aran, O., Caplier, A.: Modeling hesitation and conflict: A belief-based approach for multi-class problems. In: *Proceedings of the 5th ICMLA Conference*, Washington, DC, USA, pp. 95–100. IEEE Computer Society, Los Alamitos (2006)
10. Denoeux, T.: Analysis of evidence-theoretic decision rules for pattern classification. *Pattern Recognition* 30(7), 1095–1107 (1997)
11. Yager, R.R.: Decision making under Dempster-Shafer uncertainties. In: Yager, R.R., Liu, L. (eds.) *Classic Works of the Dempster-Shafer Theory of Belief Functions. Studies in Fuzziness and Soft Computing*, vol. 219, pp. 619–632. Springer, Heidelberg (2008)
12. Smets, P.: The degree of belief in a fuzzy event. *Inf. Sci.* 25(1), 1–19 (1981)
13. Asuncion, A., Newman, D.: UCI machine learning repository (2007)
14. Clark, D., Schreter, Z., Adams, A.: A quantitative comparison of dystal and back-propagation. In: *Proceedings of the Seventh ACNN Conference*, Australia (1996)
15. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines (2001), <http://www.csie.ntu.edu.tw/~cjlin/libsvm>

Pruning Classification Rules with Reference Vector Selection Methods

Karol Grudziński¹, Marek Grochowski², and Włodzisław Duch²

¹ Institute of Physics, Kazimierz Wielki University, Bydgoszcz, Poland

² Dept. of Informatics, Nicolaus Copernicus University, Grudziądzka 5, Poland
grudzinski.k@gmail.com, grochu@is.umk.pl, Google: W. Duch

Abstract. Attempts to extract logical rules from data often lead to large sets of classification rules that need to be pruned. Training two classifiers, the C4.5 decision tree and the Non-Nested Generalized Exemplars (NNGE) covering algorithm, on datasets that have been reduced earlier with the E_kP instance compressor leads to statistically significantly lower number of derived rules with non-significant degradation of results. Similar results have been observed with other popular instance filters used for data pruning. Numerical experiments presented here illustrate that it is possible to extract more interesting and simpler sets of rules from filtered datasets. This enables a better understanding of knowledge structures when data is explored using algorithms that tend to induce a large number of classification rules.

1 Introduction

Induction of classification rules is one of the main data mining tasks, allowing for summarization of data and understanding their structure. Numerous systems have been designed for that purpose [8]. However, it is usually hard to extract low number of very informative rules without sacrificing their generalization ability. A few methods perform well on most data generating relatively low number of rules, but most rule-based systems tend to induce quite large number of rules, making the solution obtained difficult to understand. Reducing the number of classification rules is therefore an important issue in data mining.

In this paper the effect of instance selection (pruning training data) on the number of generated rules and their generalization ability is investigated. The E_kP method [11,12] has been used to select reduced reference vectors. Two classifiers capable of rule induction have been taken for our experiments, but results should generalize to other types of covering algorithms and decision trees. The first one is the NNGE system, available in the Weka package [20], which usually generates a large number of covering hyperrectangles, or logical rules. The second method, called PART [9] is based on C4.5 decision tree [17], used recursively to generate rules (taken from largest node, and removing the data covered so far). These classifiers have been trained on original data and on training partitions reduced by E_kP . Additional computational experiments with other popular instance compressors have also been performed, but only the C4.5 decision tree results are reported here to save space.

2 Pruning Classification Rules with Reference Vector Selection Methods

Three types of classification rules may be used for data understanding [8]. Propositional logical rules use hyperboxes to define decision borders between classes, they are generated using either univariate decision trees or covering methods. Second, threshold logic rules, equivalent to hyperplanes that may be generated either by multivariate trees or by linear discriminants, for example linear support vector machines. Third and most general, rules based on similarity to prototypes may provide complex decision regions, including hyperboxes and fuzzy decision regions. Prototype-based rules (P-rules) are comprehensible if similarity functions are sufficiently simple. The study of prototype-based rules has been much less popular than of the other forms of rules [7][3][6].

Below a short description of the instance pruning algorithms which have been employed in our numerical experiments is provided.

The EkP Prototype Selection System has been used in all experiments conducted in this paper [1][11][2]. Simplex method [16] for minimization of cost function (number of errors), as implemented by M. Lampton and modified by one of us (K.G.), has been used, despite its rather high computational cost. The advantage of simplex method is that it essentially does not require any user intervention to control the minimization process. The pseudocode for the simplex initialization algorithm used in our experiments is given in Algorithm 1 and the cost function procedure of the EkP system is given in Algorithm 2.

Algorithm 1. Simplex initialization algorithm for EkP

Require: A vector of training set instances `trainInstances[]`

Require: A vector `p[]` of optimization parameters (`numProtoPerClass * numClasses * numAttributes` dimensional)

Require: A matrix `simplex` to construct a simplex

`numPoints`, the number of points to build the simplex on

```

for  $i = 0$  to numPoints - 1 do
  randomize(trainInstances[])
  for  $j = 0$  to numClasses * numProtoPerClass - 1 do
    for  $k = 0$  to numAttributes - 1 do
      simplex[i][k] := p[k + numAttributes * j] := trainInstances[i][k]
    end for
  end for
end for
simplex[i][numAttributes] := costFunction(p[])
end for

```

Algorithm 2. The EkP cost function algorithm

Require: A training set `trainInstances[]`, a vector `p[]` of optimization parameters.

`tmpTrain`, empty training set.

for $i = 0$ to `numClasses * numProtoPerClass - 1` do

 for $j = 0$ to `numAttributes - 1` do

 Extract the prototype which is stored in `p[]` and add it to `tmpTrain`

 end for

end for

Build (train) the classifier on `tmpTrain` and test it on `trainInstances`

Remember the optimal `p[]` value and the lowest value of `numClassificationErrors` associated with it.

return `numClassificationErrors`

Other Reference Vector Selection Methods Used. Only a very concise description of the instance pruning algorithms that have been used in our experiments is given below. For in-depth review of these algorithms see [15,110,119].

- **Condensed Nearest Neighbor Rule (CNN).** [13] method starts with a reference set containing one vector per class and adds incrementally to this set each instance from the training data that is wrongly classified when that reference set is used for learning and the training instances are used for testing.
- **DROP3** [19] removes instance x from the training set if it does not change classification of instances associated with x . Vectors associated with x are defined as a set of instances for which instance x is one of the k nearest neighbors.
- **Edited Nearest Neighbor (ENN).** [18] removes a given instance from the training set if it's class does not agree with the majority class of its neighbors.
- **Edited NRBF.** [15] uses Normalized RBF [14] to estimate probability $P(C_k|x, T)$ of C_k class for a given vector x and the training set T . Each vector inconsistent with its class density estimation is treated as noise and is removed from the dataset. Probability that a vector from correct class will be removed is low.
- **Iterative Case Filtering (ICF).** [4] starts from DROP3 and creates hyperspheres that contain only single-class instances, removing instances which are located inside clusters of vectors from the same class.
- **Gabriel Editing (GE).** [2] method is based on graph theory. It uses the Gabriel graph to define neighbors and removes from the training dataset all instances that belong to the same class as all their neighbors.

3 Numerical Experiments

The E_kP prototype selection method is compared here with other instance compressors, and examine rules obtained from classifiers which have been trained on pruned data. In the first part experiments with the NNGE and C4.5 systems trained on a large number of datasets filtered using the E_kP method have been described. In the second part E_kP algorithm has been matched against several instance compressors. Finally explicit example of the benefit of this approach is demonstrated by presenting greatly simplified and highly accurate rules for non-trivial dataset.

Pruning C4.5 and NNGE Rules with the E_kP Instance Selector. Numerical experiments have been performed on 17 real-world problems taken mainly from the UCI repository of machine-learning databases [1], described in Table 1. The E_kP system has been used for instance selection and the rules have been extracted with the C4.5 and NNGE classifiers. All experiments have been performed using SBLWeka, an extension of Weka system [20], done by one of us (K.G.)

In all experiments 10 simplex points for the E_kP system are used and $j=1, 2, 3$ or 5 prototypes per class selected, denoted $E_kPs10pj$. In the first experiment the influence of data pruning on classification generalization has been examined (Tables 2 and 3). One prototype per class is not sufficient, but increasing the number of prototypes to 5 per class leads to results that are statistically equivalent to training on the whole dataset, with the NNGE method on all 17 problems, and with the C4.5 system on 16 problems.

Table 1. Datasets used in numerical experiments

#Dataset	# Instances	# Attributes	# Numeric	# Nominal	# Classes	Base Rate [%]	Rnd. Choice [%]
1 Appendicitis	106	8	7	1	2	80.2	50.0
2 Breast C.W.	286	10	0	10	2	70.3	50.0
3 Horse Colic	368	23	7	16	2	63.0	50.0
4 Credit rating	690	16	6	10	2	55.5	50.0
5 German credit	1000	21	8	13	2	70.0	50.0
6 Pima I.D.	768	9	8	1	2	65.1	50.0
7 Glass	214	10	9	1	6	35.5	16.7
8 Cleveland heart	303	14	6	8	2	54.4	50.0
9 Hungarian heart	294	14	6	8	2	63.9	50.0
10 Heart Statlog	270	14	13	1	2	55.6	50.0
11 Hepatitis	155	20	2	18	2	79.4	50.0
12 Labor	57	17	8	9	2	64.7	50.0
13 Lymphography	148	19	0	19	4	54.8	25.0
14 Primary Tumor	339	18	0	18	21	24.8	4.8
15 Sonar	208	61	60	1	2	53.4	50.0
16 Voting	435	17	0	17	2	61.4	50.0
17 Zoo	101	18	0	18	7	40.6	14.3
Average	337.8	18.2	8.2	9.9	3.8	58.4	41.8

Table 2. NNGE – Generalization Ability

Data Set	NNGE	EkPs10p1	EkPs10p2	EkPs10p3	EkPs10p5
Appendicitis	83.7 ± 11.3	85.6 ± 9.9	86.0 ± 9.3	85.7 ± 9.2	86.5 ± 10.0
Breast C.W.	67.8 ± 7.1	70.8 ± 5.5	72.4 ± 5.9	72.5 ± 6.2	71.1 ± 6.7
Horse Colic	79.0 ± 6.5	66.4 ± 7.5	76.7 ± 7.2	79.9 ± 6.5	81.5 ± 6.3
Credit rating	82.8 ± 4.7	72.3 ± 5.9	80.4 ± 5.0	84.5 ± 4.1	85.2 ± 4.1
German credit	69.2 ± 4.5	69.9 ± 0.7	70.1 ± 2.0	70.0 ± 2.6	70.2 ± 1.9
Pima I.D.	72.8 ± 4.6	67.5 ± 5.0	70.3 ± 4.8	72.2 ± 4.5	72.6 ± 5.0
Glass	68.0 ± 9.3	53.6 ± 9.3	57.9 ± 9.5	61.1 ± 7.7	62.5 ± 9.0
Cleveland heart	77.8 ± 7.7	79.0 ± 7.1	77.6 ± 7.6	78.6 ± 7.7	77.6 ± 7.0
Hungarian heart	79.6 ± 6.8	81.6 ± 6.0	82.5 ± 6.5	81.7 ± 7.3	81.0 ± 7.2
Heart Statlog	77.3 ± 8.1	74.3 ± 9.2	79.0 ± 7.4	77.9 ± 7.5	77.2 ± 8.0
Hepatitis	81.9 ± 8.1	78.5 ± 6.1	82.7 ± 8.4	83.1 ± 8.0	82.9 ± 7.5
Labor	86.2 ± 15.2	85.9 ± 15.8	83.1 ± 17.4	82.8 ± 14.8	84.2 ± 13.8
Lymphography	77.1 ± 10.1	75.3 ± 10.7	73.3 ± 10.3	74.6 ± 9.5	75.7 ± 9.2
Primary Tumor	39.1 ± 7.2	34.7 ± 6.7	36.6 ± 6.9	38.4 ± 7.1	39.1 ± 6.6
Sonar	71.1 ± 9.2	63.2 ± 11.9	67.6 ± 10.3	68.6 ± 9.3	69.3 ± 9.8
Voting	95.1 ± 3.1	88.8 ± 4.5	93.0 ± 4.1	94.8 ± 3.6	94.9 ± 3.1
Zoo	94.1 ± 6.4	83.6 ± 8.5	90.0 ± 8.0	93.1 ± 6.2	95.4 ± 6.2
Average	76.6 ± 12.6	72.4 ± 13.4	75.2 ± 13.2	76.4 ± 13.0	76.9 ± 13.1
Win/Tie/Lose		0/11/6	0/16/1	0/17/0	0/17/0

o, • statistically significant improvement or degradation.

With these 5 prototypes per class statistically lower number of rules was obtained in 16 cases for NNGE and 17 times in case of C4.5 (see Table 4 and 5).

Comparison of pruning rules with various data compressors. The efficiency of EkP algorithm has also been compared with several vector selection methods listed in section 2. Table 6 presents average accuracy of classification estimated using 10-fold stratified cross-validation tests repeated 10 times. The average number of rules generated by the C4.5 algorithm is reported in Tab. 7. First column contains results for C4.5 trained on entire training dataset, and each successive column represent results for C4.5 trained on data reduced by one of the vector selection methods. Columns are sorted according to the average compression achieved by these methods, as shown in the last row of Tab. 6. The number of resulting prototypes in EkP method was set for each training to the

Table 3. C4.5 – Generalization Ability

Data Set	C4.5	EkPs10p1	EkPs10p2	EkPs10p3	EkPs10p5
Appendicitis	84.6 ± 10.3	80.2 ± 2.6	83.7 ± 8.7	84.4 ± 11.4	85.5 ± 9.7
Breast C.W.	69.4 ± 7.6	70.3 ± 1.4	71.0 ± 5.1	69.1 ± 6.7	69.5 ± 7.1
Horse Colic	84.4 ± 5.9	63.0 ± 1.1 ●	78.7 ± 8.6	81.5 ± 5.8	82.2 ± 5.9
Credit rating	84.4 ± 4.3	55.4 ± 1.2 ●	73.5 ± 11.7 ●	85.5 ± 4.0	85.5 ± 3.9
German credit	70.5 ± 4.2	70.0 ± 0.0	69.8 ± 1.1	69.6 ± 1.9	69.6 ± 1.9
Pima I.D.	73.4 ± 4.5	65.1 ± 0.3 ●	69.5 ± 5.7	71.5 ± 5.9	73.4 ± 5.0
Glass	68.7 ± 10.6	48.9 ± 9.1 ●	56.1 ± 7.1 ●	58.1 ± 8.8 ●	60.1 ± 9.6 ●
Cleveland heart	78.0 ± 7.1	74.1 ± 7.6	73.3 ± 7.1	73.5 ± 7.3	77.1 ± 8.1
Hungarian heart	81.1 ± 6.7	80.6 ± 7.9	80.8 ± 7.1	80.0 ± 6.7	79.3 ± 7.6
Heart Statlog	77.3 ± 7.8	55.6 ± 0.0 ●	71.0 ± 9.5	72.8 ± 8.3	72.6 ± 7.8
Hepatitis	79.8 ± 8.5	79.4 ± 2.3	79.5 ± 3.8	81.4 ± 6.4	82.1 ± 9.0
Labor	77.7 ± 15.5	64.7 ± 3.1 ●	79.1 ± 14.8	77.7 ± 15.8	79.8 ± 15.1
Lymphography	76.4 ± 9.3	68.7 ± 12.	72.7 ± 11.0	72.8 ± 10.9	76.5 ± 11.4
Primary Tumor	40.9 ± 6.4	31.1 ± 6.4 ●	36.9 ± 6.7	36.9 ± 7.7	38.0 ± 7.6
Sonar	77.4 ± 9.4	53.4 ± 1.6 ●	59.6 ± 12.2 ●	68.6 ± 10.3 ●	69.6 ± 10.1
Voting	96.0 ± 3.2	61.4 ± 0.8 ●	94.4 ± 4.2	95.6 ± 2.8	95.6 ± 2.8
Zoo	93.4 ± 7.3	71.6 ± 5.7 ●	81.7 ± 6.2 ●	87.3 ± 7.4 ●	91.5 ± 7.7
Average	77.3 ± 12.0	64.3 ± 12.8	72.4 ± 12.8	74.5 ± 13.1	75.8 ± 13.1
Win/Tie/Lose		0/7/10	0/13/4	0/14/3	0/16/1

○, ● statistically significant improvement or degradation.

Table 4. NNGE – Number of Rules

Data Set	NNGE	EkPs10p1	EkPs10p2	EkPs10p3	EkPs10p5
Appendicitis	16.0 ± 2.2	1.9 ± 0.2 ●	2.0 ± 0.1 ●	2.3 ± 0.5 ●	2.9 ± 1.0 ●
Breast C.W.	86.4 ± 5.1	1.7 ± 0.4 ●	2.0 ± 0.0 ●	2.1 ± 0.3 ●	2.8 ± 0.9 ●
Horse Colic	97.6 ± 18.8	1.9 ± 0.3 ●	2.0 ± 0.0 ●	2.0 ± 0.0 ●	2.2 ± 0.4 ●
Credit rating	142.4 ± 14.6	2.0 ± 0.0 ●	2.0 ± 0.0 ●	2.0 ± 0.1 ●	2.0 ± 0.2 ●
German credit	347.4 ± 26.0	1.1 ± 0.2 ●	1.8 ± 0.4 ●	2.0 ± 0.4 ●	2.8 ± 0.8 ●
Pima I.D.	263.8 ± 23.2	1.8 ± 0.4 ●	2.0 ± 0.0 ●	2.1 ± 0.3 ●	2.7 ± 0.8 ●
Glass	48.1 ± 4.8	3.9 ± 0.7 ●	6.0 ± 1.0 ●	8.1 ± 1.3 ●	11.9 ± 1.5 ●
Cleveland heart	71.6 ± 9.3	2.0 ± 0.0 ●	2.9 ± 0.7 ●	4.0 ± 0.9 ●	6.1 ± 1.3 ●
Hungarian heart	61.9 ± 7.2	2.0 ± 0.1 ●	2.6 ± 0.7 ●	3.6 ± 1.1 ●	5.9 ± 1.5 ●
Heart Statlog	68.3 ± 8.7	2.0 ± 0.0 ●	2.0 ± 0.0 ●	2.0 ± 0.1 ●	2.8 ± 0.8 ●
Hepatitis	27.6 ± 3.8	1.5 ± 0.5 ●	2.0 ± 0.1 ●	2.1 ± 0.3 ●	2.4 ± 0.6 ●
Labor	7.9 ± 1.4	2.0 ± 0.1 ●	2.0 ± 0.0 ●	2.1 ± 0.3 ●	2.6 ± 0.7 ●
Lymphography	32.0 ± 5.2	2.1 ± 0.3 ●	2.8 ± 0.7 ●	3.8 ± 1.1 ●	5.9 ± 1.3 ●
Primary Tumor	147.8 ± 4.7	13.3 ± 1.7 ●	24.4 ± 2.5 ●	35.8 ± 3.0 ●	58.2 ± 3.9 ●
Sonar	45.7 ± 7.2	2.0 ± 0.0 ●	2.0 ± 0.0 ●	2.0 ± 0.1 ●	2.4 ± 0.6 ●
Voting	29.0 ± 3.6	2.0 ± 0.0 ●	2.0 ± 0.0 ●	2.0 ± 0.1 ●	2.1 ± 0.3 ●
Zoo	7.0 ± 0.0	5.0 ± 0.5 ●	6.4 ± 0.5 ●	6.9 ± 0.3	7.0 ± 0.0
Average	88.3 ± 92.9	2.8 ± 2.8	3.9 ± 5.4	5.0 ± 8.1	7.2 ± 13.4
Win/Tie/Lose		0/0/17	0/0/17	0/1/16	0/1/16

○, ● statistically significant improvement or degradation.

value that corresponds to about 10% of the size of the original training set. On average EkP produced smallest size data among all methods compared here. Table 6 shows that the EkP algorithm, despite such high reduction of the training data size, was able to achieve good accuracy in comparison to other pruning methods tested here. For two datasets (Horse Colic and Breast Cancer Wisconsin) paired corrected t-test shows significant improvement in favor of EkP, each time producing about 6 times less rules than C4.5 with all training data. Only GE and ENN methods can compete in generalization with EkP, giving no significant difference in comparison with original C4.5. However these pruning techniques produce training data with average size reduced only to 85% in case of ENN, and 95% for GE, respectively, while the EkP method creates much smaller datasets.

Table 5. C4.5 – Number of Rules

Data Set	C4.5	EkPs10p1	EkPs10p2	EkPs10p3	EkPs10p5
Appendicitis	3.1 ± 0.6	1.0 ± 0.0●	1.7 ± 0.4●	2.0 ± 0.2●	2.0 ± 0.1●
Breast C.W.	18.4 ± 4.2	1.0 ± 0.0●	1.5 ± 0.5●	1.9 ± 0.5●	2.3 ± 0.5●
Horse Colic	9.1 ± 2.7	1.0 ± 0.0●	1.9 ± 0.3●	2.0 ± 0.0●	2.3 ± 0.5●
Credit rating	31.5 ± 7.7	1.0 ± 0.0●	2.0 ± 0.2●	2.0 ± 0.0●	2.0 ± 0.0●
German credit	69.7 ± 5.8	1.0 ± 0.0●	1.1 ± 0.3●	1.1 ± 0.4●	1.3 ± 0.6●
Pima I.D.	7.5 ± 1.5	1.0 ± 0.0●	1.8 ± 0.4●	1.9 ± 0.2●	2.0 ± 0.1●
Glass	15.2 ± 1.6	2.6 ± 0.5●	3.5 ± 0.6●	4.5 ± 0.9●	6.6 ± 1.3●
Cleveland heart	19.6 ± 2.7	2.0 ± 0.0●	2.1 ± 0.3●	2.6 ± 0.7●	3.9 ± 0.8●
Hungarian heart	8.2 ± 2.4	2.0 ± 0.0●	2.1 ± 0.3●	2.3 ± 0.6●	3.3 ± 1.2●
Heart Statlog	17.6 ± 2.4	1.0 ± 0.0●	2.0 ± 0.0●	2.0 ± 0.0●	2.2 ± 0.4●
Hepatitis	8.6 ± 1.7	1.0 ± 0.0●	1.1 ± 0.3●	1.6 ± 0.5●	2.1 ± 0.3●
Labor	3.4 ± 0.8	1.0 ± 0.0●	2.0 ± 0.1●	2.0 ± 0.0●	2.1 ± 0.3●
Lymphography	11.3 ± 2.3	2.0 ± 0.2●	2.1 ± 0.4●	2.6 ± 0.6●	3.4 ± 0.6●
Primary Tumor	41.1 ± 3.5	6.1 ± 0.9●	10.4 ± 1.3●	13.8 ± 1.9●	20.4 ± 2.3●
Sonar	7.5 ± 1.0	1.0 ± 0.0●	2.0 ± 0.0●	2.0 ± 0.0●	2.0 ± 0.0●
Voting	6.1 ± 1.1	1.0 ± 0.0●	2.0 ± 0.0●	2.0 ± 0.0●	2.0 ± 0.0●
Zoo	7.6 ± 0.5	3.0 ± 0.0●	4.9 ± 0.5●	6.0 ± 0.5●	6.9 ± 0.4●
Average	16.8 ± 16.9	1.7 ± 1.3	2.6 ± 2.2	3.1 ± 3.0	3.9 ± 4.5
Win/Tie/Lose		0/0/17	0/0/17	0/0/17	0/0/17

○, ● statistically significant improvement or degradation.

Table 6. Average classification accuracy

Data Set	C4.5	EkP	ENRBF	DROP3	ICF	CNN	ENN	GE
Appendicitis	85.3 ± 10.4	84.1 ± 10.3	80.3 ± 18.5	83.0 ± 11.3	82.0 ± 12.1	79.9 ± 14.1	83.6 ± 10.9	84.1 ± 11.1
Breast C.W.	68.6 ± 8.7	73.6 ± 7.8 ○	58.8 ± 12.0●	65.3 ± 12.1	66.0 ± 9.3	61.7 ± 10.0	70.3 ± 8.1	68.0 ± 8.2
Horse Colic	79.6 ± 6.0	84.6 ± 5.9 ○	77.0 ± 7.8	80.3 ± 6.5	76.0 ± 7.4	74.6 ± 7.2	81.2 ± 6.3	79.6 ± 6.0
Credit rating	84.2 ± 4.0	85.5 ± 3.7	70.4 ± 11.8●	81.0 ± 6.0	83.2 ± 4.6	73.8 ± 5.2 ●	85.4 ± 4.0	84.3 ± 4.1
German credit	71.1 ± 4.1	70.7 ± 4.7	60.1 ± 6.1 ●	66.6 ± 6.9	66.8 ± 4.8 ●	64.6 ± 5.1 ●	73.5 ± 4.1	71.1 ± 4.1
Pima I.D.	73.2 ± 4.1	73.7 ± 4.0	70.7 ± 7.1	71.1 ± 5.7	69.6 ± 7.1	71.1 ± 6.1	74.9 ± 4.4	73.5 ± 4.1
Glass	68.6 ± 10.5	58.4 ± 10.6●	59.1 ± 11.5●	51.4 ± 14.1●	61.7 ± 10.5	60.6 ± 12.0	68.2 ± 9.6	69.0 ± 9.9
Cleveland heart	78.1 ± 7.3	77.5 ± 7.0	72.3 ± 9.4	76.2 ± 9.4	74.2 ± 9.6	70.6 ± 8.8 ●	80.7 ± 7.4	78.0 ± 7.2
Hungarian heart	80.6 ± 7.3	78.6 ± 7.6	73.4 ± 10.9	74.0 ± 11.3	75.0 ± 9.9	73.6 ± 9.5 ●	78.5 ± 7.9	80.5 ± 7.4
Heart Statlog	77.4 ± 7.7	78.6 ± 8.3	72.1 ± 9.3	73.4 ± 9.1	73.2 ± 9.2	71.8 ± 8.6	79.3 ± 6.6	77.9 ± 8.0
Hepatitis	81.3 ± 10.6	80.2 ± 9.9	64.6 ± 17.0●	79.9 ± 11.1	78.1 ± 10.8	67.4 ± 16.0●	82.1 ± 9.4	81.5 ± 10.7
Labor	82.8 ± 12.6	82.1 ± 13.6	56.1 ± 25.4●	77.8 ± 18.8	76.9 ± 19.3	81.0 ± 17.5	81.5 ± 13.7	83.5 ± 12.6
Lymphography	75.8 ± 11.5	76.9 ± 11.0	69.3 ± 13.7	72.0 ± 12.5	73.0 ± 12.4	72.5 ± 11.1	76.6 ± 11.3	75.8 ± 11.5
Primary Tumor	40.3 ± 7.8	32.7 ± 8.2 ●	34.6 ± 8.4	27.3 ± 7.9 ●	37.4 ± 8.1	36.4 ± 8.8	39.6 ± 8.5	40.3 ± 7.8
Sonar	74.8 ± 10.7	71.3 ± 9.7	59.3 ± 12.0●	66.9 ± 10.6	71.7 ± 11.3	66.5 ± 11.1	76.5 ± 9.5	74.7 ± 10.9
Voting	95.0 ± 3.1	95.2 ± 3.0	91.0 ± 11.2	95.2 ± 3.3	94.7 ± 3.2	90.9 ± 5.8 ●	95.7 ± 2.7	95.0 ± 3.1
Zoo	92.8 ± 8.7	71.1 ± 14.7●	70.0 ± 15.4●	67.3 ± 15.7●	84.0 ± 14.1	92.2 ± 8.9	84.9 ± 13.6	92.8 ± 8.7
Average	77.0 ± 12.0	75.0 ± 13.5	67.0 ± 12.2	71.1 ± 14.7	73.1 ± 12.1	71.1 ± 12.5	77.2 ± 11.6	77.0 ± 12.0
Win/Tie/Lose		2/12/3	0/9/8	0/14/3	0/16/1	0/11/6	0/17/0	0/17/0
Wilcoxon <i>p</i> -value		0.181	0.000	0.000	0.000	0.000	0.136	0.274
Average compression [%]		9.0 ± 1.0	11.0 ± 4.9	12.4 ± 6.2	27.7 ± 9.5	46.0 ± 14.8	85.4 ± 9.8	94.6 ± 10.7

○, ● statistically significant improvement or degradation.

Rules generated for the Mushroom dataset may serve as an interesting example to see the influence of the EkP selection of reference vectors on C4.5 decision tree rules. Direct application of the C4.5 algorithm creates quite complex set of rules with odor, gill-size, ring-number, spore-print-color, stalk-shape, stalk-surface-below-ring, population and bruises used in their conditions. So far the best published set of rules that distinguish poisonous and edible mushrooms was [5]:

1. odor=NOT(almond.OR.anise.OR.none): Poisonous
2. spore-print-color=green: Poisonous
3. Else: Edible

Table 7. Average number of rules created by C4.5

Data Set	C4.5	EkP	ENRBF	DROP3	ICF	CNN	ENN	GE
Appendicitis	3.2 ± 0.7	2.0 ± 0.2 ●	1.8 ± 0.4 ●	2.0 ± 0.0 ●	2.3 ± 0.5 ●	1.9 ± 0.8 ●	3.6 ± 1.1	3.1 ± 0.6
Breast C.W.	18.7 ± 4.1	3.1 ± 0.4 ●	4.2 ± 1.9 ●	2.4 ± 0.8 ●	7.2 ± 2.2 ●	15.2 ± 3.7	11.2 ± 3.8 ●	17.5 ± 3.9
Horse Colic	20.0 ± 3.3	3.5 ± 0.7 ●	3.0 ± 1.0 ●	2.4 ± 0.7 ●	5.9 ± 2.0 ●	16.3 ± 3.1	11.3 ± 1.9 ●	20.0 ± 3.3
Credit rating	30.0 ± 3.4	5.1 ± 0.9 ●	4.3 ± 1.1 ●	3.9 ± 1.5 ●	11.4 ± 1.7 ●	23.9 ± 3.1	14.7 ± 2.0 ●	30.3 ± 3.2
German credit	69.1 ± 6.2	8.5 ± 2.1 ●	11.8 ± 2.3 ●	10.4 ± 2.2 ●	23.9 ± 3.4 ●	54.2 ± 5.5	38.9 ± 3.4 ●	69.1 ± 6.2
Pima I.D.	7.7 ± 1.7	4.8 ± 1.4 ●	5.1 ± 1.5 ●	5.7 ± 1.9 ●	3.7 ± 1.3 ●	3.6 ± 1.6 ●	9.5 ± 2.0	7.4 ± 1.6
Glass	15.5 ± 1.8	3.8 ± 0.7 ●	5.2 ± 0.6 ●	6.8 ± 1.4 ●	9.5 ± 0.9 ●	13.8 ± 2.2	10.5 ± 0.9 ●	15.6 ± 2.2
Cleveland heart	20.3 ± 2.8	3.9 ± 0.6 ●	4.1 ± 1.0 ●	4.4 ± 0.9 ●	8.4 ± 1.6 ●	16.1 ± 2.3	12.1 ± 1.8 ●	20.4 ± 3.0
Hungarian heart	16.1 ± 2.5	3.8 ± 1.2 ●	3.6 ± 1.1 ●	3.1 ± 1.1 ●	6.1 ± 1.7 ●	13.2 ± 2.3	9.6 ± 2.2 ●	15.9 ± 2.6
Heart Statlog	18.0 ± 2.5	4.0 ± 0.6 ●	3.4 ± 0.9 ●	3.6 ± 1.0 ●	8.3 ± 1.5 ●	14.5 ± 2.2	12.3 ± 1.9 ●	18.0 ± 2.5
Hepatitis	9.4 ± 1.6	2.2 ± 0.5 ●	2.4 ± 0.6 ●	2.3 ± 0.6 ●	3.1 ± 1.1 ●	8.0 ± 1.7	4.7 ± 1.2 ●	9.5 ± 1.6
Labor	3.4 ± 0.7	2.0 ± 0.0 ●	1.7 ± 0.6 ●	2.0 ± 0.1 ●	2.2 ± 0.5 ●	2.7 ± 0.7 ●	3.0 ± 0.7	3.4 ± 0.7
Lymphography	11.3 ± 2.1	3.0 ± 0.2 ●	3.5 ± 0.9 ●	2.7 ± 0.7 ●	6.0 ± 1.5 ●	8.4 ± 2.0	8.8 ± 1.9 ●	11.3 ± 2.1
Primary Tumor	46.7 ± 4.0	6.5 ± 1.2 ●	12.2 ± 1.7 ●	5.8 ± 1.7 ●	22.5 ± 2.7 ●	45.0 ± 3.9	24.7 ± 2.8 ●	46.7 ± 4.0
Sonar	7.3 ± 1.1	2.3 ± 0.4 ●	2.7 ± 0.7 ●	3.2 ± 0.7 ●	4.4 ± 0.6 ●	5.3 ± 0.9 ●	6.4 ± 0.9	7.3 ± 1.1
Voting	8.8 ± 2.6	3.8 ± 0.7 ●	3.2 ± 1.2 ●	2.0 ± 0.1 ●	5.0 ± 1.5 ●	7.6 ± 1.7	6.1 ± 1.8 ●	8.8 ± 2.6
Zoo	7.7 ± 0.5	3.0 ± 0.0 ●	3.9 ± 0.4 ●	3.7 ± 0.7 ●	6.3 ± 0.7 ●	7.6 ± 0.5	6.4 ± 0.7 ●	7.7 ± 0.5
Average	18.4 ± 16.9	3.8 ± 1.7	4.5 ± 3.0	3.9 ± 2.2	8.0 ± 6.2	15.1 ± 14.3	11.4 ± 8.7	18.3 ± 16.9
Win/Tie/Lose		0/0/17	0/0/17	0/0/17	0/0/17	0/6/11	0/4/13	0/17/0
Wilcoxon <i>p</i> -value		0.000	0.000	0.000	0.000	0.000	0.001	0.296

○, ● statistically significant improvement or degradation.

This set of rules has 99.4% accuracy in overall data reclassification, but it is quite robust and may also be found in crossvalidation tests. Slightly more accurate set of 3 rules was found combining EkP selection with C4.5 decision tree rules:

1. odor = none AND spore-print-color NOT green: Edible
2. odor NOT almond AND odor NOT anise: Poisonous
3. Default: Edible

These rules summarize the entire Mushroom dataset at the 99.7% accuracy level. For several other datasets similar excellent results may be demonstrated but are omitted here due to the lack of space.

4 Conclusions

Selection of training vectors is a powerful method that should be used more often, especially for very large datasets. The experiments presented in this paper compared results of training two inductive methods for generation of logical rules, NNGE and PART, on the full training data and on the data reduced by using several algorithms or vector selection. In particular recently introduced EkP system proved to be quite competitive, reducing the original dataset sometimes even by an order of magnitude, simplifying subsequent training and reducing the number of rules also by an order of magnitude, without significant reduction of accuracy. Rules generated in this way for the Mushroom database are surprisingly compact and easy to comprehend, the best found so far for this dataset.

It is clear that training on appropriately pruned data will be especially useful for very large datasets, giving hope to find solutions that are compact, accurate and easy to understand.

References

1. Asuncion, A., Newman, D.J.: UCI machine learning repository (2009), <http://www.ics.uci.edu/~mllearn/MLRepository.html>
2. Bhattacharya, B., Mukherjee, K., Toussaint, G.: Geometric decision rules for instance-based learning problems. In: Pal, S.K., Bandyopadhyay, S., Biswas, S. (eds.) PREMI 2005. LNCS, vol. 3776, pp. 60–69. Springer, Heidelberg (2005)
3. Blachnik, M., Duch, W.: Prototype rules from SVM. Springer Studies in Computational Intelligence, vol. 80, pp. 163–184. Springer, Heidelberg (2008)
4. Brighton, H., Mellish, C.: Advances in instance selection for instance-based learning algorithms. Data Mining and Knowledge Discovery 6(2), 153–172 (2002)
5. Duch, W., Adamczak, R., Grąbczewski, K.: A new methodology of extraction, optimization and application of crisp and fuzzy logical rules. IEEE Transactions on Neural Networks 12, 277–306 (2001)
6. Duch, W., Blachnik, M.: Fuzzy rule-based systems derived from similarity to prototypes. In: Pal, N.R., Kasabov, N., Mudi, R.K., Pal, S., Parui, S.K. (eds.) ICONIP 2004. LNCS, vol. 3316, pp. 912–917. Springer, Heidelberg (2004)
7. Duch, W., Grudziński, K.: Prototype based rules - new way to understand the data. In: IEEE International Joint Conference on Neural Networks, Washington DC, pp. 1858–1863. IEEE Press, Los Alamitos (2001)
8. Duch, W., Setiono, R., Zurada, J.: Computational intelligence methods for understanding of data. Proceedings of the IEEE 92(5), 771–805 (2004)
9. Frank, E., Witten, I.H.: Generating accurate rule sets without global optimization. In: Shavlik, J. (ed.) Machine Learning: Proceedings of the Fifteenth International Conference. Morgan Kaufmann Publishers, San Francisco (1998)
10. Grochowski, M., Jankowski, N.: Comparison of instance selection algorithms. II. Results and comments. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) ICAISC 2004. LNCS (LNAI), vol. 3070, pp. 580–585. Springer, Heidelberg (2004)
11. Grudziński, K.: *E_kP*: A fast minimization-based prototype selection algorithm. In: Intelligent Information Systems XVI, pp. 45–53. Academic Publishing House EXIT, Warsaw (2008)
12. Grudziński, K.: Selection of prototypes with the *E_kP* system. Control and Cybernetics (submitted)
13. Hart, P.E.: The condensed nearest neighbor rule. IEEE Transactions on Information Theory 14, 515–516 (1968)
14. Jankowski, N.: Data regularization. In: Rutkowski, L., Tadeusiewicz, R. (eds.) Neural Networks and Soft Computing, pp. 209–214 (2000)
15. Jankowski, N., Grochowski, M.: Comparison of instance selection algorithms. I. Algorithms survey. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) ICAISC 2004. LNCS (LNAI), vol. 3070, pp. 598–603. Springer, Heidelberg (2004)
16. Nelder, J., Mead, R.: A simplex method for function minimization. Computer Journal 7, 308–313 (1965)
17. Quinlan, J.R.: C 4.5: Programs for machine learning. Morgan Kaufmann, San Mateo (1993)
18. Wilson, D.L.: Asymptotic properties of nearest neighbor rules using edited data. IEEE Trans. Systems, Man and Cybernetics 2, 408–421 (1972)
19. Wilson, D.R., Martinez, T.R.: Reduction techniques for instance-based learning algorithms. Machine Learning 38(3), 257–286 (2000)
20. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)

Sensitivity and Specificity for Mining Data with Increased Incompleteness

Jerzy W. Grzymala-Busse^{1,2} and Shantanu R. Marepally¹

¹ Department of Electrical Engineering and Computer Science University of Kansas,
Lawrence, KS 66045, USA

² Institute of Computer Science, Polish Academy of Sciences,
01-237 Warsaw, Poland
jerzy@ku.edu, shantanu@ku.edu

Abstract. This paper presents results of experiments on data sets that were subjected to increasing incompleteness by random replacement of attribute values by symbols of missing attribute values. During these experiments the total error rate and error rates for all concepts, results of repeated 30 times ten-fold cross validation, were recorded. We observed that for some data sets increased incompleteness might result in a significant improvement for the total error rate and sensitivity (with the significance level of 5%, two-tailed test). These results may be applied for improving data mining techniques, especially for domains in which sensitivity is important, e.g., in medical area.

1 Introduction

Our previous research on mining data sets with increased incompleteness, in which some attribute values were removed incrementally, was based on the total error rate, i.e., the ratio of total number of errors to the total number of cases in the data set. It was shown that for some data sets increased incompleteness causes smaller total error rate [1,2,3,4].

The question is how increasing incompleteness changes concept error rates (i.e., error rates restricted to a concept). A *concept* (class) is a set of all cases classified (or diagnosed) the same way. Therefore, we decided to conduct additional research on concept error rates for data sets with increased incompleteness. This research has important applications to medical area. In medical area the quality of a rule set is measured by two concept accuracies called *sensitivity* and *specificity*. One of the concepts is more important and is called *primary*, e.g., describing a disease or another medical emergency, while the complementary concept, called *secondary*, is less important, since patients not affected by the disease do not need any medical help.

Sensitivity is the ratio of correctly classified cases of the primary concept to the total number of cases in primary concept. Similarly, *specificity* is the ratio of correctly classified patients of the secondary concept to the total number of cases in the secondary concept.

Table 1. Breast cancer data set, total errors, certain rule sets

Percentage of lost values	Average error rate	Standard deviation	Z score
0	28.63	0.45	–
5	28.40	0.58	–1.72
10	28.44	0.77	–1.17
15	28.51	0.93	–0.64
20	28.98	0.73	2.24
25	29.77	0.91	6.15
30	28.33	0.77	–1.84
35	27.91	1.00	–3.60
40	29.84	1.06	5.76
45	27.58	0.92	–5.62

Table 2. Breast cancer data set, errors for no recurrence events, certain rule sets

Percentage of lost values	Average error rate	Standard deviation	Z score
0	0.68	0.28	–
5	1.09	0.54	3.69
10	1.96	0.71	9.19
15	1.72	0.90	6.04
20	2.50	1.00	9.60
25	4.52	1.28	16.05
30	2.52	0.96	10.08
35	3.44	0.98	14.83
40	5.41	0.96	25.91
45	3.88	0.87	19.18

In this paper, we investigate the concept error rates, for all concepts, as the ratio of incorrectly classified cases from the concept to the total number of cases in the concept. For data with two concepts, there exists an obvious rule to convert concept error rates into sensitivity and specificity. Sensitivity and specificity, in percents, are equal to the difference between 100 and the error rate, also in percents, for the primary concept and secondary concept, respectively.

Our main objective was to study concept error rates for every concept of a given data set while increasing incompleteness of the data set. Our experiments indicate that for some data sets sensitivity improves while increasing incompleteness, for other data sets specificity improves under the same circumstances.

For rule induction from incomplete data, we used the MLEM2 data mining algorithm, for details see [5]. We used rough set methodology [6], i.e., for a given interpretation of missing attribute values, *lower* and *upper approximations* were computed for all concepts and then rule sets were induced, *certain* rules from lower approximations and *possible* rules from upper approximations. Note that for incomplete data there is a few possible ways to define approximations, we used *concept approximations* [7].

Three different kinds of missing attribute values were used in our experiments: *lost values* (the values that were recorded but currently are unavailable) [8,7], *attribute-concept values* (these missing attribute values may be replaced by any attribute value limited to the same concept) [7], and *"do not care" conditions* (the original values were irrelevant) [9].

We assumed that for each case at least one attribute value was specified, i.e., was not missing. Such an assumption limits the percentage of missing attribute values used for experiments; for example, for the *bankruptcy* data set, starting from 40% of randomly assigned missing attribute values, this assumption was violated. Additionally, we assumed that all decision values were specified.

2 Experiments

This paper presents our experiments conducted on three typical data sets. Two of these data sets, *breast* and *hepatitis* are available on the UCI ML Repository.

For *bankruptcy* data set there exist two concepts: *bankruptcy* and *survival*, the former is primary, the latter is secondary. For the *breast cancer* data set, the primary concept is *recurrence*, the secondary concept is *no recurrence*. For *hepatitis* data set, the primary concept is *die* while the secondary concept is *live*.

Table 3. Breast cancer data set, errors for recurrence events, certain rule sets

Percentage of lost values	Average error rate	Standard deviation	Z score
0	96.26	1.51	—
5	94.49	1.30	-4.87
10	92.51	2.18	-7.75
15	93.33	2.56	-5.40
20	93.05	2.33	-6.33
25	90.86	2.40	-10.43
30	90.78	2.76	-9.54
35	87.12	2.93	-15.19
40	88.97	2.85	-12.38
45	84.94	2.73	-19.87

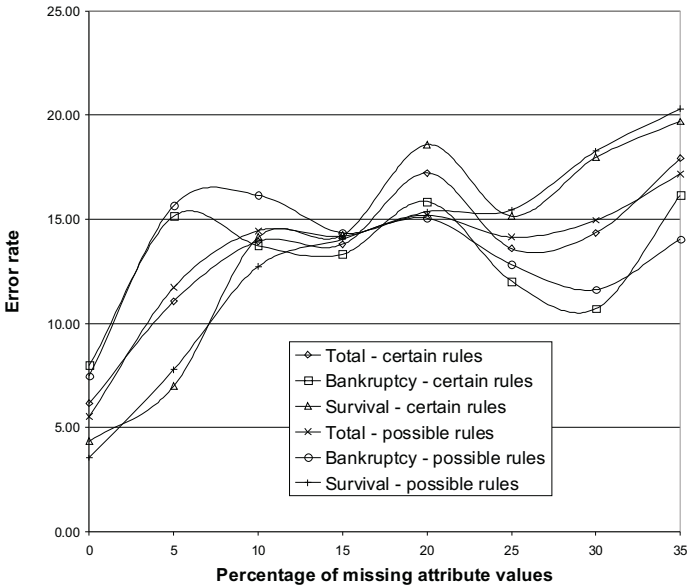


Fig. 1. Bankruptcy data set, lost values

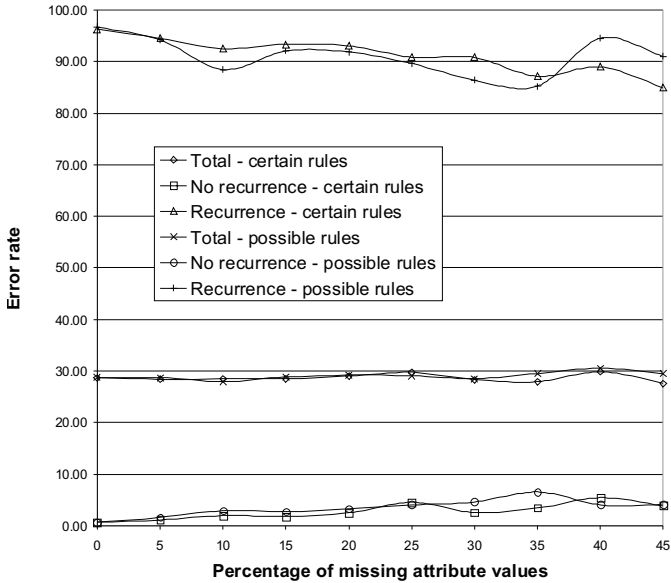


Fig. 2. Breast cancer data set, lost values

Results of our experiments, based on ten-fold cross validation repeated 30 times, are presented on Tables 1–3 and Figures 1–7. Tables 1–3 present Z-scores [4], if the Z score is smaller than -1.96 , the rule set induced from the data set with given percentage of missing attribute values is significantly better than the

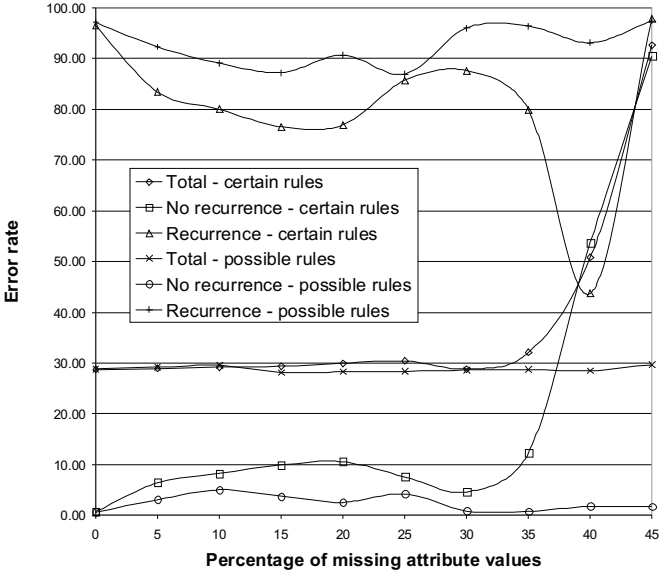


Fig. 3. Breast cancer data set, “do not care” conditions

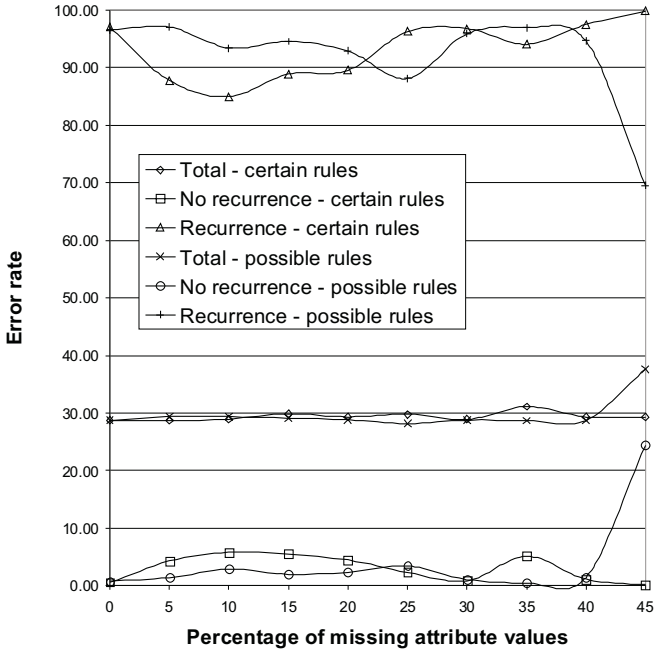


Fig. 4. Breast cancer data set, attribute-concept values

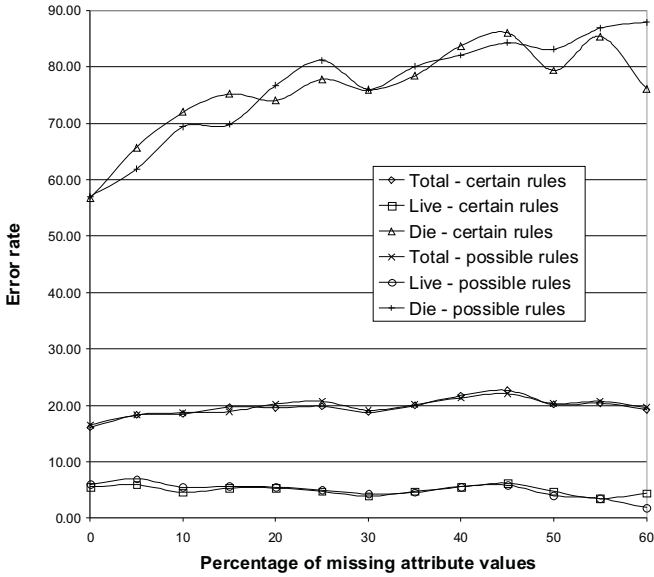


Fig. 5. Hepatitis data set, lost values

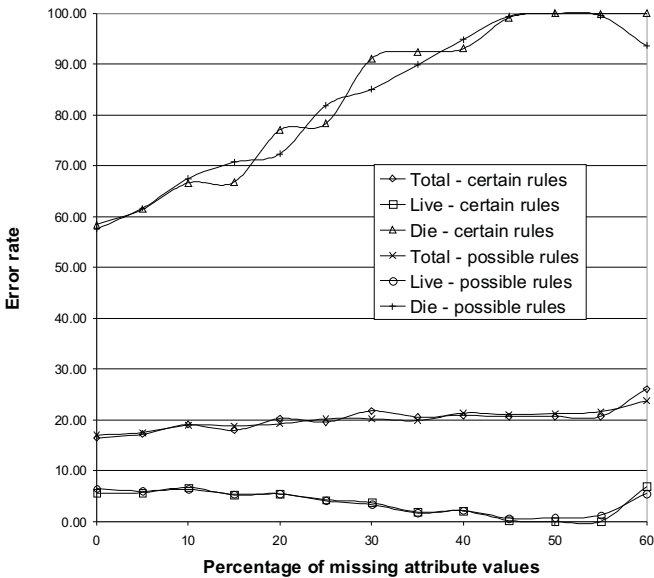


Fig. 6. Hepatitis data set, "do not care" conditions

corresponding rules set induced from the original data set, with the significance level of 5%, two-tailed test. As follows from Table 1, there are two certain rule sets better than the certain rule sets induced from the original data sets, for 35% and 45% of missing attribute values. Similarly, sensitivity for all certain rule sets

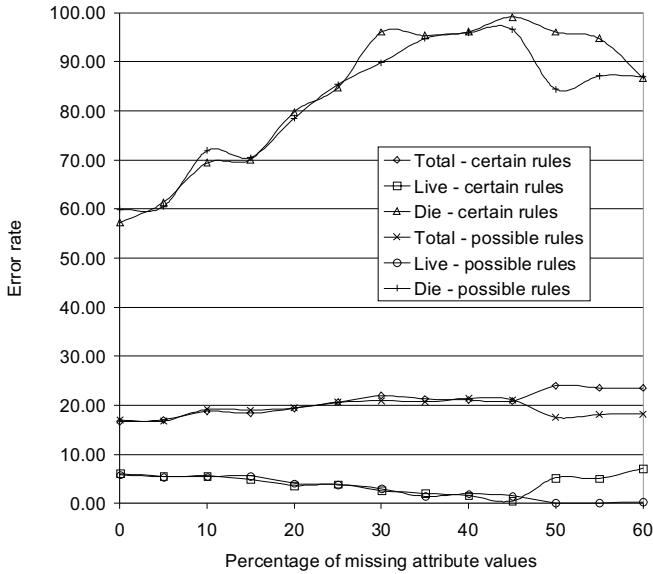


Fig. 7. Hepatitis data set, attribute-concept values

induced from data sets with some missing attribute values is better than for the sensitivity for the original data set. In Tables 1 and 3, the corresponding Z scores are presented in bold font.

3 Conclusions

As follows from our experiments, for some data sets, with increased incompleteness, sensitivity (a result of ten-fold cross validation) may significantly increase (with the significance level of 5%, two-tailed test) comparing with the sensitivity for the rule set induced from the original data set.

Our experiments show that among data sets there exist three basic types:

- data sets, such as *bankruptcy* in which increasing incompleteness increases total error rate and decreases sensitivity and specificity,
- data sets, such as *breast cancer* where increasing incompleteness may cause decreasing total error rate and improvement for sensitivity (on the cost of slightly worse specificity),
- data sets, such as *hepatitis*, where increasing incompleteness causes slightly increasing total error rate, worsening sensitivity and improving specificity.

Practically it means that there exists yet another technique for improving sensitivity based on replacing existing attribute values by symbols of missing attribute values or, in other words, on removing attribute values. A possible explanation for an occasional improvement of the quality of rule sets is redundancy of information in some data sets, such as the *breast cancer* data set, so that it is still

possible to induce not only good but sometimes even better rule sets than the rule set induced from the original data set.

Note that we conducted many other experiments supporting our conclusions but we cannot present these results here because of the space limit. We picked the most representative three data sets.

References

1. Grzymala-Busse, J.W., Grzymala-Busse, W.J.: An experimental comparison of three rough set approaches to missing attribute values. *Transactions on Rough Sets* 6, 31–50 (2007)
2. Grzymala-Busse, J.W., Grzymala-Busse, W.J., Hippe, Z.S., Rzasa, W.: An improved comparison of three rough set approaches to missing attribute values. In: *Proceedings of the 16th International Conference on Intelligent Information Systems*, pp. 141–150 (2008)
3. Grzymala-Busse, J.W., Grzymala-Busse, W.J.: Improving quality of rule sets by increasing incompleteness of data sets. In: *Proceedings of the Third International Conference on Software and Data Technologies*, pp. 241–248 (2008)
4. Grzymala-Busse, J.W., Grzymala-Busse, W.J.: Inducing better rule sets by adding missing attribute values. In: *Proceedings of the Sixth International Conference Rough Sets and Current Trends in Computing*, pp. 160–169 (2008)
5. Grzymala-Busse, J.W.: MLEM2: A new algorithm for rule induction from imperfect data. In: *Proceedings of the 9-th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pp. 243–250 (2002)
6. Pawlak, Z.: *Rough Sets*. In: *Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, Dordrecht (1991)
7. Grzymala-Busse, J.W.: Three approaches to missing attribute values—a rough set perspective. In: *Proceedings of the Workshop on Foundation of Data Mining, in conjunction with the Fourth IEEE International Conference on Data Mining*, pp. 55–62 (2004)
8. Stefanowski, J., Tsoukias, A.: Incomplete information tables and rough classification. *Computational Intelligence* 17(3), 545–566 (2001)
9. Kryszkiewicz, M.: Rules in incomplete information systems. *Information Sciences* 113(3-4), 271–292 (1999)

A New Implementation of the co-VAT Algorithm for Visual Assessment of Clusters in Rectangular Relational Data

Timothy C. Havens, James C. Bezdek, and James M. Keller

Department of Electrical and Computer Engineering,
University of Missouri
Columbia, MO 65211, USA
havenst@gmail.com, jbezdek@gmail.com,
kellerj@missouri.edu

Abstract. This paper presents a new implementation of the co-VAT algorithm. We assume we have an $m \times n$ matrix \mathbf{D} , where the elements of \mathbf{D} are pair-wise dissimilarities between m row objects O_r and n column objects O_c . The union of these disjoint sets are $(N = m + n)$ objects O . Clustering tendency assessment is the process by which a data set is analyzed to determine the number(s) of clusters present. In 2007, the *co-Visual Assessment of Tendency* (co-VAT) algorithm was proposed for rectangular data such as these. co-VAT is a visual approach that addresses four clustering tendency questions: i) How many clusters are in the row objects O_r ? ii) How many clusters are in the column objects O_c ? iii) How many clusters are in the union of the row and column objects $O_r \cup O_c$? And, iv) How many (co)-clusters are there that contain at least one of each type? co-VAT first imputes pair-wise dissimilarity values among the row objects, the square relational matrix \mathbf{D}_r , and the column objects, the square relational matrix \mathbf{D}_c , and then builds a larger square dissimilarity matrix $\mathbf{D}_{r \cup c}$. The clustering questions can then be addressed by using the VAT algorithm on \mathbf{D}_r , \mathbf{D}_c , and $\mathbf{D}_{r \cup c}$; \mathbf{D} is reordered by shuffling the reordering indices of $\mathbf{D}_{r \cup c}$. Subsequently, the co-VAT image of \mathbf{D} may show tendency for co-clusters (problem iv). We first discuss a different way to construct this image, and then we also extend a path-based distance transform, which is used in the iVAT algorithm, to co-VAT. The new algorithm, co-iVAT, shows dramatic improvement in the ability of co-VAT to show cluster tendency in rectangular dissimilarity data.

1 Introduction

Consider a set of objects $O = \{o_1, \dots, o_N\}$. These objects can represent virtually anything—vintage bass guitars, pure-bred cats, cancer genes expressed in a microarray experiment, cake recipes, or web-pages. The object set O is *unlabeled data*; that is, each object has no associated class label. However, it is assumed that there are subsets of similar objects in O . These subsets are called *clusters*.

Numerical object data is represented as $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\} \subset \mathbb{R}^p$, where each dimension of the vector \mathbf{x}_i is a feature value of the associated object o_i . These features can be a veritable cornucopia of numerical descriptions, i.e., RGB values, gene expression, year of manufacture, number of stripes, etc. Another way to represent the

objects in O is with numerical *relational* data, which consist of N^2 values that represent the (dis)similarity between pairs of objects. These data are commonly represented by a relational matrix $\mathbf{R} = [r_{ij} = \text{relation}(o_i, o_j) | 1 \leq i, j \leq N]$. The relational matrix often takes the form of a *dissimilarity* matrix \mathbf{D} . Dissimilarity can be interpreted as a distance between objects. For instance, numerical data X can always be converted to \mathbf{D} by $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|$ (any vector norm on \mathbb{R}^p). There are, however, similarity and dissimilarity relational data sets that do not begin as numerical object data; for these, there is no choice but to use a relational algorithm. Hence, relational data represent the “most general” form of input data.

An even more general form of relational data is *rectangular*. These data are represented by an $m \times n$ dissimilarity matrix \mathbf{D} , where the entries are the pair-wise dissimilarity values between m row objects O_r and n column objects O_c . An example comes from web-document analysis, where the row objects are m web-pages, the columns are n words, and the (dis)similarity entries are occurrence measures of words in web-pages [1]. In each case, the row and column objects are non-intersecting sets, such that the pair-wise relation among row (or column) objects is unknown. Conventional relational clustering algorithms are ill-equipped to deal with rectangular data. Additionally, the definition of a cluster as a group of similar objects takes on a new meaning. There can be groups of similar objects that are composed of only row objects, of only column objects, or of mixed objects (often called *co-clusters*). In this paper we consider these four types of clusters in rectangular data.

Clustering is the process of grouping the objects in O in a sensible manner. This process is often performed to elucidate the similarity and dissimilarity among and between the grouped objects. Clustering has also been called unsupervised learning, typology, and partitioning [2]. Although clustering is typically thought of as only the act of separating objects into the proper groups, cluster analysis actually consists of three concise questions: i) *cluster tendency*—how many clusters are there?; ii) *partitioning*—which objects belong to which cluster and to what degree?; and iii) *cluster validity*—are the partitions “good”? The VAT [3] and co-VAT [4] algorithms address problem i).

1.1 VAT and iVAT Algorithms

The VAT algorithm displays an image of reordered and scaled dissimilarity data [3]. Each pixel of the grayscale VAT image $I(\mathbf{D}^*)$ displays the scaled dissimilarity value of two objects. White pixels represent high dissimilarity, while black represents low dissimilarity. Each object is exactly similar with itself, which results in zero-valued (black) diagonal elements of $I(\mathbf{D}^*)$. The off-diagonal elements of $I(\mathbf{D}^*)$ are scaled to the range $[0, 1]$. A dark block along the diagonal of $I(\mathbf{D}^*)$ is a sub-matrix of “similarly small” dissimilarity values; hence, the dark block represents a cluster of objects that are relatively similar to each other. Thus, cluster tendency is shown by the number of dark blocks along the diagonal of the VAT image.

The VAT algorithm is based on Prim’s algorithm [5] for finding the *minimum spanning tree* (MST) of a weighted connected graph [3]. Algorithm 1 illustrates the steps of the VAT algorithm. The resulting VAT-reordered dissimilarity matrix \mathbf{D}^* can be normalized and mapped to a gray-scale image with black representing the minimum dissimilarity and white the maximum.

Algorithm 1. VAT Ordering Algorithm [3]

Input: \mathbf{D} — $n \times n$ dissimilarity matrix
Data: $\mathbf{K} = \{1, 2, \dots, n\}$; $\mathbf{I} = \mathbf{J} = \emptyset$; $P = (0, 0, \dots, 0)$.
 Select $(i, j) \in \arg \max_{p \in \mathbf{K}, q \in \mathbf{K}} D_{pq}$.
 Set $P(1) = i$; $\mathbf{I} = \{i\}$; and $\mathbf{J} = \mathbf{K} - \{i\}$.
for $r = 2, \dots, n$ **do**
 Select $(i, j) \in \arg \min_{p \in \mathbf{I}, q \in \mathbf{J}} D_{pq}$.
 Set $P(r) = j$; Replace $\mathbf{I} \leftarrow \mathbf{I} \cup \{j\}$ and $\mathbf{J} \leftarrow \mathbf{J} - \{j\}$.
 Obtain the ordered dissimilarity matrix \mathbf{D}^* using the ordering array P as:
 $D_{pq}^* = D_{P(p), P(q)}$, for $1 \leq p, q \leq n$.
Output: Reordered dissimilarity \mathbf{D}^*

Algorithm 2. Recursive calculation of iVAT image

Input: \mathbf{D}^* - VAT-reordered dissimilarity matrix
Data: $\mathbf{D}'^* = [0]^{n \times n}$
for $r = 2, \dots, n$ **do**
 1 $j = \arg \min_{k=1, \dots, r-1} D_{rk}^*$
 2 $D_{rc}^* = D_{rc}^*$, $c = j$
 3 $D_{rc}^* = \max \{D_{rj}^*, D_{jc}^*\}$, $c = 1, \dots, r-1, c \neq j$
 \mathbf{D}'^* is symmetric, thus $D_{rc}^* = D_{cr}^*$.

Reference [6] proposed an *improved* VAT (iVAT) algorithm that uses a path-based distance measure from [7]. Consider \mathbf{D} to represent the weights of the edges of a fully-connected graph. The path-based distance is defined as

$$D'_{ij} = \min_{p \in P_{ij}} \max_{1 \leq h < |p|} D_{p[h]p[h+1]}, \tag{1}$$

where $p \in P_{ij}$ is an acyclic path in the set of all acyclic paths between vertex i (o_i) and vertex j (o_j), $p[h]$ is the index of the h th vertex along path p , and $|p|$ is the number of vertexes along the path. Hence, $D_{p[h]p[h+1]}$ is the weight of the h th edge along path p . Essentially the cost of each path p is the maximum weight of its $|p|$ edges. The distance between i and j is the cost of the minimum-cost path in P_{ij} . The authors of [6] first transform \mathbf{D} into \mathbf{D}' with [1], then they use VAT on the transformed dissimilarity matrix. The iVAT images show considerable improvement over VAT images in showing the cluster tendency for “tough” cases. Note that computing \mathbf{D}' exhaustively is very computationally expensive. We can show that i) the iVAT dissimilarity matrix \mathbf{D}' can be computed recursively from the VAT-reordered data \mathbf{D}^* (see Algorithm 2) and, ii) the matrix \mathbf{D}' computed by Algorithm 2 is already in VAT-order (we have an article in preparation that includes proofs of these assertions). We will denote \mathbf{D}' as \mathbf{D}'^* to indicate that it is a VAT-ordered matrix. Essentially, iVAT is a distance transform that improves the visual contrast of the dark blocks along the VAT diagonal.

1.2 co-VAT Algorithm

The co-VAT algorithm begins by creating a square matrix $\mathbf{D}_{r \cup c}$, part of which is composed of the rectangular dissimilarity matrix \mathbf{D} . $\mathbf{D}_{r \cup c}$ is created by first estimating the

dissimilarity matrices \mathbf{D}_r and \mathbf{D}_c , which are, respectively, square dissimilarity matrices that relate the objects in \mathbf{O}_r and \mathbf{O}_c to themselves — i.e. $[D_r]_{ij} \approx d(o_i, o_j)$ and $[D_c]_{ij} \approx d(o_{m+i}, o_{m+j})$. $\mathbf{D}_{r \cup c}$ is organized as in Eq. (2).

$$\mathbf{D}_{r \cup c} = \begin{bmatrix} \mathbf{D}_r & \mathbf{D} \\ \mathbf{D}^T & \mathbf{D}_c \end{bmatrix} \approx \begin{bmatrix} \begin{bmatrix} d(o_1, o_1) & \cdots & d(o_1, o_m) \\ \vdots & \ddots & \vdots \\ d(o_m, o_1) & \cdots & d(o_m, o_m) \end{bmatrix} & \begin{bmatrix} d(o_1, o_{m+1}) & \cdots & d(o_1, o_{m+n}) \\ \vdots & \ddots & \vdots \\ d(o_m, o_{m+1}) & \cdots & d(o_m, o_{m+n}) \end{bmatrix} \\ \begin{bmatrix} d(o_1, o_{m+1}) & \cdots & d(o_m, o_{m+n}) \\ \vdots & \ddots & \vdots \\ d(o_1, o_{m+n}) & \cdots & d(o_m, o_{m+n}) \end{bmatrix} & \begin{bmatrix} d(o_{m+1}, o_{m+1}) & \cdots & d(o_{m+1}, o_{m+n}) \\ \vdots & \ddots & \vdots \\ d(o_{m+n}, o_{m+1}) & \cdots & d(o_{m+n}, o_{m+n}) \end{bmatrix} \end{bmatrix} \quad (2)$$

The elements in \mathbf{D}_r and \mathbf{D}_c are estimated from \mathbf{D} using any vector norms on \mathbb{R}^n and \mathbb{R}^m ,

$$[\mathbf{D}_r]_{ij} = \lambda_r \|\mathbf{d}_{i*} - \mathbf{d}_{j*}\|, \quad 1 \leq i, j \leq m, \quad (3)$$

$$[\mathbf{D}_c]_{ij} = \lambda_c \|\mathbf{d}_{*i} - \mathbf{d}_{*j}\|, \quad 1 \leq i, j \leq n, \quad (4)$$

where \mathbf{d}_{i*} is the i th row of \mathbf{D} , \mathbf{d}_{*j} is the j th column of \mathbf{D} , and λ_r and λ_c are scale factors such that the mean of the off-diagonal elements of \mathbf{D}_r and \mathbf{D}_c match the mean of \mathbf{D} .

Section 2.1 presents a new method for finding the reordering of the rectangular dissimilarity matrix \mathbf{D} . Section 2.2 adapts the iVAT distance transform to co-VAT, which shows improved performance over the standard co-VAT method. Section 3 presents a numerical example of the new implementations of co-VAT and Section 4 concludes this paper.

2 New Implementations of co-VAT

2.1 Alternate co-VAT Reordering Scheme

The original co-VAT algorithm, outlined in Algorithm 3, reorders the rectangular matrix \mathbf{D} by shuffling the VAT-reordering indexes of $\mathbf{D}_{r \cup c}$. Thus, co-VAT is very dependent on the construction of $\mathbf{D}_{r \cup c}$. We have discovered that the original co-VAT fails to show cluster tendency in certain cases; we have an upcoming paper that discusses these cases in detail. Algorithm 4 presents a reordering scheme that is not dependent on the reordering of $\mathbf{D}_{r \cup c}$ — this matrix does not even need to be constructed. However, we still need the matrix of the union if we intend to assess cluster tendency in $O_{r \cup c}$. Essentially, the reordering of the row indexes of \mathbf{D} are taken from the VAT-reordering of \mathbf{D}_r and the reordering of the column indexes are taken from the VAT-reordering of \mathbf{D}_c . Another advantage of this alternate reordering scheme is that the scale factors, λ_r and λ_c in (3) and (4), can be ignored.

Algorithm 3. co-VAT Algorithm [4]**Input:** \mathbf{D} - $m \times n$ rectangular dissimilarity matrixBuild estimates of \mathbf{D}_r and \mathbf{D}_c using Eqs. (3) and (4), respectively.Build $\mathbf{D}_{r \cup c}$ using Eq. (2).Run VAT on $\mathbf{D}_{r \cup c}$, saving permutation array $P_{r \cup c} = \{P(1), \dots, P(m+n)\}$ **Initialize** $rc = cc = 0$; $RP = CP = 0$.

- ```

1 for $t = 1, \dots, m+n$ do
2 if $P(t) \leq m$ then
3 $rc = rc + 1$, rc is row component
4 $RP(rc) = P(t)$, RP are row indexes
 else
5 $cc = cc + 1$, cc is column component
6 $CP(cc) = P(t) - m$, CP are column indexes

```

Form the co-VAT ordered rectangular dissimilarity matrix,

 $\mathbf{D}^* = [D_{ij}^*] = [D_{RP(i)CP(j)}]$ ,  $1 \leq i \leq m$ ;  $1 \leq j \leq n$ **Output:** Reordered dissimilarity matrices  $\mathbf{D}^*$ ,  $\mathbf{D}_r^*$ ,  $\mathbf{D}_c^*$ , and  $\mathbf{D}_{r \cup c}^*$ **Algorithm 4.** Alternate co-VAT Reordering Scheme**Input:**  $\mathbf{D}$  -  $m \times n$  rectangular dissimilarity matrixBuild estimates of  $\mathbf{D}_r$  and  $\mathbf{D}_c$  using Eqs. (3) and (4), respectively.

- 1 Run VAT on  $\mathbf{D}_r$ , saving permutation array,  $RP = \{RP(1), \dots, RP(m)\}$
- 2 Run VAT on  $\mathbf{D}_c$ , saving permutation array,  $CP = \{CP(1), \dots, CP(n)\}$
- 3 Form the co-VAT ordered rectangular dissimilarity matrix,

 $\mathbf{D}^* = [D_{ij}^*] = [D_{RP(i)CP(j)}]$ ,  $1 \leq i \leq m$ ;  $1 \leq j \leq n$ **Output:** Reordered dissimilarity matrices,  $\mathbf{D}^*$ ,  $\mathbf{D}_r^*$ , and  $\mathbf{D}_c^*$ **2.2 Using the iVAT Distance Transform in co-VAT**

We apply the iVAT distance transform in (1) to the three square co-VAT matrices,  $\mathbf{D}_r^*$ ,  $\mathbf{D}_c^*$ , and  $\mathbf{D}_{r \cup c}^*$ , using the recursive formulation in Algorithm 2. We denote the transformed matrices as  $\mathbf{D}_r^{i*}$ ,  $\mathbf{D}_c^{i*}$ , and  $\mathbf{D}_{r \cup c}^{i*}$  (examples of these matrices are shown in Figs. 2(a,c,d), respectively). Although, by definition, (1) could applied to the rectangular dissimilarity matrix  $\mathbf{D}$  by considering  $\mathbf{D}$  to represent a partially-connected graph (edges only exist between row objects and column objects), applying this transform directly is computationally expensive. However, if we consider the elements of  $\mathbf{D}_{r \cup c}^{i*}$  that correspond to elements of the rectangular matrix, we can build the reordered rectangular matrix  $\mathbf{D}^{i*}$  from  $\mathbf{D}_{r \cup c}^{i*}$ .

The rectangular co-iVAT image is created as follows:

1. Build  $\mathbf{D}_{r \cup c}$  and run VAT to produce  $\mathbf{D}_{r \cup c}^*$ , where the reordering indexes are  $P_{r \cup c} = \{P(1), \dots, P(m+n)\}$ .
2. Compute  $\mathbf{D}_{r \cup c}^{i*}$  from  $\mathbf{D}_{r \cup c}^*$  using the recursive iVAT distance transform outlined in Algorithm 2.



3. Build the rectangular co-iVAT image  $\mathbf{D}'$  from the corresponding elements of  $\mathbf{D}'_{r \cup c}$ . First, create the reordering indexes  $K$  and  $L$ , where  $K$  are the indexes of the elements of  $P_{r \cup c} \leq m$  and  $L$  are the indexes of the elements of  $P_{r \cup c} > m$  ( $m$  is the number of row objects). Then create  $\mathbf{D}'^*$  by

$$\mathbf{D}'^* = [D'_{ij}^*] = \left[ (D'_{r \cup c})_{K(i),L(j)} \right], 1 \leq i \leq m, 1 \leq j \leq n. \tag{5}$$

We have also adapted iVAT to the new reordering scheme of co-VAT presented in Algorithm 4. This adaptation requires the construction of  $\mathbf{D}'_{r \cup c}$ , as above, and the corresponding elements of  $\mathbf{D}'_{r \cup c}$  are extracted to create the reordered rectangular matrix  $\mathbf{D}'^*$ . We denote the co-VAT matrices built with the iVAT distance transform as co-iVAT images.

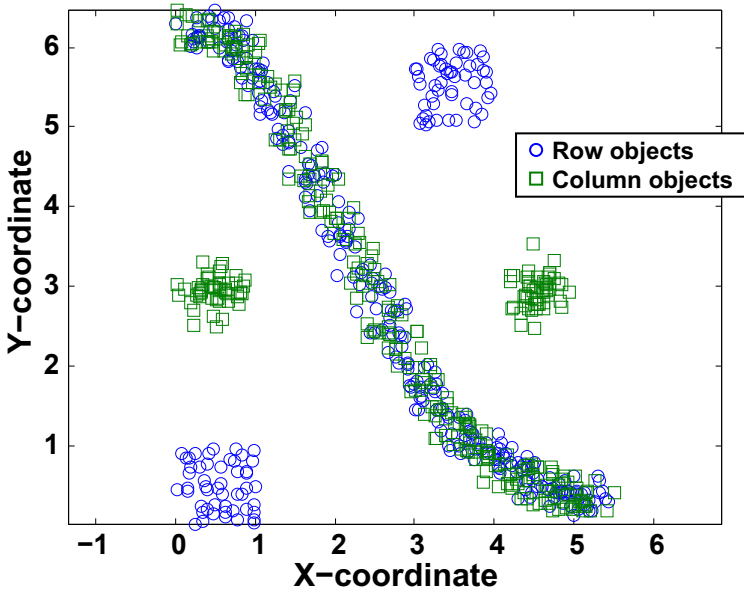
Next we present an example that shows the effectiveness of the new implementations of co-VAT.

### 3 Numerical Example

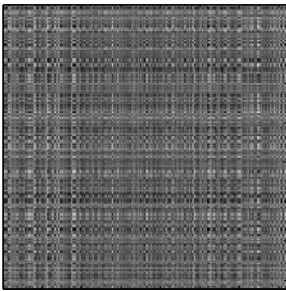
Our numerical example is composed of 360 row objects and 360 column objects, as displayed in Fig. 1(a). Note that, although the number of row objects  $N$  equals the number column objects  $M$ , this data set is rectangular data because  $O_r \cap O_c = \emptyset$ . The associated dissimilarity data, calculated using Euclidean distance, is shown in Fig. 1(b). The column objects (shown as green squares) are composed of three groups, the two groups of 50 objects located around coordinates (1.5,3) and (4.5,3), and the 260 objects organized along the curved line extending from the upper-left to the lower-right. The row objects (shown as blue circles) are composed of three groups, the two groups of 50 objects located around coordinates (1.5,0.5) and (3.5,5.5), and the 260 objects organized along the curved line extending from the upper-left to the lower-right. Hence, this example has a preferable cluster tendency of 3 clusters of row objects, 3 clusters composed of column objects, 5 clusters in the union of the row and column objects, and 1 co-cluster.

Figure 1(c,d) shows that both co-VAT and the new co-VAT display the 1 co-cluster as a diagonal band in the upper-left of the image, with both giving approximately equally pleasing results. The co-VAT images of  $\mathbf{D}'_r$  and  $\mathbf{D}'_c$  in Figs. 1(e,f), respectively, clearly show the smaller 2 clusters in each of the row objects and column objects as 2 smaller dark blocks in the lower-right of the image. Again the large co-cluster is shown as a dark diagonal band in the upper-left. The image of  $\mathbf{D}'_{r \cup c}$  is shown in Fig. 1(g); it shows the 5 clusters in  $O_r \cup O_c$  as the four dark blocks in the lower-right and the dark diagonal band in the upper-left. While we hesitate to say that co-VAT and the new co-VAT have failed for this example, we believe that the large diagonal band leads to ambiguity as to the cluster tendency of these data. Further more, the contrast in Figs. 1(c,d) make it difficult to determine the number of co-clusters.

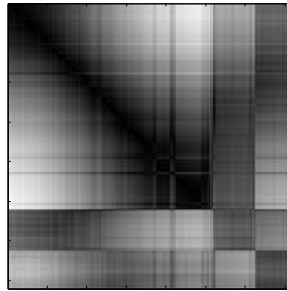
Figure 2 shows the corresponding co-iVAT images of the rectangular dissimilarity data shown in Fig. 1. The co-iVAT images give a very clear view of the cluster tendency for each of the four types of clusters:  $\mathbf{D}'^*$  shows 1 co-cluster,  $\mathbf{D}'_r$  shows 3 row-clusters,  $\mathbf{D}'_c$  shows 3 column-clusters, and  $\mathbf{D}'_{r \cup c}$  shows 5 clusters in  $O_r \cup O_c$ .



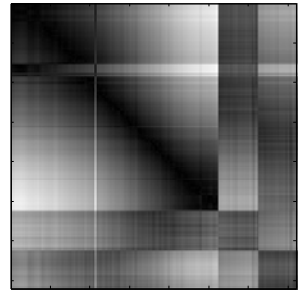
(a) Object Data



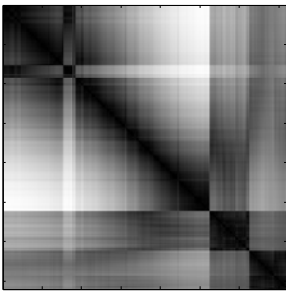
(b) Dissimilarity Data -  $D$



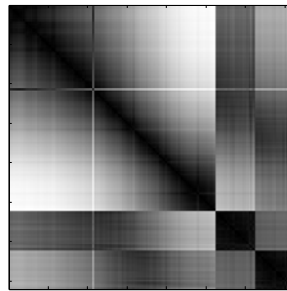
(c) co-VAT image -  $D^*$



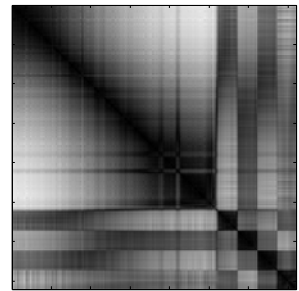
(d) New co-VAT image -  $D^*$



(e) co-VAT image -  $D_r^*$

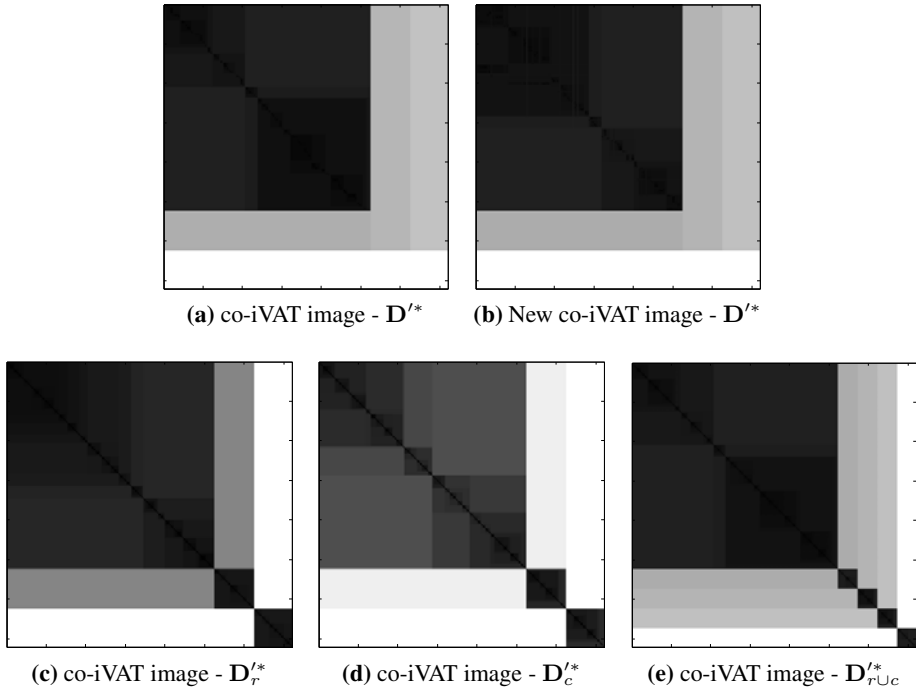


(f) co-VAT image -  $D_c^*$



(g) co-VAT image -  $D_{rUc}^*$

**Fig. 1.** co-VAT images of 360 row objects and 360 column objects represented by rectangular dissimilarity data  $D$



**Fig. 2.** co-iVAT images of 360 row objects and 360 column objects represented by rectangular dissimilarity data  $D$

## 4 Conclusion

This paper presented a new implementation of the co-VAT algorithm with two innovations; a new reordering scheme was presented in Algorithm 4 and each co-VAT algorithm was adapted to include the distance transform in (11). Although the numerical example shown in Fig. 1 does not clearly show the strength of the alternate reordering scheme presented in Algorithm 4, we are currently preparing a paper that will present several examples, many of which show that the alternate reordering scheme performs well in cases where the original co-VAT fails. Moreover, we emphasize that the alternate reordering scheme is computationally less expensive as  $D_{r \cup c}$  does not need to be constructed or VAT-reordered.

Figure 2 shows that the co-iVAT images are clearly superior to the original co-VAT images in showing the cluster tendency of the four types of clusters in the rectangular data. Additionally, due to our recursive implementation in Algorithm 2, these improved images come at very little additional computational cost.

We left many questions unanswered in this paper and are currently penning an article that describes, in detail, each of the following:

1. What are the advantages and disadvantages of the alternate reordering scheme presented in Section 2.1? We have discovered “pure” relational data (data for which

object data  $X$  does not exist) on which the original co-VAT formulation fails and on which the alternate reordering scheme is successful. Additionally, we have revisited the normalization of  $\mathbf{D}_r$  and  $\mathbf{D}_c$  in (3) and (4) and have devised different ways to normalize these matrices which show improvement in performance.

2. We have developed proofs of our assertions on iVAT presented in Section 1.1 and the recursive formulation of (1) outlined in Algorithm 2.
3. The path-based distance transform can be applied to the rectangular dissimilarity data directly. We are working on an algorithm, similar to Algorithm 2, that will compute the distance transform of the rectangular dissimilarity data, without having to exhaustively search through every path in the partially connected tree.
4. The sco-VAT [8] algorithm performs the operations of co-VAT for very-large (un-loadable) data. We are extending the co-VAT implementation presented here to the sco-VAT algorithm.
5. As always, we wish to demonstrate the performance of our algorithms on “real” data. However, gold-standard rectangular data-sets are not as ubiquitous as square relational data-sets. Hence, we are identifying data on which we can validate our work.

## References

1. Dhillon, I.: Co-clustering documents and words using bipartite spectral graph partitioning. In: Proc. 7th ACM SIGKDD Int. Conf. on Knowledge Discovery Data Mining, San Francisco, CA, pp. 269–274 (2001)
2. Theodoridis, S., Koutroumbas, K.: Pattern Recognition, 3rd edn. Academic Press, San Diego (2006)
3. Bezdek, J., Hathaway, R.: VAT: A tool for visual assessment of (cluster) tendency. In: Proc. IJCNN 2002, Honolulu, HI, pp. 2225–2230 (2002)
4. Bezdek, J., Hathaway, R., Huband, J.: Visual assessment of clustering tendency for rectangular dissimilarity matrices. IEEE Trans. Fuzzy Systems 15(5), 890–903 (2007)
5. Prim, R.: Shortest connection networks and some generalisations. Bell System Tech. J. 36, 1389–1401 (1957)
6. Wang, L., Nguyen, T., Bezdek, J., Leckie, C., Ramamohanarao, K.: iVAT and aVAT: enhanced visual analysis for cluster tendency assessment (2009) (in review)
7. Fisher, B., Zoller, T., Buhmann, J.: Path based pairwise data clustering with application to texture segmentation. In: Figueiredo, M., Zerubia, J., Jain, A.K. (eds.) EMMCVPR 2001. LNCS, vol. 2134, pp. 235–250. Springer, Heidelberg (2001)
8. Park, L., Bezdek, J., Leckie, C.: Visualization of clusters in very large rectangular dissimilarity data. In: Gupta, G.S., Mukhopadhyay, S. (eds.) Proc. 4th Int. Conf. Autonomous Robots and Agents, February 2009, pp. 251–256 (2009)

# User Behavior Prediction in Energy Consumption in Housing Using Bayesian Networks

Lamis Hawarah, Stéphane Ploix, and Mireille Jacomino

G-SCOP Laboratory, INP Grenoble, UJF, CNRS,  
46, avenue Felix Viallet - 38031 Grenoble, France  
{lamis.hawarah, stephane.Ploix, mireille.jacomino}@g-scop.inpg.fr  
<http://www.g-scop.inpg.fr/>

**Abstract.** This paper deals with the problem of the user behavior prediction in a home automation system. Anticipating the needed energy for a service is based on the available prediction (like user requests) which contains the uncertainties. When the future users requests are not available in a home automation system thanks to programmatic, it is interesting to predict it to anticipate the energy needed in order to avoid some problems like peak consumption. A general method to predict users requests for services in energy consumption is proposed. The method relies on Bayesian networks to predict and diagnose user's behavior in housing. Some results and perspectives are presented in this paper.

**Keywords:** energy consumption, prediction, Data Mining, Bayesian network.

## 1 Introduction

A home automation system basically consists of household appliances connected by an energy network and by a communication network allowing the interaction between appliances. Home and building automation is traditionally used to increase comfort, to enable remote access to buildings and to increase the efficiency of buildings. These systems may also aim at determining the best energy assignment plan and a good compromise between energy production and energy consumption [5], [6]. In this paper, energy is restricted to the electricity consumption and production.

Housing with the appliances aims at providing comfort to inhabitants thanks to services. The services can be decomposed into three kinds: the end-user services, the intermediate services and the support services which produce electrical power to intermediate and end-user services. Generally, when the home automation system is able to modify the behavior of a service, this service is qualified as *modifiable* by the system, for example, modification of the starting time of a cooking service or interruption of a washing service, etc. A service is qualified as *permanent* if its energetic consumption/production/storage covers the

whole time range of the energy assignment plan, otherwise, the service is named *temporary* service. In a home automation system, the user is not supposed to inform the system about his expectations (requested services). When the user's demand is not known during a given period, the system must take into account this uncertainty by anticipating the energy needed for services. This helps the system to avoid some problems like peak consumption in this period. Therefore, the behavior of the inhabitant has to be modeled and integrated into the home automation system.

In this paper, the user's behavior prediction problem in housing is only dealing with. A general method is proposed to predict the possible inhabitant service requests for each hour in energy consumption of a 24 hours anticipative time period. The idea is based on the use of the Bayesian Network (BN) to predict the user's behavior. Bayesian Networks (BNs) [2] are a field of Machine Learning, capable to represent and manipulate arbitrary probability distributions over arbitrary random variables. They are especially well suited for modeling uncertain knowledge in expert systems [3]. In this paper, first, related works concerning the problem of the energy consumption prediction are presented. The next section shows how a BN is learned and used. The approach to predict the user behavior in housing is explained. It is based on a real database concerning 100 houses in France. Finally, some results and perspectives are discussed.

## 1.1 Related Works

Various studies have been done in the field of impact of the user's behavior on the total amount of energy consumption in the households. [9], [10] study the interaction between the user and the appliances. The appliances are grouped into four categories of complexity according to their level of automation and number of settings. For example, the level of automation of the *Iron* is low and the number of settings is high. Therefore, the user must need to be in the proximity of the appliance and be available to monitor the end-uses. They achieved up to 10~20% reduction in energy consumption of households by changing the user behavior. Other studies are interested in modeling and simulation of user activity in control systems [8]. They integrated the behaviors of individuals and user groups into building performance simulators to get more realistic results. This approach models all users and user groups as individual agents with different behaviors. Different roles and function units such as work places are also modeled. The main results of this work is that user activities of individuals and groups in office environments can be modeled on the basis of communicating agents. [7] studies and analyzes generally the user behavior in home environment.

## 1.2 Bayesian Networks

A Bayesian network is a graphical model for probabilistic relationships among a set of variables [4]. BNs model causal relationships. They are represented as directed acyclic graphs, where each node represents a different random variable. A directed edge from the node X (*cause node*) to the node Y (*effect node*) indicates

that  $X$  has a direct influence on  $Y$ . This influence is quantified by the conditional Probability  $P(Y|X)$ , stored at node  $Y$ . A conditional probability Table (CPT) is assigned to each node in the network. Such probabilities may be set by an expert or using a historical database. The nodes in a network can be divided in two types: *evidence node* when its value is observed, and *query node* when its value is to predict. BNs are based on the conditional independence; each node is conditionally independent of its non-descendants given its parents. When a node has no parent, its CPT specifies the prior probability. There are two types of learning: 1) *the structure learning* in which the best graph representing the problem is researched; 2) *the parametric learning* in which the network structure is known but the conditional probability will be estimated at each node. Once the Bayesian Network is constructed, it can be used to compute the probability distribution for a **query** variable, given a set of **evidence** variables. This operation is called *inference*. For example, identify the causes by calculating the most probable cause given some information. Or, predict the effects by calculating the most frequent value of a node given some observations.

## 2 Problem Statement

To anticipate the energy needed for a service in a home automation system, the system must take into account the uncertainty which can be provided by the user. The user may not inform the system about his energetic plan during a day or may completely cancel the service which he wanted to use. The objective of this work is to statistically predict the user energetic service requests at each hour using a Bayesian network. The nodes are determined with their values and the relationships between them. However, the Conditional Probability Distribution at each node is computed using an actual database concerning the energy consumption in housing.

### 2.1 Databases

A database is obtained from *Residential Monitoring to Decrease Energy Use and Carbon Emissions in Europe (REMODECE)*<sup>1</sup> which is a European database on residential consumption, including Central and Eastern European Countries, as well as new European Countries (Bulgaria and Romania). This database stores the characterization of residential electricity consumption by end-user and by country. The *IRISE* project has been chosen from *REMODECE* which deals only with houses in France. Each database concerns one house; in such a database, information is recorded every 10 minutes for each appliance in the house and over one year. This information represents the consumed energy by each service, its date and its time. An example of this data is given in the figure [12](#). Moreover, it is possible to know the number of people who live in each house. However, this data is not directly available. Let us notice that appliances are just involved in

<sup>1</sup> <http://www.isr.uc.pt/~remodece/>

<sup>2</sup> In this figure, the Date column represents the date and the time.

| Date                    | Electric-ø | Halogen-ø | TV-(55cm) | Clothes-ø | Fridge-freezer-(K1tche) |
|-------------------------|------------|-----------|-----------|-----------|-------------------------|
| "1999-04-01 00:00:00.0" | "0"        | "0"       | "0"       | "0"       | "14"                    |
| "1999-04-01 00:10:00.0" | "0"        | "0"       | "0"       | "0"       | "6"                     |
| "1999-04-01 00:20:00.0" | "0"        | "0"       | "0"       | "0"       | "0"                     |
| "1999-04-01 00:30:00.0" | "0"        | "0"       | "0"       | "0"       | "11"                    |
| "1999-04-01 00:40:00.0" | "0"        | "0"       | "0"       | "0"       | "9"                     |
| "1999-04-01 00:50:00.0" | "0"        | "0"       | "0"       | "0"       | "0"                     |
| "1999-04-01 01:00:00.0" | "0"        | "0"       | "0"       | "0"       | "7"                     |
| "1999-04-01 01:10:00.0" | "0"        | "0"       | "0"       | "0"       | "14"                    |
| "1999-04-01 01:20:00.0" | "0"        | "0"       | "0"       | "0"       | "0"                     |
| "1999-04-01 01:30:00.0" | "0"        | "0"       | "0"       | "0"       | "2"                     |
| "1999-04-01 01:40:00.0" | "0"        | "0"       | "0"       | "0"       | "19"                    |
| "1999-04-01 01:50:00.0" | "0"        | "0"       | "0"       | "0"       | "0"                     |
| "1999-04-01 02:00:00.0" | "0"        | "0"       | "0"       | "0"       | "0"                     |
| "1999-04-01 02:10:00.0" | "0"        | "0"       | "0"       | "0"       | "20"                    |
| "1999-04-01 02:20:00.0" | "0"        | "0"       | "0"       | "0"       | "1"                     |
| "1999-04-01 02:30:00.0" | "0"        | "0"       | "0"       | "0"       | "0"                     |
| "1999-04-01 02:40:00.0" | "0"        | "0"       | "0"       | "0"       | "20"                    |

Fig. 1. A part of a database representing a house in France

services: they are not central from the inhabitant point of view. Consequently, they are not explicitly modeled. The presence of the user is important but it is not predictable at the moment.

### 3 Using Data to Learn Parameters of the BN

Before processing the data, we must model the problem by characterizing the user demand. From the data given in the figure 1, a user request concerns one or more services like *cooking in oven*, *clothe washing*, *water heating*, etc.

#### 3.1 Modeling Data

To anticipate a service in a home automation system, it is interesting to predict:

- *When is the service requested?* that means the starting hour, month and week-day;
- *How much energy does the service consume?*
- *What is the duration of the service?*

This information characterizes the service request itself and is available in the database except for the services duration. The information about the inhabitants characteristics like age, profession and presence in housing are not given.

#### 3.2 Behavior Profile

The aim is to find a profile of the user’s need in energy using only the information given in the database in order to improve the prediction. Thus, the timestamp given in the column Date will be dealt with to obtain the month, weekday and hour separately. A day may be Saturday, Sunday, Monday, Tuesday, Wednesday, Thursday and Friday. Thus, the first step of this work distinguishes only between Saturday, Sunday and weekday (WE). The idea is to show if the behavior is different between the week-end and a weekday without distinguishing between weekdays. Only the useful information concerning an appliance is extracted from the database: hourly interval in which the service has been started, duration,



| Date       | Hour    | Duration | Energy | WE         | Month   | Starting number |
|------------|---------|----------|--------|------------|---------|-----------------|
| 1999-04-01 | [19-20[ | 20       | 30     | "no"       | "April" | 1               |
| 1999-04-01 | [19-20[ | 50       | 687    | "no"       | "April" | 2               |
| 1999-04-01 | [21-22[ | 20       | 8      | "no"       | "April" | 1               |
| 1999-04-07 | [12-13[ | 70       | 1257   | "no"       | "April" | 1               |
| 1999-04-07 | [19-20[ | 50       | 971    | "no"       | "April" | 1               |
| 1999-04-08 | [12-13[ | 30       | 814    | "no"       | "April" | 1               |
| 1999-04-08 | [13-14[ | 10       | 2      | "no"       | "April" | 1               |
| 1999-04-08 | [19-20[ | 60       | 1026   | "no"       | "April" | 1               |
| 1999-04-09 | [20-21[ | 50       | 1246   | "no"       | "April" | 1               |
| 1999-04-09 | [21-22[ | 10       | 2      | "no"       | "April" | 1               |
| 1999-04-10 | [12-13[ | 80       | 1499   | "Saturday" | "April" | 1               |
| 1999-04-10 | [19-20[ | 50       | 836    | "Saturday" | "April" | 1               |
| 1999-04-11 | [13-14[ | 20       | 466    | "Sunday"   | "April" | 1               |
| 1999-04-13 | [19-20[ | 100      | 1425   | "no"       | "April" | 1               |
| 1999-04-14 | [12-13[ | 110      | 1541   | "no"       | "April" | 1               |
| 1999-04-14 | [19-20[ | 130      | 1541   | "no"       | "April" | 1               |
| 1999-04-14 | [21-22[ | 10       | 1      | "no"       | "April" | 1               |

Fig. 2. The database after treatment

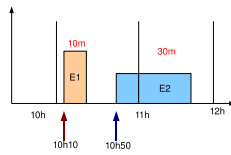


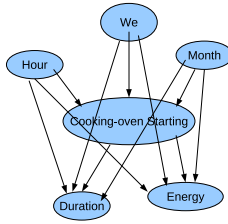
Fig. 3. Two services starting in the same hour

energy, month, WE, starting number<sup>3</sup>. The figure 3 shows that the service started two times in the same hourly interval. It consumed  $E_1$  and went on 10 minutes for the first starting. For the second, it consumed  $E_2$  and went on 30 minutes. Thus, another database is obtained (figure 2) which represents only one appliance. Each appliance is modeled into a database.

### 3.3 Learning the Structure of the BN

The structure of the network is built without using any learning algorithm. To use a learning algorithm, a database is needed. The available database given in the figure 1 can not be use for this purpose, because the derived Bayesian network could only express the relationship between services that is not the interesting information. The total duration and the total energy for each starting time of every service is not computed in this database. Moreover, the database given in the figure 2 is not complete to calculate the independence between the nodes, because it contains only the useful information about one appliance. A Bayesian Network for a *cooking-oven* service is characterized by the following nodes: *Hour* node takes 24 values from 0 to 23 which represents the interval in which the service starts and not the exact time. *Month* node takes 12 values from *January* to *December* because each house in the database is studied during one year. *WE* node takes 3 values {*Saturday*, *Sunday*, *weekday*}. *Service-starting* node represents any energetic service and takes two values {*yes*, *no*}. This node is added because a service may be started 3 times in the same hourly interval. The

<sup>3</sup> Because the information is given every 10 minutes, an appliance may be started many times at most 3 times in the same hour interval (figure 3).



**Fig. 4.** Bayesian Network for cooking-oven service

network is built by taking into account only the first starting. Otherwise, it is possible to deal with the three starts by replacing the values  $\{ yes, no\}$  of the node *service-starting* by the values  $\{first-starting, second-starting, third stating, no\}$ . Both *Duration* node and *Energy* node are deterministic. Each node has its value specified exactly by the values of its parents, with no uncertainty [2]. In this network, the causal nodes are  $\{hour, WE, month\}$  because they influence the consumption. *Service-starting* is the direct effect of the causal nodes; it is also a causal node for the *duration* and the *energy* because the energy and the duration of a service is obtained after its starting. There are effectively a dependence between the *energy* and the time setting  $\{hour, WE, month\}$  because the time setting influences the consumed energy and the *service starting* is not enough only to predict the energy. If the *starting hour* is not connected with the *energy*, changing the *starting hour* does not have any influence on the consumed energy and its probability. In reality, this is not true. However, if the *starting hour* is connected with the *energy*, the consumed energy and its probability change according to the *starting hour*. For the same reason, the *energy* and the *duration* are connected with the *WE* and *month*. The Bayesian network for a *cooking-oven* service is given in the figure 4. The Conditional probability Distribution at each node is calculated from a processed database as given in the figure 2. A statistical estimation is used to calculate the frequency of the node in the database (equation 1).

$$P(X_i = x_k | pa(X_i) = x_j) = \frac{N_{i,j,k}}{\sum_k N_{i,j,k}} \tag{1}$$

Where  $N_{i,j,k}$  is the number of events in the database for which the variable  $X_i$  takes the value  $x_k$  and its parents take the values  $x_j$ .

### 3.4 Application to the Example of Housings

This section presents the probabilities of starting on some houses from *IRISE* database. In this database, there are 27 houses using *Electric-oven*, the *Cooking* is the service chosen to illustrate the method. The result on the house 2000997, in which the people number is 5, is presented. The targeted appliances are *Electric-oven* and *Microwave-oven*.

The figure 5 shows the probability that an appliance starts at each hour on weekdays, Saturday or Sunday (from the top to the bottom) over all the months.

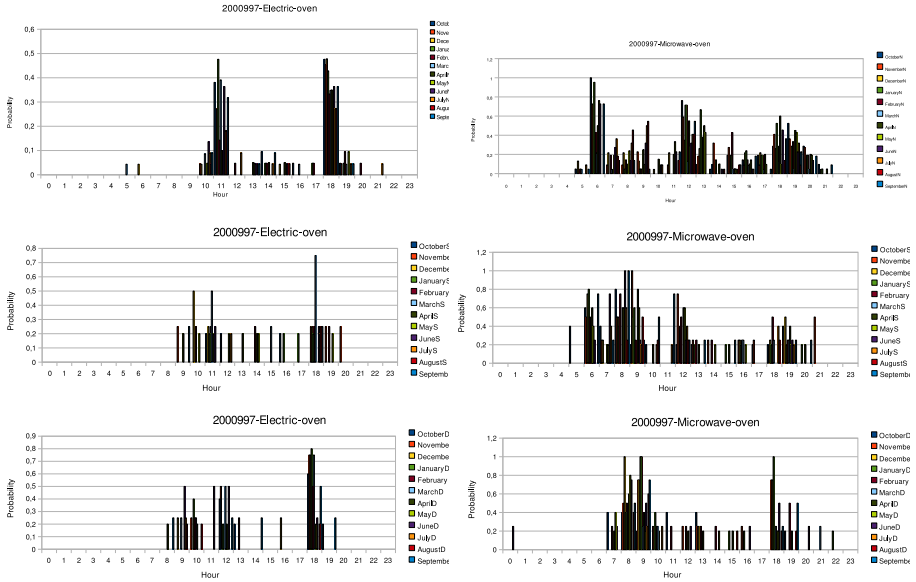


Fig. 5. Probability of service starting in the house 2000997

At the left, the appliance is an *Electric-oven*; at the right, it is a *Microwave-oven*. The user behavior in this household on weekdays is not the same on Saturday or on Sunday. On weekdays, there are two interesting uses: 1) between 11am and 12am; 2) between 6pm and 7pm. The most frequent value of probability is 0.5. However, the use of the *Electric-oven* is less frequent on Saturday than on weekdays but it is more frequent on Sunday evening for some months. On the other hand, the use of the *Microwave-oven* is more frequent than the *Electric-oven*. The probability reaches the value 1 for some months and at some hours. For example, the probability that the *Microwave* starts on weekdays between 6am and 7am is 1. This probability becomes 0.8 between 8 am and 9 am on Saturday. Other houses are tested on these appliances, but the results are not given in this paper. Therefore, the user behavior is not identical in all houses. The Bayesian network in this paper represents only one service in one house. It is possible to add to this network all the energetic services used in the housing.

## 4 Conclusion and Perspectives

This paper focuses on the prediction of user behavior in housing and in energy consumption, because it is a very important problem in a home automation system. The objective is to construct a model able to predict the user behavior in housing. The aim is to compute at each hour the probability of starting of each energetic service in housing. These probabilities are calculated using databases which consists of the energy consumed by the services in several houses

in France. These probabilities are introduced in a Bayesian Network to be used easy by a home automation system. This help the system to organize energy production and consumption and to decide which appliance will be used at each hour (energy planing). In the future work, a Learning system will be built. This system contains a set of profiles for each service and is able to choose the appropriate profile for the user given a service.

## References

1. Abras, S., Ploix, S., Pesty, S., Jacomino, M.: A Multi-Agent Design for a Home Automation System dedicated to power management. In: Proceedings of the IFIP Conference on Artificial Intelligence Applications and Innovations, Athen, Greece, September 19-21 (2007)
2. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach. Pearson Education, London (2003)
3. Heckerman, D.: A Tutorial on Learning Bayesian Networks. Communications of the ACM (1995)
4. Pearl, J.: Fusion, propagation, and structuring in belief networks. *Artif. Intell.* 29(3), 241–288 (1986)
5. Palensky, P., Dietrich, D., Posta, R., Reiter, H.: Demand Side Management in private homes by using LonWorks. In: Vortrag: WFCS97 2nd IEEE Workshop on Factory Communication Systems, Barcelona, pp. 341–347 (1997)
6. Ha, L.D., Ploix, S., Zamaï, E., Jacomino, M.: A home automation system to improve the household energy control. In: INCOM 2006 12th IFAC Symposium of Information Control Problems in Manufacturing, Saint Etienne, France (2006)
7. Ha, S., Jung, H., Oh, Y.: Method to analyze user behavior in home environment. *Personal Ubiquitous Comput.* 10(2-3), 110–121 (2006)
8. Zimmerman, G.: Modeling and simulation of individual user behavior for building performance predictions. In: SCSC: Proceedings of the 2007 summer computer simulation conference, San Diego, CA, USA, pp. 913–920 (2007)
9. Wood, G., Newborough, M.: Dynamic energy-consumption indicators for domestic appliances: environment, behaviour and design. *Energy and Buildings*, 821–841 (September 2003)
10. Wood, G., Newborough, M.: Influencing user behaviour with energy information display systems for intelligent homes. *International journal of energy research* 31(1), 56–78 (2007)

# Increasing Efficiency of Data Mining Systems by Machine Unification and Double Machine Cache

Norbert Jankowski and Krzysztof Grąbczewski

Department of Informatics  
Nicolaus Copernicus University  
Toruń, Poland  
{norbert,kg}@is.umk.pl  
<http://www.is.umk.pl/>

**Abstract.** In advanced meta-learning algorithms and in general data mining systems, we need to search through huge spaces of machine learning algorithms. Meta-learning and other complex data mining approaches need to train and test thousands of learning machines while searching for the best solution (model), which often is quite complex. To facilitate working with projects of any scale, we propose intelligent mechanism of machine unification and cooperating mechanism of machine cache. Data mining system equipped with the mechanisms can deal with projects many times bigger than systems devoid of machine unification and cache. Presented solutions also reduce computational time needed for learning and save memory.

## 1 Introduction

All data mining systems are always limited by the amount of memory of the computer system used to solve given task and by the time assigned to solve the task. The limits will always exist. We can never spend unlimited time or use unlimited memory resources. Today, none of commonly known data mining systems like Weka [1], Rapid Miner [2], Knime [3], SPSS Clementine [4], GhostMiner [5] (see [6] for more) care so much about that limits. As a result, when given learning machine is needed again, it is constructed again. It is not a rare case that given machine is used several times. For example consider testing several configurations of classifier committees which share some member machines. Comparably frequent example is that a data transformation precedes several classifiers. In consequence, time is being lost for repeated processes, while machines sharing configurations should be built just once.

Another problem is observed when *learning from data* of large size. Memory quickly gets full, when checking a number of learning machine configurations (manually or by meta-learning). Of course, there are methods of dealing with huge data, but we want to point this problem from a little different point of view. It is possible to use advanced machine learning techniques on huge data using complex configurations of machines, if only the data mining system being used, is supported by specialized mechanism of intelligent machine caching.

This is why we investigate the ideas of machine unification and machine cache in the following sections. All described elements have been implemented in Intemi—a general data mining system [7].

## 2 Machine Unification and Machine Cache

In general, the idea of unification lies in determination whether given machine (with given configuration and specification of input sources) has already been constructed (as a result of earlier request) and is available for reuse. It means that unification can reduce CPU consumption by not running any machine process twice. Up to now, no data mining system has proposed such feature, although it may significantly save computational power. Additionally, when it is not necessary to compute machine again, we do not use additional memory resources either, which would take place in the case of relearning of given machine.

To reuse already created machines, the machine cache is constructed. In fact, we propose two cooperating caches. One works in memory and cooperates with the other (a disk cache), which is used to extend the capacity of memory. But in contrary to swap mechanism nested in operating systems, it is not oriented in page swapping but in machine swapping, which is much more effective.

**Machine, Configuration and Learning Task.** Before we move into deeper detail about possibility of defining machine unification and construction of efficient machine cache, we introduce a general, formal view of learning machine.

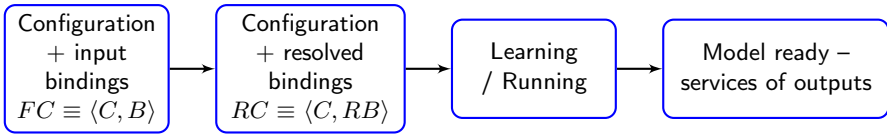
By *learning machine* we mean any (implemented) instance of learning algorithm. The term *model* means the result of learning. For example, in consequence of running a classifier machine we obtain a ready to use classification model. It is more advantageous to extend the term of learning machine to more general term of *machine*. Machine need not be a *learning* machine. In this way, the same entity may embrace, for example, algorithms to standardize data, to provide cross-validation tests or even to load or import data. The common feature of all machines is that they have *inputs*, their inner configuration (states of free parameters) and that they provide *outputs*. Inputs and outputs reflect machine roles, the goal of machine: a classifier output plays the role of a classifier, data standardization routine play the role of data transformer etc.

Each *machine configuration* may be defined by

$$C := \langle p, io, \{C_i : i = 1, \dots, n\} \rangle \quad (1)$$

where  $p$  represents machine process parameters,  $io$  specifies inputs and outputs (counts, names and types), and  $\{C_i : i = 1, \dots, n\}$  is a set of optional subconfigurations (enables construction of complex machines—each machine may have submachines).

The output of given machine may be *direct* (provided directly by given machine) or *indirect* (provided as a submachine output—it is defined by submachine path and output name). This is very important that outputs may be defined in



**Fig. 1.** Machine construction time line and learning task

so flexible way. Thanks to this, future complex machines do not need to reimplement each output but the output of a submachine may be simply exhibited.

Before given machine may be created, it must be completely defined by pair of machine configuration and its *input bindings*:

$$FC := \langle C, B \rangle. \tag{2}$$

The input bindings define symbolic input connections (input connections point symbolically appropriate source of input i.e. an output of another machine). Because machine may have several inputs, the input bindings  $B$  have a form of a set of pairs:

$$B := \{ \langle \textit{input name}, \textit{binding} \rangle \}, \tag{3}$$

which assigns a binding to each input.

The inputs compose acyclic directed graph (machines are vertices and input–output connections are edges). On the other hand we may see complex machines as machine trees, because each machine may have submachines (the tree subnodes).

Input binding may have one of three types:

$$\begin{aligned} \textit{binding} := & \langle \textit{parent input name} \rangle \mid \tag{4} \\ & \langle \textit{id of sibling machine}, \textit{sibling output name} \rangle \mid \\ & \langle \textit{submachine path}, \textit{output name} \rangle \end{aligned}$$

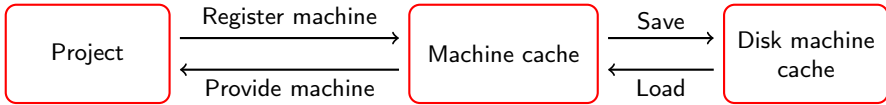
It may be a connection to a parent input, to a sibling machine output or to an output of a machine defined by a submachine path (which may point a child, a grand-child, etc.). Such three types of bindings facilitate sufficient and flexible definition of input–output connections (even for extremely complex machines).

Machine construction timeline is sketched in Figure 1. When a fully defined machine (defined by configuration and input bindings) is requested, the request waits until all inputs become ready to use (the learning processes of machines which provide those inputs are finished). When all the inputs are ready, the input bindings are transformed to *resolved inputs*:

$$RB := \langle \textit{input name}, \textit{rbinding} \rangle \tag{5}$$

i.e. a collection of resolved inputs of the form:

$$\textit{rbinding} := \langle \textit{machine stamp}, \textit{output name}, \textit{output stamp} \rangle, \tag{6}$$



**Fig. 2.** Cooperation between project, machine cache and disk cache

Resolved input provides machine stamp<sup>1</sup> (identifying the machine that really provides the required output) and the name of output of that machine. Note that some outputs are indirect and resolved binding provides link to the (final) machine which really provides the output.

The input resolving process, converts the pair  $FC$  to:

$$RC := \langle C, RB \rangle. \quad (7)$$

The pair  $RC$  provides all information about configuration of the requested machine, necessary to run the machine process, so the request is either submitted to the task spooler (when it is the first request for the machine) or directly filled with proper machine reference (when such machine has already been constructed and may be reused). Below we discuss the latter case in more detail.

**Machine Caching and Unification.** The goal of machine cache is to save computational time and reuse previously constructed and trained machines. When a machine is requested and the cache can not return it instantly, because the machine has not been constructed and trained yet, the request becomes a task to be accomplished by proper task running server. Each machine, just after learning, is *registered* in the cache and each next request for the same machine will be instantly served by the machine cache. Please compare Figure 2.

The *provide machine* functionality is not trivial, because each machine type may be configured in different ways and get different inputs. Moreover, machines are often complex, so their configurations include subconfigurations (of different machines). Another important aspect of machine unification is that it can not be based just on machine configuration and input bindings  $FC$  as in Eq. 2, because such  $FC$  is not yet definitively determined. This means that because of dynamical symbolic definition of bindings and especially because of opportunity of defining indirect outputs for machines, the definite and unambiguous is the pair  $RC$  from Eq. 7. The resolved input bindings point (completely and directly) the machines that provide required outputs, and name the outputs. The  $FC$  may be converted to  $RC$  when all connected machines are ready to use. In consequence, machine requests have to wait until all inputs can be resolved (the inputs are ready to use). After conversion from  $FC$  to  $RC$  the project may *ask* machine cache to provide machine basing on the pair  $RC$ .

Another two important problems occur, when the machine cache is expected to keep thousands of machines (not just tens):

<sup>1</sup> The necessity of machine stamps will be clarified later.



- Memory for saving machines is limited. In real cases the number of machines which can be kept in memory is relatively small. This is solved by disk cache cooperating with the memory cache.
- Searching for machines in machine cache must be highly effective. This is achieved by efficient machine unification based on the  $RC$  pairs.

The disk cache as a general concept is not described in this paper, because of space limit. For the purpose of this article it is enough to know that disk cache provides functionality of machine saving and machine loading as it was depicted in Figure 2. The functions are used on demand via machine cache functions as can be seen below—see functions `RegisterMachine` and `ProvideMachine`.

The problem of machine unification can not be realized by means of plain comparison of two  $RC$  configurations. It would be too slow to compare searched  $RC$  pair with each  $RC$  pair in the cache. The search complexity depends linearly on the number of machines in machine cache, but when the  $RC$  pairs are complex, complexity is equal to the sum over all parameters of all  $RC$ 's in cache:

$$\sum_{rc \in cache} |rc|, \quad (8)$$

where  $|rc|$  is the length of  $rc$ .

To make machine search much quicker than the naive solution, we have built specialized machine cache using three hash dictionaries for three types of mappings:

- unificator dictionary, mapping from  $RC$  pair to unique machine stamp. It means that the machine cache may provide appropriate machine only if the unificator dictionary contains appropriate  $RC$  key.
- unificatorRev dictionary, providing mapping inverse to unificator (from machine stamps to  $RC$  pairs).
- cache, mapping machine stamps to machines. It cooperates with the disk cache: before a machine is released from memory, it is first saved in the disk cache. Thanks to this, a single machine may be shared in many places (for example in several complex machines).

The three hash dictionaries obviously need fast calculation of hash codes, but as a result, they guarantee access in approximated complexity  $O(|rc|)$  and independence from the number of machines in machine cache (very important for scalability of data mining systems).

**Machine Cache Functionality.** As mentioned above, each machine, just after it is ready, is registered in the machine cache. The registration is sketched in Figure 3.

In the first line, outputs of machines are registered, to simplify further unification and to accelerate the process of transformation of input bindings into resolved bindings. Basing on such construction, translation to resolved binding may be done just once for each output, even if the output is used many times.

```

1 function RegisterMachine(machine); 13
2 RegisterOutputs(machine.Outputs); 14
3 s = GetNextUniqStamp(); 15
4 RC_pair = machine.GetRC(); 16
5 unificator[RC_pair] = s; 17
6 unificatorRev[s] = RC_pair; 18
7 cache[s] = machine; 19
8 diskCache.PushSaveRequest(s, machine); 20
9 end 21
10 22
11 function ProvideMachine(RC_pair); 23
12 s = unificator[RC_pair]; 24 end
 if (s != 0) {
 machine = cache[s];
 if (machine == 0)
 machine = cache[s] =
 diskCache.Load(s);
 } else {
 machine =
 taskSpooler(RC_pair);
 RegisterMachine(machine);
 }
 return machine;

```

**Fig. 3.** Listings of RegisterMachine and ProvideMachine

In line 3, new unique stamp is assigned to the newly prepared machine. Next, the resolved configuration  $RC$  and the stamp are used to define two mappings: from  $RC$  to stamp (in `unificator` dictionary) and vice versa (`unificatorRev` dictionary). After that, the final mapping from the machine stamp to machine is added to the cache dictionary. The last line of `RegisterMachine` pushes the request for asynchronous save of the machine to disk cache. Note that complexity of this procedure is  $O(|rc|)$  except the subcall of procedure `RegisterOutputs` which complexity depends linearly on the number of machine outputs (usually a very small number).

The machine cache search for requested machine, basing on requested  $RC$  pair of configuration and resolved input bindings, is realized by `ProvideMachine` function—see Figure 3. First, the `unificator` dictionary is checked whether it already contains requested  $RC$  pair. If it does, then the machine is contained within the machine cache or at least within disk cache (if it has been released from memory). In such case the machine is extracted from cache (code line 14) or loaded back to memory cache (code line 17).

If machine has not been deposited in machine cache yet, then the task spooler is used to construct and learn machine basing on  $RC$  pair (see code line 20). After that, in the next line, the machine is registered in machine cache.

It is very important that each machine is provided in the way presented above, so that each two equivalent machines may be unified, even if one (or both) of them is a submachine. From the cache point of view each machine (regardless of whether they are submachines at any level) are handled in the same way. They are identified just by their stamps.

Another advantage of machine cache is that even if a machine is removed from the project or it is no longer a submachine, it is not necessary to remove it from machine cache. Next requests for this machine will find it ready for reuse.

*Releasing machines.* Machine cache periodically observes (with `TryRelease` function) the usage of each machine and depending on the usage statistics, it decides whether given machine can be released or not. Each time a machine is

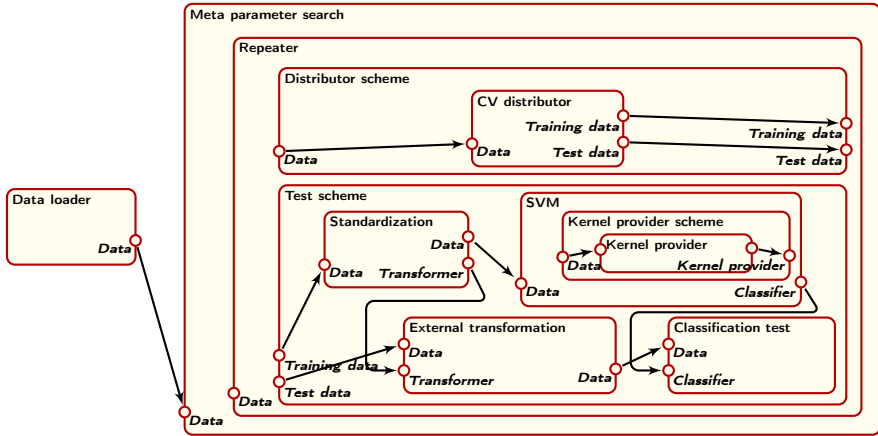


Fig. 4. A meta parameter search project configuration

constructed or becomes not used, its lastUseTime is updated and it is added to special list analyzed within TryRelease (see the code below). Of course, machine can not be released before it is saved in the disk cache.

```

25 function TryRelease(machineList)
26 foreach m in machineList
27 if (now - m.lastUseTime > timelimit && m.isSavedInCache) {
28 cache[m.stamp] = null;
29 machineList.Remove(m);
30 }
31 end

```

### 3 Unification Example

As mentioned in section 2, machine unification is especially advantageous in meta-learning. Even one of the simplest meta-learning approaches, a simple meta parameter search (MPS), is a good example. Imagine a project configuration depicted in Figure 4, where the MPS machine is designed to repeat 5 times 2-fold CV of an attractiveness test for different values of SVM C and kernel  $\sigma$  parameters. MPS machines are hierarchical and use different submachines to examine specified test tasks. To test the C parameter within the set  $\{2^{-12}, 2^{-10}, \dots, 2^2\}$  and  $\sigma$  within  $\{2^{-1}, 2^1, \dots, 2^{11}\}$ , we need to perform the whole  $5 \times 2$  CV process  $8 \times 7$  times.

As enumerated in Table 1, such a project contains (logically) 4538 machines. Thanks to the unification system, only 1928 different machines are created, saving both time and memory. The savings are possible, because we perform exactly the same CV many times, so the data sets can be shared and also the SVM machine is built many times with different C parameters and the same kernel  $\sigma$ , which facilitates sharing the kernel tables by quite large number of SVM machines.

**Table 1.** Numbers of machines that exist in the project logically and physically

| Machine                 | logical count | physical count |
|-------------------------|---------------|----------------|
| Data loader             | 1             | 1              |
| Meta parameter search   | 1             | 1              |
| Repeater                | 56            | 56             |
| Distributor scheme      | 280           | 5              |
| CV distributor          | 280           | 5              |
| Test scheme             | 560           | 560            |
| Standardization         | 560           | 10             |
| External transformation | 560           | 10             |
| SVM                     | 560           | 560            |
| Kernel provider scheme  | 560           | 80             |
| Kernel provider         | 560           | 80             |
| Classification test     | 560           | 560            |
| Sum                     | 4538          | 1928           |

## 4 Summary

The concepts of machine unification and machine cache, presented above, are efficient and sufficiently general to be usable for many purposes in the scope of data mining. Discussed advantages clearly show that a machine realizing given configuration should be run once and next should be reused wherever requested, instead of running it many times, as it has been done in the past. Very attractive complexity of presented machine unification and disk cache management, saves memory and CPU time and facilitates much more sophisticated data analysis including advanced meta-learning.

## References

1. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, San Francisco (2005)
2. Rapid miner: Rapid miner 4.4 (2009), <http://rapid-i.com>
3. Knime: Knime konstanz information miner (2009), <http://www.knime.org/>
4. SPSS: Clementine—pasw modeler (2009)
5. Jankowski, N., Grąbczewski, K., Duch, W.: GhostMiner 3.0. FQS Poland, Fujitsu, Kraków, Poland (2004)
6. KDnuggets: Software suites for Data Mining and Knowledge Discovery (2009), <http://www.kdnuggets.com/software/suites.html>
7. Grąbczewski, K., Jankowski, N.: Meta-learning architecture for knowledge representation and management in computational intelligence. International Journal of Information Technology and Intelligent Computing 2(2), 27 (2007)

# Infosel++: Information Based Feature Selection C++ Library

Adam Kachel<sup>1</sup>, Jacek Biesiada<sup>1,3</sup>, Marcin Blachnik<sup>1</sup>, and Włodzisław Duch<sup>2</sup>

<sup>1</sup> Silesian University of Technology, Electrotechnology Department,  
Katowice, Krasińskiego 8, Poland  
Jacek.Biesiada@polsl.pl

<sup>2</sup> Department of Informatics, Nicolaus Copernicus University,  
Grudziądzka 5, Toruń, Poland  
Google: W. Duch

<sup>3</sup> Division of Biomedical Informatics, Children's Hospital Research Foundation,  
3333 Burnet Ave., Cincinnati, Ohio 45229

**Abstract.** A large package of algorithms for feature ranking and selection has been developed. Infosel++, Information Based Feature Selection C++ Library, is a collection of classes and utilities based on probability estimation that can help developers of machine learning methods in rapid interfacing of feature selection algorithms, aid users in selecting an appropriate algorithm for a given task (embed feature selection in machine learning task), and aid researchers in developing new algorithms, especially hybrid algorithms for feature selection. A few examples of such possibilities are presented.

## 1 Introduction

Feature selection and ranking is an essential step in all data mining projects [1]. In bioinformatics, text analysis, object recognition or in modeling of complex technological processes large number of features is generated, and from a specific point of view, frequently related to recognition of some target concepts, only a small subset of features is usually relevant. Moreover, strategy based on systematic construction of many types of features followed by selection appears to be very powerful in finding simple models of data [2]. With limited amount of available data many spurious models in highly dimensional feature spaces may be created due to the accidental correlations between the target concept and various ways of partitioning the data, making these solutions worthless. To deal with such problems dimensionality of the feature space has to be reduced first. This may be done by ranking these features and selecting the most important ones, selecting a subset of relevant features or by combining (aggregating) subsets of features to create new, more informative features.

According to fundamental results in computational learning theory no single method is the best in all situations, and no single feature selection algorithm is the best for all data and all tasks. Many feature selection and feature ranking methods have been proposed in the literature [3,4,5]. Although numerous libraries of learning methods have been created (as detailed in the next section) libraries of feature selection algorithms

are not so popular. In this contribution we present InfoSel++, a library based on standard as well as novel algorithms. These algorithms may be used in several ways: for ranking, feature selection based on filters, or as a combination of filters and wrappers called frappers [6]. Ranking of features neglects their possible interactions, assigning relevancy to individual features and introducing partial order among features. Many measures of relevancy based on statistics and information theory suitable for ranking methods have been implemented. Wrappers use the results of predictors to evaluate the usefulness of features, but need a learning algorithm to control feature selection process. Wrapper methods employ statistical re-sampling techniques (such as cross-validation) using specific learning algorithms to estimate the accuracy of feature subsets. This approach has proved useful, but may be computationally very demanding because the learning algorithm is called repeatedly. Model selection techniques should be used to avoid over-fitting. For this reason wrappers do not scale well to large datasets containing many features. Filter methods, on the other hand, operate independently of any learning algorithm, searching for potentially useful dependencies between target task and distribution of feature values. Typically they attempt to rank features according to a relevancy score, but may also be used for selection of subsets of features [6].

In the next section various projects where feature selection has been prominent are reviewed. The third section describes our InfoSel library, section 4 contains a few results and comparisons, and the final section a brief discussion.

## 2 Related Work

Several large-scale efforts that implement libraries of machine learning algorithms have been undertaken in the past, and many of them include special modules for feature selection. Some are designed for general tasks, and some are specialized in such areas as microarray gene selection analysis [3]. Large projects involving feature selection as a part of a bigger system are listed first (MLC++, Weka, GhostMiner, Matlab and R packages, ToolDiag), followed by smaller and more specialized projects (RankGene, Feature Selection Toolbox).

1. **The Machine Learning in C++ library** (MLC++) [7] for supervised data mining problems has been developed at Stanford University. It includes decision trees and decision tables, Naive Bayes, instance based algorithms and many other machine learning techniques. It provides an implementation of wrapping approach for feature selection utilizing best first search, forward and backward selection methods.

2. **Weka** [8] is a popular large data mining environment developed at Waikato University, New Zealand, that is still being rapidly developed and used as a part of newer packages, such as RapidMiner [9]. Among many computational tools implemented, it contains about 15 attribute and subset evaluator methods, extended by 10 search algorithms for feature selection. Weka's feature selection algorithms are grouped into two subcategories: *feature evaluators* and *search methods*. The former group is used for evaluation of relevance of single features or feature subsets, usually by estimating various ranking coefficients, including: information gain, gain ratio, Chi squared statistic, oneR tree index, significance index, symmetrical uncertainty index. The latter category is an implementation of different optimal and sub-optimal search methods that use feature evaluators as a cost function.

3. **GhostMiner** is a commercial data mining tool distributed by Fujitsu. It has several classifiers, including various versions of SVM, decision tree, kNN with feature weighting and instance selection methods, incremental neural network, and a Feature Space Mapping neuro-fuzzy network. It also implements a few most effective feature selection methods like forward and backward selection, ranking based on various coefficients, and estimation of feature subset quality by wrapping with any classifier.

4. **PRTool and Spider** are Matlab toolboxes designed for data mining, neural networks and machine learning. **PRTool** has been developed by the Pattern Recognition Group at Delft University and is freely distributed. This toolbox implements various pattern recognition algorithms (kNN, decision trees, Parzen classifier, etc) that may be combined with feature selection methods (ranking of individual features, various search methods). **Spider** (Max Planck Institute of Biological Cybernetics, Germany) is delivered under the GNU license. It allows for an easy creation of two-staged data mining tasks. Available feature selection methods include mutual information filters, Fisher/Correlation score, greedy selection algorithm,  $L_0$  zero-norm minimization, primal zero-norm based feature selection, feature scaling using SVMs, non-linear feature elimination, and multi-class feature selection using spectral clustering. These methods may be combined with various classification and regression algorithms.

5. **R-project packages** (GNU license), similarly to Matlab toolboxes, are designed to solve dedicated computational problems. Feature selection using R is available through the *FSelector* package. It includes all feature selection algorithms implemented in Weka. Additional smaller packages are also available. *PenalizedSVM*, provides the smoothly clipped absolute deviation (SCAD) and  $L_1$ -norm penalty functions for SVM based feature selection. The *SDA* package (Shrinkage Discriminant Analysis and Feature Selection) offers a classifier that can be trained using Stein-type shrinkage estimators where features are ranked using correlation-adjusted  $t$ -scores. *Bioconductor* is an open source software project for the analysis of genomic data, gene selection and association analysis. Finally, the *predmixcor* package creates classification rules based on Bayesian mixture models with feature selection bias corrected approach.

6. **ToolDiag** is another general data mining and pattern recognition tool. It includes several decision making algorithms such as artificial neural networks (MLP, RBF, LVQ), kNN, linear, quadratic and Parzen classifiers. Also available are basic preprocessing and statistical analysis methods. Model estimation approaches include resubstitution error, holdout, cross-validation, bootstrap and leave-one-out. Tools for feature selection include 5 search algorithms (best features, sequential forward and sequential backward selection, branch and bound, exhaustive search) that can be combined with three groups of selection criteria: estimated minimal error probability, inter-class distance (using different distance matrices), and probabilistic distances (Chernoff, Bhattacharyya, Jeffreys-Matusita, Patrick-Fisher and Mahalanobis distance, KL divergence).

7. **Feature Selection Toolbox** [10] is an advanced tool developed by Petr Somol and his group. The software includes some classical and new methods of dimensionality reduction, classification and data representation. The main advantage of this package is a wide range of search methods implemented, including sub-optimal sequential search methods (Sequential Forward Search, SFS; Sequential Backward Search, SBS; Sequential Floating Forward Search, SFFS; Plus-L-Minus-R Search), generalized methods

(Adaptive-SFFS, several Oscillating Search versions, etc.), optimal search methods like Exhaustive Search, classical Brand and Bound (BB) and its extended versions, predictive BB search etc. This project is currently discontinued but the authors are working on a new open source C++ library (private communication).

Many specialized software tools also rely on feature selection, for example **RankGene** [11] designed to analyze gene expression data. For many more projects that include feature selection modules see the packages listed at the KDnuggets site [www.kdnuggets.com](http://www.kdnuggets.com).

### 3 Infosel++ for End-User

While Infosel++ enables easy development of new feature selection algorithms, most users are interested only in testing and comparing different already implemented algorithms. Most algorithms in this library are based on estimation of probability distributions, also several statistical algorithms (t-score, Correlation Coefficient, F-score) have been implemented for easy comparison. Different search method (ranking, forward, backward and exhaustive search) may be combined with various cost functions whenever it is appropriate.

#### 3.1 Filter Algorithms

Structure of generalized filter, wrapper and hybrid algorithms, as proposed in [12], has been implemented. For filters (Fig. 1), given a data set  $\mathcal{D}$  the search starts from a subset  $S$  (an empty set, a full set, or any randomly selected subset), exploring the space of combinations of features using particular search strategy. Each generated subset  $S$  is evaluated by some measure of relevancy  $M(S)$ , and if it is an improvement replaces the best subset found so far. The search iterates until a predefined stopping criterion based on information or correlation measures is reached. Changing the search strategies and evaluation measures used in steps 5 and 6 of the algorithm, new algorithms are created. The filter model applies evaluation criteria (distance, information, consistency or dependency measures) independent of predictive algorithms, therefore it avoids their biases and is computationally efficient in comparison with wrapper or hybrid algorithms.

#### 3.2 Implemented Algorithms

Feature selection algorithms implemented in the Infosel++ have been split into 4 distinct groups (many formulas are given in [13,6] and [14]). Acronyms in parenthesis are used in the Infosel++ menu system:

**1. Ranking methods**, including: CC (pcc), Pearson's Correlation Coefficient,  $t$ -score (tsc),  $t$ -score statistics (for two class problem) [15], F-score (fsc), F-score statistics (for multi-class problem) [16],  $\chi^2$  (chq),  $\chi^2$ -score statistics, MI (mi), Mutual Information, SUC (suc), Symmetrical Uncertainly Coefficient, distance rankings according to MDr (mdr) Matusita, KDr (kdr) Kolmogorov, KLDr (klldr) Kullback, BDr (bdr) Bhat-tacharatya, and SDr (sdr) Sammon index [14,6].



**Filter Algorithm**


---

**input:**  $\mathcal{D}(F_0, F_1, \dots, F_{N-1})$ ; a training data set with  $N$  features  
 $S$ ; a subset from which to Start the search (Starting subset)  
 $\delta$ ; a stopping criterion

**output:**  $F_{best}$ ; Final subset selected

```

01 begin
02 initialize: $F_{best} = S$;
03 $\gamma_{best} = eval(S, \mathcal{D}, M)$; evaluate S by an independent measure M
04 do begin
05 $S = generate(\mathcal{D})$; generate a subset for evaluation
06 $\gamma = eval(S, \mathcal{D}, M)$; evaluate the current subset S using M
07 if ($\gamma \geq \gamma_{best}$)
08 $\gamma_{best} = \gamma$;
09 $S_{best} = S$;
10 if (δ is true) end; check stopping criterion
11 return $F_{best} = S$;
12 end;
```

---

**Fig. 1.** Generalized filter algorithm

**2. Ranking with shifting of redundant features:** MIFS (mifs): Mutual Information Feature Selection [17], MIFS-U (mifsu): MIFS under Uniform Information Distribution [18], AMIFS (amifs): Adaptive MIFS [19], MID (mid): Mutual Information Difference and MIQ (miq) Quotient [16], FCD (fcd): F-test Correlation Difference and FCQ (fcq) Quotient [16].

**3. Ranking with removal of redundant features:** FCBF (fcfb): Fast Correlation Based Filter [20], K-S CBF (ks\_cbf): A Kolmogorov-Smirnov Correlation-Based Filter [21][22], K-SC CBF (ksc\_cbf): A Kolmogorov-Smirnov Class Correlation-Based Filter [23].

**4. Other methods** include Markov Blanket approximation (mbr) [24][25], and GD-distance ranking (gdd) [26].

These indices are based on dependency measures, information, distance and consistency measures. Ranking filters are the least expensive algorithms for feature ordering, but they cannot discover important interactions between features nor reject redundant features. Ranking with shifting of redundant features uses a heuristic based on a minimal-redundancy maximal-relevancy (MRMR) approach [17][16] to *shift* redundant features towards less important positions. Proposed heuristics and their improvements are presented in Tab. 1. Mutual Information  $MI(f_i, C)$ , is one of the most common measures of dependency,  $F(f_i, C)$  stands for F-score statistics and  $c(f_i, f_j)$  is a correlation coefficient.  $\beta$  is an arbitrary parameter in range  $[0, 1]$ .

The second group of ranking methods selects optimal non-redundant subsets of features removing all redundant features. Liu [20] has used “predominant features” for that purpose, similar approach has been proposed by Biesiada *et al.* [21][22], where two features are recognized as redundant if they have the same probability distributions or the

**Table 1.** Formulas used in Maximum Relevancy Minimum Redundancy (MRMR) ranking

| Type       | Acronym | Full Name                    | Formula                                                                    |
|------------|---------|------------------------------|----------------------------------------------------------------------------|
| Discrete   | MIFS    | Mutual info. feat. sel. [17] | $MI(f_i, C) - \beta \sum_{j \in S} MI(f_i, f_j)$                           |
|            | MIFS-U  | MIFS uniform distr. [18]     | $MI(f_i, C) - \beta \sum_{j \in S} \frac{MI(f_i, C)}{H(f_i)} MI(f_i, f_j)$ |
|            | AMIFS   | Adaptive MIFS [19]           | $MI(f_i, C) - \frac{1}{\ S\ } \sum_{j \in S} \frac{MI(f_i, f_j)}{H(f_j)}$  |
|            | MID     | Mutual info. difference [16] | $MI(f_i, C) - \frac{1}{\ S\ } \sum_{j \in S} MI(f_i, f_j)$                 |
| Continuous | MIQ     | Mutual info. quotient [16]   | $MI(f_i, C) / \frac{1}{\ S\ } \sum_{j \in S} MI(f_i, f_j)$                 |
|            | FCD     | F-test corr. difference [16] | $F(f_i, C) - \frac{1}{\ S\ } \sum_{j \in S}  c(f_i, f_j) $                 |
|            | FCQ     | F-test corr. quotient [16]   | $F(f_i, C) / \frac{1}{\ S\ } \sum_{j \in S}  c(f_i, f_j) $                 |

same joint distributions (feature and class) [23]. Various forward, backward and greedy search methods that use MI or SUC as evaluation functions have been implemented.

Modular construction of Infosel++ facilitates development of new methods with little coding. The technical details useful for developers will be published separately in a longer paper.

### 4 Illustrative Results on Synthetic Data

Results on 3 synthetic datasets are presented here to test our implementation. They are easy to understand and point to deficiencies of some methods. Extensive tests including novel combinations of ranking and selection methods will be published elsewhere. A very simple dataset used by Shridhar *et al.* [27] contains 12 patterns with 4 features (Tab. 2). All variables in this problem are discrete, with feature  $f_3 = f_2^2$  and  $f_4$  as irrelevant, and the dependent variable  $y$  is given by  $y = f_1 * f_2$ . In our experiments the original dataset was copied 10 times to avoid limitations of statistical tests.

Synthetic ‘‘Corral dataset’’ proposed by John *et al.* [28] has been used to test relevancy and irrelevancy. It has 6 features, and the target concept is defined as the combination of  $[(A0 \wedge A1) \vee (B0 \wedge B1)]$ . Two additional features are Irrelevant (Orr) and Correlated (Cor), introducing 25% error rate (noise). The last of the synthetic datasets, Gauss8, has been used in our previous study [21][29]. Gauss4 is based on sampling of 4 Gaussian functions with unit dispersion in 4 dimensions, each cluster representing a separate class. The first function is centered at (0, 0, 0, 0), the next at (1, 1/2, 1/3, 1/4), (2, 1, 2/3, 1/2), and (3, 3/2, 3, 3/4), respectively. The dataset contains 4000 vectors, 1000 per each class. Gauss8 is an extension of Gauss4, with 4 additional features that are approximately linearly dependent  $f_{i+4} = 2f_i + \epsilon$ , where  $\epsilon$  is a uniform noise.

The summary of results for all datasets and selection algorithms is presented in Tab. 3. For the Shridhar dataset almost all ranking and redundancy shifting methods worked correctly. If the relevancy threshold is set to 0.05, features marked in bold (irrelevant) will be automatically removed. Two algorithms ‘‘ks\_cbf’’ and ‘‘ksc\_cbf’’ should not really be used for this data because the Kolmogorov-Smirnov based tests are designed for continuous features only (yet they still produced reasonable results, selecting

**Table 2.** Dataset used in analysis [27]

| $f_1$ | $f_2$ | $f_3$ | $f_4$ | $y$ | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $y$ | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $y$ |
|-------|-------|-------|-------|-----|-------|-------|-------|-------|-----|-------|-------|-------|-------|-----|
| 0     | 1     | 1     | 2     | 0   | 1     | 2     | 4     | 2     | 2   | 2     | 1     | 1     | 2     | 0   |
| 1     | 1     | 1     | 1     | 1   | 2     | 2     | 4     | 2     | 4   | 0     | 1     | 1     | 1     | 0   |
| 2     | 1     | 1     | 2     | 2   | 0     | 2     | 4     | 1     | 0   | 1     | 1     | 1     | 2     | 1   |

**Table 3.** Ordering of features after feature selection for 3 synthetic datasets

|            |                         | Shridhar dataset [27] |       | Corral dataset [28] |       | Gauss8 dataset [21,29] |       |       |       |                       |     |       |       |       |       |       |       |       |       |
|------------|-------------------------|-----------------------|-------|---------------------|-------|------------------------|-------|-------|-------|-----------------------|-----|-------|-------|-------|-------|-------|-------|-------|-------|
|            | Acronym                 | Most – Less Important |       |                     |       | Most – Less Important  |       |       |       | Most – Less Important |     |       |       |       |       |       |       |       |       |
| Rankings   | pcc                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_6$ | $f_2$ | $f_7$ | $f_3$ | $f_4$ | $f_8$ |
|            | fsc                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_8$ | $f_4$ |
|            | chq                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_4$ | $f_8$ |
|            | mi                      | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_5$ | $f_1$ | $f_2$ | $f_6$ | $f_7$ | $f_3$ | $f_4$ | $f_8$ |
|            | suc                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_4$ | $f_8$ |
|            | mdr                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_4$ | $f_8$ |
|            | kdr                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_4$ | $f_8$ |
|            | kldr                    | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_4$ | $f_8$ |
|            | bdr                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_4$ | $f_8$ |
|            | sdr                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_4$ | $f_8$ |
| Redundancy | mifs ( $\beta = 0.5$ )  | $f_1$                 | $f_2$ | $f_4$               | $f_3$ | cor                    | $B_0$ | $B_1$ | $A_0$ | $A_1$                 | irr | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ | $f_6$ | $f_7$ | $f_8$ |
|            | mifsu ( $\beta = 0.5$ ) | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $B_0$ | $B_1$ | $A_0$ | $A_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_3$ | $f_4$ | $f_6$ | $f_7$ | $f_8$ |
|            | amifs                   | $f_1$                 | $f_2$ | $f_4$               | $f_3$ | cor                    | $B_0$ | $B_1$ | $A_0$ | $A_1$                 | irr | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ | $f_6$ | $f_7$ | $f_8$ |
|            | mid                     | $f_1$                 | $f_2$ | $f_4$               | $f_3$ | cor                    | $B_0$ | $B_1$ | $A_0$ | $A_1$                 | irr | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ | $f_6$ | $f_7$ | $f_8$ |
|            | miq                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_3$ | $f_4$ | $f_6$ | $f_7$ | $f_8$ |
|            | fid                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $B_0$ | $B_1$ | $A_0$ | $A_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_8$ | $f_4$ |
|            | liq                     | $f_1$                 | $f_2$ | $f_3$               | $f_4$ | cor                    | $B_0$ | $B_1$ | $A_0$ | $A_1$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_8$ | $f_4$ |
|            | fcfb                    | $f_1$                 | $f_2$ |                     |       | cor                    | $A_0$ | $A_1$ | $B_0$ | $B_1$                 | irr | $f_1$ | $f_2$ | $f_3$ |       |       |       |       |       |
|            | ks_cbf                  | $f_1$                 | $f_2$ |                     |       | cor                    |       |       |       |                       |     | $f_1$ | $f_2$ | $f_3$ | $f_4$ |       |       |       |       |
|            | ksc_cbf                 | $f_1$                 | $f_2$ |                     |       | cor                    |       |       |       |                       |     | $f_1$ | $f_2$ | $f_3$ | $f_4$ |       |       |       |       |
| Other      | mbr                     | $f_1$                 | $f_3$ | $f_4$               | $f_2$ | cor                    | $B_1$ | $B_0$ | $A_1$ | $A_0$                 | irr | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_8$ | $f_7$ | $f_6$ | $f_5$ |
|            | gdd                     |                       |       | N.A.                |       | cor                    | $A_1$ | $A_2$ | $B_1$ | $B_0$                 | irr | $f_1$ | $f_5$ | $f_2$ | $f_6$ | $f_3$ | $f_7$ | $f_8$ | $f_4$ |

features  $f_1$  and  $f_3$ ). The same applies to the algorithm based on Markov Blanket (mbr), where two most important features selected are  $f_1$  and  $f_3$ . The last method listed (gdd) is very sensitive to the presence of redundant features, but it can still be used as a test for occurrences of duplicated attributes.

For the Corral dataset the optimal solution consists of only 4 features used to define the target function. All algorithms failed to identify them correctly. The correlated feature (cor) was always selected as the most important, followed by 4 relevant features, with the irrelevant feature (irr) as the least important, removed by the relevancy threshold. Algorithms based on statistical tests are again not applicable in this case.

For the Gauss8 dataset an ideal ranking method should give the following order of features:  $f_1 > f_5 > f_2 > f_6 > f_3 > f_7 > f_4 > f_8$ . Moreover the selection algorithms should also reject all 4 linearly dependent features as redundant leaving  $f_1 > f_2 > f_3 > f_4$  order. K-S CBF and a few other methods based on statistical tests completed this task without difficulties, FCBF [20] selected 3 features only, while MI filter placed  $f_5$  at the first position and reversed  $f_3$  and  $f_7$  order. All ranking methods worked as expected.

**Table 4.** Accuracy of 4 classifiers on selected subsets of features for the Gauss8 dataset

| Title    | Selected features |             |             |                      |
|----------|-------------------|-------------|-------------|----------------------|
|          | Full set          | FCBF        | Ranking     | K-S CBF and K-SC CBF |
| Features | 1 to 8            | 1 to 3      | 1 to 8      | 1 to 4               |
| NBC      | 82.1 (+1.2)       | 81.6 (+1.2) | 82.1 (+1.2) | 82.0 (+1.3)          |
| INN      | 73.4 (+13)        | 68.1 (+13)  | 73.4 (+13)  | 73.4 (+13)           |
| C4.5     | 78.3 (+3.9)       | 76.2 (+2.5) | 78.7 (+3.9) | 78.7 (+3.4)          |
| SVM      | 81.9 (+1.3)       | 77.0 (+1.4) | 81.7 (+1.3) | 81.7 (+2.0)          |
| Average  | 79.0 (+4.9)       | 75.7 (+4.6) | 79.0 (+4.9) | 79.0 (+4.9)          |

On top of the selection/ranking results for Gauss8 data described above, 4 different classification methods have been tested: Naive Bayes Classifier (NBC) (Weka implementation, [8]), the nearest neighbor algorithm (1NN) with Euclidean distance function, C4.5 tree (Weka) and the Support Vector Machine with a linear kernel (Ghostminer 3.0 implementation). The results obtained with selected subset of features are presented in Tab. 4.

## 5 Summary

A C++ library (Infosel++) for implementing feature selection algorithms has been presented. The library provides users with the ability to easily test-drive standard feature selection methods, combining them with various predictive methods, and to develop and test novel methods. The software includes algorithms based on estimation of probability distributions, correlation coefficients,  $t$ -score and  $F$ -score, and a few additional functions for estimation of relevancy, plus a dozen of feature selection algorithms that are not easy to find in other software packages. The library, available from authors of this paper, is meant for educational, research and data mining purposes.

Due to the lack of space only results on synthetic data have been presented to verify correctness of algorithm implementation. The methods included in the library have also been tested on real datasets, such as the microarray experiments (gene expression data) and SNP data from Genome-Wide Association Studies (GWAS). The results will be presented in extended version of this paper. More details about the structure of Infosel++ library and examples of programming utilizing its functions will be presented in an extended version of this paper.

**Acknowledgement.** This work has been supported by the Polish Committee for Scientific Research grant 2007-2010 No.: N N519 1506 33. JB is grateful to Larry Haitkamp for fruitful discussion and support during writing of this paper.

## References

1. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L.: Feature Extraction, Foundations and Applications. Studies in Fuzziness and Soft Computing Series. Springer, Heidelberg (2006)
2. Duch, W., Maszczyk, T.: Universal learning machines. In: Chan, J.H. (ed.) ICONIP 2009, Part II. LNCS, vol. 5864, pp. 206–215. Springer, Heidelberg (2009)
3. Saeys, Y., Inza, I., Larrañaga, P.: A review of feature selection techniques in bioinformatics. *Bioinformatics* 23(19), 2507–2517 (2007)
4. Liu, H., Motoda, M. (eds.): Computational Methods of Feature Selection. CRC Press, Boca Raton (2007)
5. Saeys, Y., Liu, H., Inza, I., Wehenkel, L., de Peer, Y.V.: New challenges for feature selection in data mining and knowledge discovery. In: *JMLR Workshop and Conf. Proc.* (2008)
6. Duch, W.: Filter methods. In: Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L. (eds.) Feature extraction, foundations and applications, pp. 89–118. Springer, Heidelberg (2006)
7. Kohavi, R., Sommerfield, D., Dougherty, J.: Data mining using MLC++, a machine learning library in C++. *Int. J. of Artificial Intelligence Tools* 6(4), 537–566 (1997)
8. Witten, I., Frank, E.: *Data Mining: Practical machine learning tools and techniques*, 2nd edn. Morgan Kaufmann, San Francisco (2005)

9. Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M., Euler, T.: YALE: Rapid prototyping for complex data mining tasks. In: Proc. of the 12th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, KDD 2006 (2006)
10. Pudil, P., Novovicova, J., Somol, P.: Feature selection toolbox software package. *Pattern Recognition Letters* 23(4), 487–492 (2002)
11. Su, Y., Murali, T., Pavlovic, V., Schaffer, M., Kasif, S.: Rankgene: identification of diagnostic genes based on expression data. *Bioinformatics* 19, 1578–1579 (2003)
12. Liu, H., Yu, L.: Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans. on Knowledge and Data Engineering* 17(4), 491–502 (2005)
13. Press, W., Teukolsky, S., Vetterling, W., Flannery, G.: *Numerical recipes in C. The art of scientific computing*. Cambridge University Press, Cambridge (1988)
14. Vilmansen, T.: Feature evaluation with measures of probabilistic dependence. *IEEE Transaction on Computers* 22(4), 381–388 (1973)
15. Golub, T., et al.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* 286, 531–537 (1999)
16. Ding, C., Peng, F.: Minimum redundancy feature selection from microarray gene expression data. *Journal of Bioinformatics and Computational Biology* 3(2), 185–205 (2004)
17. Battiti, R.: Using mutual information for selecting features in supervised neural net learning. *IEEE Trans. Neural Networks* 5(4) (July 1994)
18. Kwak, N., Choi, C.H.: Input feature selection for classification problems. *IEEE Transactions on Evolutionary Computation* 13(1), 143–159 (2002)
19. Tesmer, M., Estevez, P.: AMIFS: Adaptive feature selection by using mutual information. In: Proc. of Int. Joint Conf. on Neural Networks, Budapest, pp. 1415–1420. IEEE Press, Los Alamitos (2004)
20. Yu, L., Liu, H.: Efficient feature selection via analysis of relevance and redundancy. *Journal of Machine Learning Research, JMLR* 5, 1205–1224 (2004)
21. Duch, W., Biesiada, J.: Feature Selection for High-Dimensional Data: A Kolmogorov-Smirnov Correlation-Based Filter Solution. In: *Advances in Soft Computing*, pp. 95–104. Springer, Heidelberg (2005)
22. Biesiada, J., Duch, W.: A Kolmogorov-Smirnov correlation-based filter solution for microarray gene expressions data. In: Ishikawa, M., Doya, K., Miyamoto, H., Yamakawa, T. (eds.) *ICONIP 2007, Part II. LNCS*, vol. 4985, pp. 285–294. Springer, Heidelberg (2008)
23. Blachnik, M., Duch, W., Kachel, A., Biesiada, J.: Feature Selection for Supervised Classification: A Kolmogorov-Smirnov Class Correlation-Based Filter. In: *AIMeth, Symposium on Methods of Artificial Intelligence*, Gliwice, Poland, November 10-19 (2009)
24. Koller, D., Sahami, M.: Toward optimal feature selection. In: Proc. of the 13th Int. Conf. on Machine Learning, pp. 284–292. Morgan Kaufmann, San Francisco (1996)
25. Xing, E., Jordan, M., Karp, R.: Feature selection for high-dimensional genomic microarray data. In: Proc. of the 8th Int. Conf. on Machine Learning (2001)
26. Lorenzo, J., Hernandez, M., Mendez, J.: GD: A Measure based on Information Theory for Attribute Selection. In: Coelho, H. (ed.) *IBERAMIA 1998. LNCS (LNAI)*, vol. 1484, pp. 124–135. Springer, Heidelberg (1998)
27. Sridhar, D., Barlett, E., Seagrave, R.: Informatic theoretic subset selection for neural networks models. *Computers & Chemical Engineering* 22(4), 613–626 (1998)
28. John, G., Kohavi, R., Pfleger, K.: Irrelevant features and the subset selection problem. In: Proc. Eleventh Inter. Conf. on Machine Learning, pp. 121–129. Morgan Kaufmann, San Francisco (1994)
29. Biesiada, J., Duch, W.: Feature Selection for High-Dimensional Data: A Pearson Redundancy Based Filter. In: *Advances in Soft Computing*, vol. 45, pp. 242–249. Springer, Heidelberg (2008)

# Stacking Class Probabilities Obtained from View-Based Cluster Ensembles

Heysem Kaya<sup>1,\*</sup>, Olcay Kurşun<sup>2</sup>, and Hüseyin Şeker<sup>3</sup>

<sup>1</sup> Department of Computer Engineering, Bogazici University, 34342, Bebek, Istanbul, Turkey  
pursuing his PhD degree  
heysem@boun.edu.tr

<sup>2</sup> Department of Computer Engineering, Istanbul University, 34320, Avcilar, Istanbul, Turkey  
okursun@istanbul.edu.tr

<sup>3</sup> Bio-Health Informatics Research Group at the Centre for Computational Intelligence,  
Department of Informatics, Faculty of Technology, De Montfort University,  
Leicester, UK, LE19BH

**Abstract.** In pattern recognition applications with high number of input features and insufficient number of samples, the curse of dimensionality can be overcome by extracting features from smaller feature subsets. The domain knowledge, for example, can be used to group some of the features together, which are also known as “views”. The features extracted from views can later be combined (i.e. stacking) to train a final classifier. In this work, we demonstrate that even very simple features such as class-distributions within clusters of each view can serve as such valuable features.

**Keywords:** Feature Extraction; Ensemble Methods; Stacking; Multi-view Learning; ARTMAP; Protein Sub-nuclear Location Classification.

## 1 Introduction

Combining multiple learners using different algorithms, hyper-parameters, sub-problems and training sets is known to enhance generalization in machine learning [1-4]. In some datasets, typical in bioinformatics, high dimensional features are naturally organized into several groups, named as “views” [5] which enable inherent formation of different base-learners. The techniques using multiple views in learning exploit independent properties of each view and more effectively learn complex distributions. In other words, the reason to use multiple views instead of using one single “combined” view is that using views separately to train independent classifiers and then fusing their outputs achieves better generalization accuracy than that of merging all the raw features into a single view [5-8].

Works on multi-view machine learning gained importance since [5] and [6] pointed out that multiple views can lead to better classification accuracy than the full

---

\* This work as part of his MS degree at the Department of Computer Engineering, Bahcesehir University, 34349, Besiktas, Istanbul, Turkey.

set. Multi-view classification attracts many researchers recently because yet there is no known “best” way of fusing the information in multiple views. In [7], it has been showed that multi-view clustering performs better than single view clustering even though the setting contains only two views which they argued either one suffices for learning. Empirical success of multi-view approaches has been noted in many areas of computer science including Natural Language Processing, Computer Vision, and Human Computer Interaction [8]. On the other hand, [9] points out to the importance of reducing the feature space dimensionality to minimum yet descriptive size for effective classification. Therefore, the aim of this study is to further investigate the currently used methods as well as proposing new methods for handling highly dimensional biomedical datasets exploiting their multi-view nature.

The paper layout is as follows. In Section 2 the methods used in this study are briefly reviewed. In Section 3 the experimental results, and in Section 4 conclusions are presented.

## 2 Methods

In this study, a series of methods and techniques were elaborated for extraction and classification purposes. The dataset under study was classified using Fuzzy ARTMAP Neural Network [10-12], k-Nearest Neighbor classifier (k-NN), and Support Vector Machines (SVM).

Due to its simplicity, first, k-NN was used for classification in order to provide insight about problem complexity. The performance of k-NN with full set of features (single view) considered as a baseline.

Later, more complex pattern recognition algorithms, Fuzzy ARTMAP NN and SVM, were applied to single view problem setting of the dataset. Fuzzy ARTMAP NN is capable of fast, stable incremental learning [10-12]. Details regarding Fuzzy ARTMAP can be attained from [10]. CNS Tech Lab implementation [13] of Fuzzy ARTMAP and LIBSVM [14] implementation of SVM was used in the study.

Then, in multi view setting, k-NN ensemble method which was shown to be successful in [15] was used. Furthermore, k-Medoids (k-M) clustering algorithm, which is known to be resistant to outliers, was used to obtain features for subsequent classification. Views compressed via k-M clustering were stacked to Fuzzy ARTMAP. This is done by stacking the class distributions of each view based cluster to Fuzzy ARTMAP, in a setting called k-MART.

The algorithm of k-M is given in Figure 1. For each view, samples of the training set are subjected to clustering by k-M and samples of the test set are assigned to the closest medoid obtained from the training set. Then, the probability distribution of the classes in the cluster each sample falls into are provided to Fuzzy ARTMAP; that is a feature vector of  $C \times V$  length, containing probabilities in [0-1] range are presented to Fuzzy ARTMAP, where  $C$  is the number of classes and  $V$  is the number of views. Figure 2 depicts the architecture of k-MART and Figure 3 presents a pseudo code algorithm for it.

**Algorithm:** k-medoids. PAM, a k-medoids algorithm for partitioning based on medoid or central objects.

**Input:**

$k$ : the number of clusters,

$D$ : a data set containing  $n$  objects.

**Output:** A set of  $k$  clusters.

**Method:**

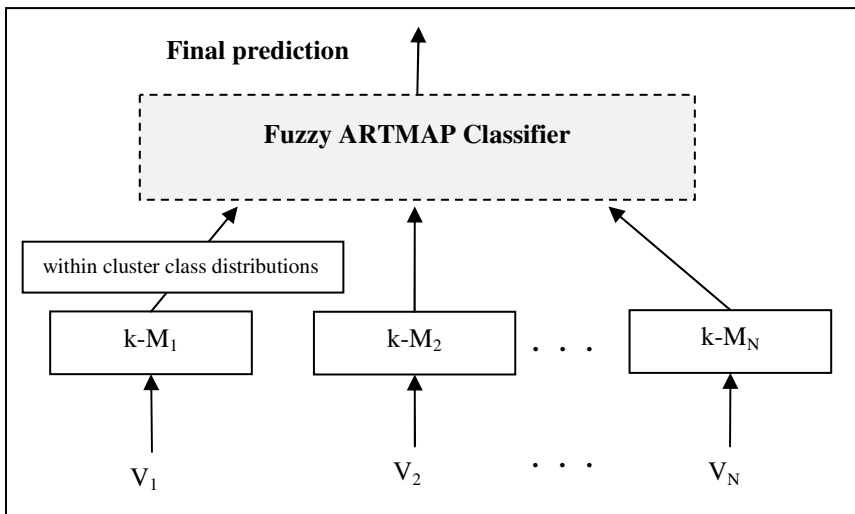
1. arbitrarily choose  $k$  objects in  $D$  as the initial representative objects or seeds
2. **repeat**
3. assign each remaining object to the cluster with the nearest representative object
4. randomly select a non-representative object,  $o_{\text{random}}$
5. compute the minimal total cost,  $S$ , of swapping each representative object,  $o_j$ , with  $o_{\text{random}}$
6. if  $S < 0$  then swap  $o_j$  with  $o_{\text{random}}$  to form the new set of  $k$  representative objects
7. **until** no change

where

$$E = \sum_{j=1}^k \sum_{p \in C_j} |p - o_j| \quad (1)$$

$$S = E_{n+1} - E_n \quad (2)$$

**Fig. 1.** PAM, the k-Medoids partitioning algorithm used in the study [16]



**Fig. 2.** The k-MART stacking architecture. Each view is subjected to clustering independently and class distributions of clusters are stacked to Fuzzy ARTMAP.



**Algorithm:** k-MART, a method for stacking class probabilities obtained from view based k-Medoids clustering to Fuzzy ARTMAP

**Input:**

$nV$ : the number of views,  
 $nL$ : the number of classes (labels),  
 $nC$ : the number of clustering runs for each view,  
 $D$ : the dataset arranged as combination of views

**Output:** Test dataset classification results

**Method:**

```

1. Instantiate and clear 2D double array trns_ds as
 transformed dataset
2. for $i=0$ to $nC-1$
3 randomly select k parameter
4. for $v=0$ to nV
5. cluster current view using K-Medoids and store
 clustering in clstr
6. calculate class distributions within clstr and
 store it in distrib
7. for $j=0$ to $|D|-1$
8. for $c=0$ to $nL-1$
9. $trns_ds[j][nV*v+c] += distrib[j][c]$ //add the
 c^{th} class distribution data of j^{th} sample in
 view v to trns_ds
10. end
11. end
12. end
13.end
14.calculate average of nC runs for trns_ds
15.train and test trns_ds in Fuzzy ARTMAP
16.return test results

```

---

**Fig. 3.** k-MART, a method for stacking k-M clustering to Fuzzy ARTMAP

Since k-M clustering is stochastic, it gives different results at each run. In order to overcome this deviation, we averaged cluster class distributions over sufficiently many clustering runs, each of which are instantiated with random  $k$  values drawn from a reasonable range that can be determined empirically.

### 3 Experimental Results

A real biomedical dataset, Protein Sub-nuclear Location dataset [17] was used due to its challenging multi-view structure. k-NN tests involved  $k$  values ranging from 1 to 11. For ARTMAP, learning rate ( $\beta$ ) was left to its default value of 1 and four baseline vigilance ( $\bar{\rho}$ ) was tested, namely {0.25, 0.50, 0.75, 0.99}. Since the best results were obtained with 0.99 for the single-view setting, only this value was used in stacking k-M clustering class probabilities.

### 3.1 Protein Dataset

Dataset is composed of 714 samples having 1497 features categorized in 53 views. Views of this dataset are sub-categories of the following 8 main set of views: (1) Amino acid composition, (2) Dipeptide composition, (3) Normalized moreau-broto correlation, (4) Moran autocorrelation, (5) Geary autocorrelation, (6) Composition, Transition & Distribution, (7) Sequence Order, and (8) Pseudo amino acid composition. Recently [18] proposed a method exploiting multi view nature of this dataset using SVMs using these 8 main views. [18] pointed out to the challenge of 53-view setting stating that sub-categories are not considered to be sufficient enough to learn the complex target concept individually. Since, sub-categories are more specific and contain partial information about the target. Therefore, this study aims at finding a less complicated method to a more challenging setting of the problem.

Owing to the under sampled and highly dimensional nature of Protein Dataset (see Table 1), this dataset was randomly bipartitioned so as to obtain 2-fold results and higher number of folds were not intended.

Table 1 depicts the challenge which is typical to bioinformatics. The classes are unevenly distributed and most of them are under sampled. Combined with curse of dimensionality, these conditions pose a great challenge for pattern recognition.

As stated before, in order to gain insight about problem complexity a simple pattern recognition algorithm, k-NN, was used. After testing all views with one classifier, a k-NN classifier for each view was instantiated and an ensemble was formed. Validating [15], through multi view ensemble, we attained higher prediction results than the best single-view k-NN baseline of 49.84% obtained with  $k=3$ .

**Table 1.** Class Distribution of Protein Dataset

| ID           | Description          | Training   | Test       | Total      |
|--------------|----------------------|------------|------------|------------|
| 1            | Chromatin            | 50         | 49         | <b>99</b>  |
| 2            | Heterochromatin      | 11         | 11         | <b>22</b>  |
| 3            | Nuclear Envelope     | 31         | 30         | <b>61</b>  |
| 4            | Nuclear Matrix       | 15         | 14         | <b>29</b>  |
| 5            | Nuclear Pore Complex | 40         | 39         | <b>79</b>  |
| 6            | Nuclear Speckle      | 34         | 33         | <b>67</b>  |
| 7            | Nucleolus            | 154        | 153        | <b>307</b> |
| 8            | Nucleoplasm          | 19         | 18         | <b>37</b>  |
| 9            | Nuclear PML Body     | 7          | 6          | <b>13</b>  |
| <b>Total</b> |                      | <b>361</b> | <b>353</b> | <b>714</b> |

**Table 2.** k-NN Results for Protein Dataset

| Method / k             | 1             | 3             | 5             | 7             | 9             | 11            |
|------------------------|---------------|---------------|---------------|---------------|---------------|---------------|
| <b>Full Set Fold 1</b> | 48.44%        | 48.44%        | 47.59%        | 47.03%        | 47.88%        | 49.29%        |
| <b>Full Set Fold 2</b> | 48.20%        | 51.25%        | 50.97%        | 48.48%        | 45.98%        | 48.20%        |
| <b>Average</b>         | <b>48.32%</b> | <b>49.84%</b> | <b>49.28%</b> | <b>47.75%</b> | <b>46.93%</b> | <b>48.75%</b> |
| <b>Ensemble Fold 1</b> | 54.11%        | 51.27%        | 49.58%        | 47.88%        | 47.31%        | 47.31%        |
| <b>Ensemble Fold 2</b> | 55.40%        | 53.74%        | 53.74%        | 52.35%        | 51.80%        | 49.58%        |
| <b>Average</b>         | <b>54.75%</b> | <b>52.51%</b> | <b>51.66%</b> | <b>50.11%</b> | <b>49.55%</b> | <b>48.45%</b> |

After k-NN investigation, a series of single view tests were done using Fuzzy ARTMAP and SVM. In Fuzzy ARTMAP, the prediction success was found to increase with increasing  $\bar{p}$  in fluctuating manner. The best accuracy attained ( $\bar{p} = 0.99$ , 54.59%) was better than the baseline k-NN but slightly lower than the k-NN ensemble. Single view SVM test was carried out with RBF kernel with the kernel bandwidth,  $g$  parameter, was left to its default value  $1/k$ , where  $k$  is the number of features; and  $C$  parameter is set to 50 in accordance with [18]. SVM in single view setting was found to give better accuracy than Fuzzy ARTMAP (see Table 3). However, in 53-view setting SVM ensemble accuracy was 43.9%. Finally, k-MART was applied using 50 runs and randomly selected  $k$  values from 50-90 range. As it can be seen in Table 3, using class probabilities in a stacking framework yields better recognition rates. An average increase of 4.49 points from single view Fuzzy ARTMAP and 2.66 points from single view SVM was attained using this framework.

**Table 3.** Best Accuracies of Different Methods for Protein Dataset

| Fold           | k-NN          | k-NN Ensemble | SVM           | SMV Ensemble  | Fuzzy ARTMAP  | k-MART        |
|----------------|---------------|---------------|---------------|---------------|---------------|---------------|
| 1              | 48.44%        | 54.11%        | 54.39%        | 43.34%        | 52.12%        | 57.22%        |
| 2              | 51.25%        | 55.40%        | 58.45%        | 44.47%        | 57.06%        | 60.94%        |
| <b>Average</b> | <b>49.84%</b> | <b>54.75%</b> | <b>56.42%</b> | <b>43.90%</b> | <b>54.59%</b> | <b>59.08%</b> |

As seen in Table 3, SVM behaves different than the other two algorithms. SVM yields the best single view accuracy, while the accuracy decreases dramatically in 53-view ensemble setting which was formed by simple voting of view-based class probability estimations. Relative single view success of SVM can be attributed to effective handling of high dimensional dataset. High error rate in ensemble setting implies the insufficiency of partially informative individual views.

## 4 Conclusions

Clustering is a simple and unsupervised preprocessing method. We have shown that class probabilities can be extracted via an ensemble of many clustering runs which are instantiated with randomly chosen hyper-parameters. We have also shown that these probabilities can also serve as “features” that can be stacked; that is, they can be used to train a final classifier in order to be fused for a single ultimate prediction for the class labels. Moreover, using unsupervised tools, such as the ones described in this paper, reduces the risk of over-fitting for this subsequent learning task. We have demonstrated that our method performs better than using k-NN ensembles which is known to be significantly better than simply taking the union of all views to get a single feature vector to learn from. The results are in accordance with [19] which demonstrated that efficient stacking techniques perform better than selecting the best classifier from the ensemble.

## References

1. Alpaydm, E.: *Introduction to Machine Learning*. MIT Press, Cambridge (2004)
2. Berthold, M., Hand, D.J. (eds.): *Intelligent data analysis: an introduction*. Springer, Berlin (1999)
3. Bishop, C.M.: *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford (1995)
4. Hastie, T., Tibshirani, R., Friedman, J.H.: *The Elements of Statistical Learning: Data mining, Inference, and Prediction: with 200 Full-Color Illustrations*. Springer, New York (2001)
5. Yarowsky, D.: Unsupervised word sense disambiguation rivaling supervised methods. In: *Proceedings of the Thirty-Third Annual Conference of the Association for Computational Linguistics*, Cambridge, MA, pp. 189–196 (1995)
6. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, Madison, WI, pp. 92–100 (1998)
7. Bickel, S., Scheffer, T.: Multi-view clustering. In: *Proceedings of the Forth IEEE International Conference on Data Mining*, Brighton, UK, pp. 19–26 (2004)
8. Christoudias, C.M., Urtasun, R., Darrell, T.: Multi-View Learning in the Presence of View Disagreement. In: *Proceedings of the Conference on Uncertainty in Artificial Intelligence* (2008)
9. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. *Journal of Machine Learning Research* 3, 1157–1182 (2003)
10. Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J.H., Rosen, D.B.: Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps. *IEEE Transactions on Neural Networks* 3(5), 698–713 (1992)
11. Carpenter, G.A., Grossberg, S., Reynolds, J.H.: ARTMAP: Supervised Real-Time Learning and Classification of Nonstationary Data by a Self-Organizing Neural Network. *Neural Networks* 4, 565–588 (1991)
12. Carpenter, G.A., Grossberg, S., Rosen, D.B.: Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks* 4, 759–771 (1991)
13. The Boston University Department of Cognitive and Neural Systems Technology Laboratory, <http://techlab.bu.edu/classer/>
14. Hsu, C.W., Lin, C.J.: A Comparison of Methods for Multi-Class Support Vector Machines. *IEEE Trans. Neural Networks* 13, 415–425 (2002)
15. Okun, O., Priisalu, H.: Multiple Views in Ensembles of Nearest Neighbor Classifiers. In: *Proceedings of the ICML Workshop on Learning with Multiple Views (in conjunction with the 22nd International Conference on Machine Learning)*, Bonn, Germany, pp. 51–58 (2005)
16. Jiawei, H., Kamber, M.: *Data Mining: Concepts and Techniques*, pp. 310–312, 406. Morgan Kaufmann Publishers, San Francisco (2006)

17. Nanuwa, S., Seker, H.: Investigation into the role of sequence-driven-features for prediction of protein structural classes. In: 8th IEEE International Conference in Bioinformatics and Bioengineering, Athens, Greece (2008)
18. Sakar, C.O., Kursun, O., Seker, H., Gorgen, F., Aydin, N.: Parallel Interacting Multiview Learning: An Application to Prediction of Protein Sub-nuclear Location. In: Proceedings of the 9th International Conference on Information Technology and Applications in Biomedicine (ITAB), Larnaca, Cyprus (2009)
19. Džeroski, S., Ženko, B.: Is Combining Classifiers with Stacking Better than Selecting the Best One? *Machine Learning* 54(3), 255–273 (2004)

# Market Trajectory Recognition and Trajectory Prediction Using Markov Models

Przemysław Klęsk and Antoni Wiliński

Faculty of Computer Science and Information Systems,  
Westpomeranian University of Technology,  
ul. Żołnierska 49, 71-210, Szczecin, Poland  
{pklesk,awilinski}@wi.zut.edu.pl

**Abstract.** We analyze a time series data set from a financial market. The set contains over sixty thousand OHLC candles of EUR/USD currency pair collected once every hour during the period of ten consecutive years. We build Markov models for this data and consider two approaches of analysis respectively for: *trajectory recognition* and *trajectory prediction*. We provide results of our experiments and comment on them.

## 1 Introduction

### 1.1 The Data Set

The task of financial market prediction by technical analysis has been a challenge for thousands of analysts over decades, and in recent years for millions of internet trade practitioners. The task is considered to be very difficult to be solved exactly [3]. However, many financial mathematicians find it feasible to obtain practically useful predictions, even with application of simple rules/patterns [11,13]. In [5,4] the range of techniques is extended onto *japanese candles*[4]. Candles can be graphed in different forms [10,4], which often is a source of inspiration for investing strategies, e.g. [10]. Patterns and configurations of candles are commonly applied for perceptual assessment of the market [12,8,2] and are rather simple tools. Among more sophisticated methods (especially in academic community) is the dominating analysis of regression[2,11,14].

Authors of this paper propose an approach based on Markov models, in which the training procedure takes into account the whole history of former states (not barely a certain fragment) and transitions between them. The approach is based solely on time series quotations, with no information support of fundamental character, hence: a pure technical analysis approach. Experiments were carried out on a time series consisting of 62735 candles (collected once every hour) for the EUR/USD currency pair, which corresponds to approx. 10 years period[2]. Some first rows of the time series data are shown in the fig. 1. For further considerations, we denote the number of observed dimensions by  $n = 4$ .

<sup>1</sup> Formally, it is a 4-dimensional vector OHLC (*Open, High, Low, Close*) denoting: the observable value of a variable at the start of a sampling period (O), at the end (C), its maximal value within this period (H), its minimal value within this period (L).

<sup>2</sup> Taking into consideration gaps in the world trading.

| time moment<br>$t$ | O<br>opening quotation<br>$x_1$ | H<br>maximal quotation<br>$x_2$ | L<br>minimal quotation<br>$x_3$ | C<br>closing quotation<br>$x_4$ |
|--------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|
| 1                  | 1.1801                          | 1.1819                          | 1.1769                          | 1.1789                          |
| 2                  | 1.1795                          | 1.1819                          | 1.1780                          | 1.1791                          |
| 3                  | 1.1791                          | 1.1803                          | 1.1788                          | 1.1798                          |
| 4                  | 1.1797                          | 1.1804                          | 1.1781                          | 1.1782                          |
| 5                  | 1.1780                          | 1.1820                          | 1.1775                          | 1.1802                          |
| ⋮                  | ⋮                               | ⋮                               | ⋮                               | ⋮                               |

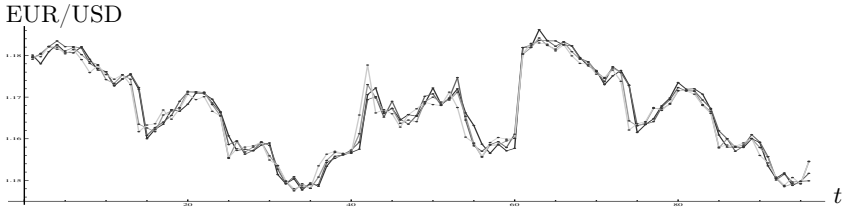


Fig. 1. Data set sample and a plot along time

### 1.2 Stochastic Processes, Markov Models

A set of random values  $\{q(t)\}$  which depend only on one parameter is called a *random process* or a *stochastic process*. Commonly, the  $t$  parameter is identified with time [6]. A *discrete Markov Model* or a *Markov Chain* is a special case of a general random process with the following three constraints [6,9,7]:

1. both the time and the set of possible values — states — in the sequence are discrete:  $q(t) \in \{S_1, S_2, \dots, S_N\}, t = 1, 2, \dots, T$  — *stochastic chains*;
2. the probability of transition to a certain state depends only on the former state, not on the whole history:  $P(q(t) = S_j | q(t-1) = S_i, q(t-2) = S_k, \dots) = P(q(t) = S_j | q(t-1) = S_i)$  — *Markov chains*;
3. probabilities of transitions are constant in time:  $\forall t P(q(t) = S_j | q(t-1) = S_i) = \text{const.} = a_{ij}, 1 \leq i, j \leq N$  — *stationary chains*.

Clearly, in case of systems like trade markets the stationarity condition is probably not met and commonly they are considered non-stationary [3,11,3,2].

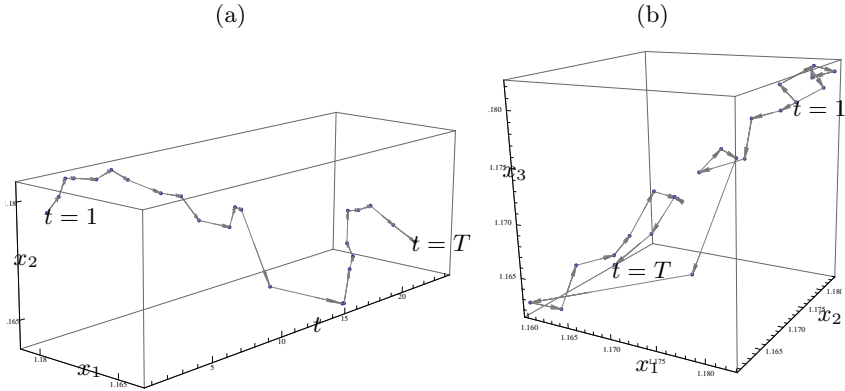
More simply, one can regard a Markov Model as a pair  $(A, \pi)$ , where  $A = \{a_{ij}\}$  is the matrix of transition probabilities:  $a_{ij} = P(q(t) = S_j | q(t-1) = S_i) \quad \forall t$ ; and  $\pi = (\pi_1, \pi_2, \dots, \pi_N)$  is the probability distribution for the initial time moment  $t = 1$ ;  $\pi_i = P(q(1) = S_i)$ . This distribution reflects our belief or exact knowledge about occurrence of particular states in  $t = 1$ . In particular,  $\pi$  can be a one-point distribution if we start from a fixed state with certainty.

### 1.3 Trajectories

One should understand the system of currency-pair quotations as a dynamic system with discrete time. If we take several variables (dimensions)  $x_1, x_2, \dots, x_n$

<sup>3</sup> Despite the fact that stationarity condition is not necessarily satisfied, the approach presented by authors of this paper must not be seen as useless, if only multiple test results (with different parameters, weights, discretizations, etc.) prove otherwise.

under consideration in such a system then we may think of a sequence of quotations as of a multidimensional *trajectory* progressing along time, see the fig. 2.



**Fig. 2.** Examples of trajectories: (a) trajectory in two selected dimensions  $x_1 \times x_2$  along time, (b) trajectory in  $x_1 \times x_2 \times x_3$  (time jumps along the curve)

## 2 Data Preprocessing, Parameters, Notation

Instead of looking at the values of quotations, we prefer to look at their growths or falls i.e. *velocities*. The velocity can be regarded as the first derivative of a quotation over discrete time. Thus, for each row, except the last one, we calculate:

$$v_i(t) = \Delta x_i(t) = x_i(t + 1) - x_i(t), \quad 1 \leq i \leq n, \quad 1 \leq t \leq T - 1. \quad (1)$$

Velocities<sup>4</sup> are in fact the *returns* after each time step and for given currency pair are measured in *pips* — changes on the fourth decimal place.

Now, we want to transform the time series to become discrete as regards the values in the sequence (not only the time), so that we can later build a Markov chain for it. First, we calculate the mean absolute velocity as:

$$|\bar{v}_i| = \frac{1}{T - 1} \sum_{t=1}^{T-1} |v_i(t)|. \quad (2)$$

Next, we relate each velocity  $v_i(t)$  to its absolute mean  $|\bar{v}_i|$ . Let  $w_1 < w_2$  be two positive coefficients, then an exemplary discretization can be done as follows:

<sup>4</sup> It is worth to note that the definition above causes a radical change of relation between all four dimensions of the time series. It disturbs the natural correlation between these dimensions.




$$q_i(t) = \begin{cases} 2, & \text{for } w_2 \leq \frac{v_i(t)}{|v_i|}; \\ 1, & \text{for } w_1 \leq \frac{v_i(t)}{|v_i|} < w_2; \\ 0, & \text{for } -w_1 \leq \frac{v_i(t)}{|v_i|} < w_1; \\ -1, & \text{for } -w_2 \leq \frac{v_i(t)}{|v_i|} < -w_1; \\ -2, & \text{for } \frac{v_i(t)}{|v_i|} < -w_2, \end{cases} \quad (3)$$

where  $q_i(t)$  denotes the obtained discretized value.

The selection of  $w_1, w_2$  coefficients can be experimental. In our experiments we tried several pairs, mostly:  $w_1 = 0.25, w_2 = 0.5$  or  $w_1 = 0.75, w_2 = 1.5$ , depending on what sensitivity to the change of velocity we wanted to have. Obviously, instead of discretizing each dimension to possibilities  $\{-2, -1, 0, 1, 2\}$ , one might introduce a discretization only to  $\{-1, 0, 1\}$ , using just  $w_1$ , or a discretization to  $\{-3, -2, -1, 0, 1, 2, 3\}$ , using certain  $w_1, w_2, w_3$ . It all depends on how much information we want to keep or to cut out. Clearly, by distinguishing more levels of velocity (first derivative), we have a better information about acceleration i.e. the second derivative of quotation over discrete time. After transformations, a resulting time series may look like this:  $(-2, 0, 2, 0) \rightarrow (-1, -2, 2, 2) \rightarrow (2, 0, -2, -2) \rightarrow \dots \rightarrow (1, -2, -2, 0)$ .

In the sections to follow, we describe how Markov models are built and trained using such sequences. It is sufficient to tell now that each unique Markov *state* in these models, will simply be a *point* in a certain 4-dimensional trajectory.

### 3 Trajectory Recognition

In each experiment (no matter what parametrization or discretization) three models were built for classes/contexts: *fall* (bear market), *no change*, *rise* (bull market). Let us name these models  $\lambda_{-1}, \lambda_0, \lambda_1$  respectively<sup>5</sup>. Such contexts are characteristic for trading at stock markets and broker’s platforms <sup>6</sup>.

For each of these models, separate training and testing sets were prepared. They consisted of many trajectories (*tufts of trajectories*), each of certain length  $T$ . We experimented with several lengths ranging from  $T = 4$  to  $T = 24$ . The training procedure was supervised. To assign a certain trajectory  $Q = (q(1), q(2), \dots, q(T))$  to a particular context (fall, no change, rise), we were taking the next state  $q(T + 1)$  and we were looking at its  $i$ -th dimension, the one we wanted to reason about. When  $q_i(T + 1) < 0$  we assigned the trajectory  $Q$  to the context *fall*, when  $q_i(T + 1) = 0$  then to *no change*, when  $q_i(T + 1) > 0$  then to *rise*. In the example below, we show a trajectory assigned to the context *rise*, the dimension of interest was  $i = 4$  — closing value of the candle:

<sup>5</sup> For discretization  $\{-2, \dots, 2\}$ , still only 3 models were built for *fall, no change, rise*.

<sup>6</sup> The mentality of investors causes a relevant fear of loss and relatively smaller satisfaction from success. Thus, most investors are interested in frequent gains (positive returns) and do not especially care if this gain is small.

$$(-2, 0, 2, 0) \rightarrow (-1, -2, 2, 2) \rightarrow (2, 0, -2, -2) \rightarrow \dots \rightarrow (1, -2, -2, 0) \rightarrow (2, 0, -1, \mathbf{1}) .$$

$t = 1$                        $t = 2$                        $t = 3$                                        $t = T$                        $t = T + 1$

The closing value is the typical choice. Other choices  $i = 1, 2, 3$  may possibly be innovations and authors plan to research this in future. After the training, the goal is to recognize new coming trajectories i.e. given an unknown trajectory we want to correctly guess its  $q_4$  for the next time moment  $t = T + 1$ , namely: is it in the context fall, no change or rise.

### 3.1 Training Procedure

For non-hidden Markov models the training procedure consists of simple counting of conditional probabilities. When a set of training sequences  $\{Q^1, Q^2, \dots, Q^K\}$  is given, where  $Q^k = (q^k(1), q^k(2), \dots, q^k(T_k))$ , the formulas are [7]:

$$a_{ij} = \frac{\sum_{k=1}^K \sum_{t=1}^{T_k-1} 1}{\sum_{k=1}^K \sum_{\substack{t=1 \\ q^k(t)=S_i}}^{T_k-1} 1}, \quad 1 \leq i, j \leq N. \tag{4}$$

$$\pi_i = \frac{1}{K} \sum_{\substack{k=1 \\ q^k(1)=i}}^K 1, \quad 1 \leq i \leq N. \tag{5}$$

The complexity of such a calculation is  $O(\sum_{k=1}^K T_k)$  [8].

### 3.2 Trajectory Recognition Results

We start with an important remark. When recognizing a given trajectory  $Q$ , one may look at the maximal probability of this trajectory conditional on model:

$$\arg \max_{i=-1,0,1} P(Q|\lambda_i) \tag{6}$$

or at the maximal a posteriori probability [9]:

$$\arg \max_{i=-1,0,1} P(Q|\lambda_i) \cdot P(\lambda_i). \tag{7}$$

In both cases:  $P(Q|\lambda) = \pi_{q(1)} a_{q(1),q(2)} a_{q(2),q(3)} \dots a_{q(T-1),q(T)}$ .

<sup>7</sup> In situations when new states occurred in testing (non-identified in the training), we introduced an additional single state for all of such states. Let us name it ‘?’-state. We set transition probabilities (from/to) this state to a certain  $\epsilon$  with order of magnitude  $\epsilon \sim 1 /$  (number of distinct identified states in the training data  $\cdot 10$ ).

<sup>8</sup> Formulas above are equivalent to the Baum-Welch reestimation procedure for hidden Markov models with *emission matrix* set up to an identity.

<sup>9</sup> To be precise: we look just at the numerator of the a posteriori probability formula.

**Table 1.** Results for trajectory recognition approach

| no. | dimensions used      | discretization        | $(w_1)$ or $(w_1, w_2)$ | $N$ | $T$ | correctly recognized as <i>fall</i> $\lambda_{-1}$ | correctly recognized as <i>no change</i> $\lambda_0$ | correctly recognized as <i>rise</i> $\lambda_1$ | fake money simulation, gain per step   |
|-----|----------------------|-----------------------|-------------------------|-----|-----|----------------------------------------------------|------------------------------------------------------|-------------------------------------------------|----------------------------------------|
| 1   | $x_4$                | $\{-1, 0, 1\}$        | (0.25)                  | 4   | 4   | 35.5%                                              | 36.6%                                                | 32.9%                                           | $4.8 \cdot 10^{-5}$                    |
| 2   | $x_4$                | $\{-2, -1, 0, 1, 2\}$ | (0.25, 0.5)             | 6   | 4   | 38.3%                                              | 47.1%                                                | 30.0%                                           | <b><math>9.98 \cdot 10^{-5}</math></b> |
| 3   | $x_4$                | $\{-1, 0, 1\}$        | (0.75)                  | 4   | 4   | 28.9%                                              | 58.8%                                                | 30.4%                                           | $5.24 \cdot 10^{-5}$                   |
| 4   | $x_4$                | $\{-2, -1, 0, 1, 2\}$ | (0.25, 0.5)             | 6   | 8   | 33.6%                                              | 49.6%                                                | 33.7%                                           | $5.67 \cdot 10^{-5}$                   |
| 5   | $x_4$                | $\{-2, -1, 0, 1, 2\}$ | (0.75, 1.5)             | 6   | 8   | 16.7%                                              | 58.8%                                                | 30.0%                                           | <b><math>-2.2 \cdot 10^{-6}</math></b> |
| 6   | $x_1, x_4$           | $\{-2, -1, 0, 1, 2\}$ | (0.25, 0.5)             | 26  | 4   | 36.6%                                              | 33.5%                                                | 36.3%                                           | $4.71 \cdot 10^{-5}$                   |
| 7   | $x_1, x_4$           | $\{-1, 0, 1\}$        | (0.75)                  | 10  | 4   | 31.8%                                              | 55.7%                                                | 34.0%                                           | <b><math>6.16 \cdot 10^{-5}</math></b> |
| 8   | $x_1, x_4$           | $\{-2, -1, 0, 1, 2\}$ | (0.75, 1.5)             | 26  | 4   | 28.9%                                              | 58.0%                                                | 30.9%                                           | <b><math>-2.6 \cdot 10^{-6}</math></b> |
| 9   | $x_1, x_4$           | $\{-1, 0, 1\}$        | (0.25)                  | 10  | 4   | 34.4%                                              | 33.4%                                                | 36.6%                                           | $5.03 \cdot 10^{-5}$                   |
| 10  | $x_1, x_4$           | $\{-2, -1, 0, 1, 2\}$ | (0.25, 0.5)             | 26  | 24  | 26.1%                                              | 41.0%                                                | 41.6%                                           | $5.03 \cdot 10^{-5}$                   |
| 11  | $x_1, x_2, x_4$      | $\{-1, 0, 1\}$        | (0.25)                  | 28  | 4   | 36.4%                                              | 42.0%                                                | 31.3%                                           | $5.9 \cdot 10^{-5}$                    |
| 12  | $x_1, x_2, x_4$      | $\{-2, -1, 0, 1, 2\}$ | (0.25, 0.5)             | 126 | 4   | 35.5%                                              | 32.3%                                                | 36.6%                                           | <b><math>6.43 \cdot 10^{-5}</math></b> |
| 13  | $x_1, x_2, x_4$      | $\{-1, 0, 1\}$        | (0.25)                  | 28  | 8   | 34.4%                                              | 43.2%                                                | 35.3%                                           | $5.4 \cdot 10^{-5}$                    |
| 14  | $x_1, x_2, x_4$      | $\{-2, -1, 0, 1, 2\}$ | (0.25, 0.5)             | 126 | 8   | 31.2%                                              | 31.9%                                                | 38.5%                                           | $5.48 \cdot 10^{-5}$                   |
| 15  | $x_1, x_3, x_4$      | $\{-1, 0, 1\}$        | (0.25)                  | 28  | 8   | 35.5%                                              | 37.3%                                                | 35.5%                                           | <b><math>6.09 \cdot 10^{-5}</math></b> |
| 16  | $x_1, x_2, x_3, x_4$ | $\{-1, 0, 1\}$        | (0.75)                  | 82  | 4   | 31.3%                                              | 54.2%                                                | 34.0%                                           | $4.91 \cdot 10^{-5}$                   |

In the second case, the result is additionally weighted by the *a priori* probabilities of classes. E.g. with the discretization  $\{-1, 0, 1\}$  using  $w_1 = 1.5$ , we observed in the data the distribution:  $P(\lambda_{-1}) = 11.7\%$ ,  $P(\lambda_0) = 75.1\%$ ,  $P(\lambda_1) = 13.2\%$ . So, due to the fact that the context 0 (*no change*) is the most frequent, then obviously it is the easiest context to classify<sup>10</sup>. Clearly, the second approach is unfair for our problem, since we are mostly interested in recognizing *rises* and *falls*<sup>11</sup>, not the *no-changes*.

We remark that for discretization  $\{-1, 0, 1\}$  using  $w_1 = 0.5$ , the observed *a priori* distribution was:  $P(\lambda_{-1}) = 31.2\%$ ,  $P(\lambda_0) = 34.5\%$ ,  $P(\lambda_1) = 34.3\%$ , hence closer to uniform. The class 0 does not have such advantage, and both approaches become similar. Furthermore, for  $w_1 = 0.25$ , the distribution was  $P(\lambda_{-1}) = 38.7\%$ ,  $P(\lambda_0) = 21.0\%$ ,  $P(\lambda_1) = 40.3\%$  — “focused” on falls and rises.

In table 1 we show results of trajectories recognition. The variable of interest is  $q_4$  — session closing quotation. All results are stated for testing sets (30% of all data) and calculated according to (6). Apart from correct recognition ratios we also show the result of simulation for a fake money player. Whenever the recognition indicated a *fall* or a *rise*, we performed a suitable buy/sell transaction. Then we checked the actual value of quotation for the next time moment and accordingly we registered a suitable gain or loss. When the recognition indicated *no change*, no transaction was made. The ‘play’ was carried out for  $10^3 \div 10^4$  time steps, depending on computational cost. In the table we bold out most interesting results. Although most models resulted in a positive pay-off, there occurred certain models with a negative pay-off (shown as well).

<sup>10</sup> A drastic way to look at (7) is to blindly always respond with the class 0. By doing this we obtain a classifier with 75.1% correctness, assuming *i.i.d.* data.

<sup>11</sup> A trader can profit *both* from rise and fall if he guesses the direction of change right.

## 4 Trajectory Prediction

In this approach we do not build separate models for: fall, no change, rise; but just one model  $\lambda$ . Given a trajectory  $Q = (q(1), \dots, q(T))$ , we can use its *tail* to try to predict a states probability distribution for the moment  $t = T + 1$ .

At extreme, one can use only the very last piece of information in the tail, from  $t = T$ . Denote the states probability distribution at  $t = T$  by  $\pi(T) = (0, \dots, 0, 1, 0, \dots, 0)$ , where  $\pi_i(T) = 1$  if  $q(T) = S_i$ , otherwise  $\pi_i(T) = 0$ . The distribution for the following moment  $t = T + 1$  is then:  $\pi(T + 1) = \pi(T) \cdot A$ . If we start earlier in the tail, from  $t = T - 1$ , where  $\pi_i(T - 1) = 1$  if  $q(T - 1) = S_i$  and  $\pi_i(T - 1) = 0$  otherwise, then after two steps of time the distribution is:  $\pi(T + 1) = \pi(T - 1) \cdot A^2$ . In general, if we start from  $t = T - h$ , the distribution after  $h + 1$  steps becomes:

$$\pi(T + 1) = \pi(T - h) \cdot A^{h+1}. \quad (8)$$

We propose to use a linear combination of several distributions like the above. Then, the final result of prediction is the state  $S^*$ , for which the maximum probability of the combined distribution is attained:

$$S^* = \arg \max \sum_{h=0}^H c_h \pi(T - h) \cdot A^{h+1}. \quad (9)$$

In our experiments, the coefficients  $c_h$  were set to  $1/(H + 1)$  (uniform weighing), but they could be set otherwise if one wants to explicitly indicate that some time moments in the tail are more important than others. We considered  $H = 0, 1, 2$  i.e. tails of length at maximum three.

**Table 2.** Results for trajectory prediction approach

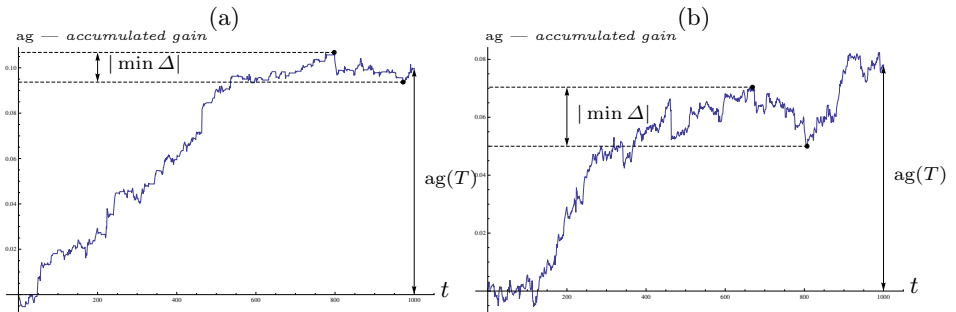
| no. | dimensions used | discretization        | $(w_1)$ or $(w_1, w_2)$ | $N$ | $h$ | correctly predicted falls | correctly predicted no changes | correctly predicted rises | fake money simulation, gain per step |
|-----|-----------------|-----------------------|-------------------------|-----|-----|---------------------------|--------------------------------|---------------------------|--------------------------------------|
| 1   | $x_4$           | $\{-1, 0, 1\}$        | (0.25)                  | 4   | 1   | 46.2%                     | 0.0%                           | 68.4%                     | $4.68 \cdot 10^{-5}$                 |
| 2   | $x_4$           | $\{-2, -1, 0, 1, 2\}$ | (0.25, 0.5)             | 6   | 1   | 54.1%                     | 0.0%                           | 60.5%                     | $7.68 \cdot 10^{-5}$                 |
| 3   |                 |                       |                         |     | 2   | 51.1%                     | 0.0%                           | 63.1%                     | $6.54 \cdot 10^{-5}$                 |
| 4   |                 |                       |                         |     | 3   | 50.5%                     | 0.0%                           | 63.9%                     | $6.8 \cdot 10^{-5}$                  |
| 5   | $x_1, x_4$      | $\{-1, 0, 1\}$        | (0.25)                  | 10  | 1   | 56.6%                     | 0.0%                           | 58.6%                     | $7.34 \cdot 10^{-5}$                 |
| 6   |                 |                       |                         |     | 2   | 56.0%                     | 0.0%                           | 59.4%                     | $7.8 \cdot 10^{-5}$                  |
| 7   |                 |                       |                         |     | 3   | 55.7%                     | 0.0%                           | 59.4%                     | $7.56 \cdot 10^{-5}$                 |
| 8   | $x_1, x_4$      | $\{-2, -1, 0, 1, 2\}$ | (0.25, 0.5)             | 26  | 1   | 54.4%                     | 17.5%                          | 46.7%                     | $7.02 \cdot 10^{-5}$                 |
| 9   |                 |                       |                         |     | 2   | 53.8%                     | 18.7%                          | 45.4%                     | $7.54 \cdot 10^{-5}$                 |
| 10  |                 |                       |                         |     | 3   | 54.1%                     | 17.1%                          | 46.7%                     | $5.52 \cdot 10^{-5}$                 |

In table 2 we show results for the prediction approach. Again, in the last column the outcome of fake money play is shown. We present only models based either on  $x_4$  alone or on  $x_1 \times x_4$ , since most models based on more dimensions including  $x_2$  and  $x_3$  revealed negative pay-offs in prediction experiments.

## 5 Summary

We modeled a EUR/USD currency-pair system with Markov chains. Two approaches of analysis were carried out: *trajectory recognition* and *trajectory prediction*. Although the results do not appear very satisfactory, still it was interesting to note that in both approaches a play with a positive pay-off was possible. Most models showed a gain of order  $0.5 \div 1$  pip per step<sup>12</sup>. Obviously, some steps were void, since the recognition/prediction indicated a *no-change*. If we take only non-void steps into account then gains for the best models (no. 2 for recognition, no. 6 for prediction) are respectively: 1.8 pips per step (44.6% of steps were void) and 0.81 pips per step (4% of steps were void).

The influence of parameters on the results varied, but generally one could give the following comments. Sequences of shorter lengths  $T$  led to better results in the recognition approach. The increased discretization  $\{-2, -1, 0, 1, 2\}$  with fixed  $(w_1, w_2) = (0.25, 0.5)$  improved the results on average in both approaches. It was difficult to point out the influence of  $h$  parameter in the prediction approach.



**Fig. 3.** Accumulated gains along  $10^3$  steps of time during simulation: (a) trajectory recognition model no. 2, (b) trajectory prediction model no. 6. Calmar ratio  $ag(T)/|\min \Delta|$ : (a) 7.62, (b) 3.82. Calmar measure: (a) 0.11, (b) = 0.063.

In the fig. 3 we show plots of accumulated gains in simulations for the two best models discovered in each approach. The Calmar ratio (the ratio of accumulated gain at the last time step to the largest drop along time) for these models was calculated:  $ag(T)/|\min \Delta|$ , where  $|\min \Delta| = |\min_{t_1 < t_2} (ag(t_2) - ag(t_1))|$ , and was equal respectively 7.62 and 3.82. As regards the Sharpe measure (the ratio of mean gain to gain standard deviation) it was equal respectively: 0.11 and 0.063.

One aspect not considered in the paper is transaction costs related to opening and closing a position on broker platforms. In general, these are varying costs of order  $2 \div 5$  pips and can be verified *only* in practical testing<sup>13</sup>. As one can see, such costs surpass potential gains of presented models<sup>14</sup>.

<sup>12</sup> Note that for given data, at maximum an 11.1 pips per step gain was possible.

<sup>13</sup> They are not much compliant with broker's declaration his platform is the cheapest.

<sup>14</sup> Costs can be lowered if one does not count void steps or if one does not always close an open position (short or long) when the next context is predicted to be the same.

Another important aspect is the non-stationarity of the time series. Despite the stationarity assumption, which can be false, it seems reasonable to consider the stationarity as a *moving* phenomenon. So, one might look at a non-stationary process as *locally* stationary. Then, the training procedure could take into account only last e.g. several hundred candles and would be of *moving* nature with a retraining made time after time. Hopefully, such approach could improve the results and the authors plan to investigate it.

In authors' opinion the application of Markov chains for short-term recognitions/predictions is a good direction of research for financial mathematics.

## References

1. Brock, W., Lakonishok, J., LeBaron, B.: Simple technical trading rules and stochastic properties of stock returns. *Journal of Finance* 47, 1731–1764 (1992)
2. Elder, A.: *Come Into My Trading Room*. Wiley, Trading (2002)
3. Fama, E.: Efficient capital markets. *Journal of Financial Economics* 11, 1575–1617 (1991)
4. Gareth, B.: *Trading and Investing in the Forex Markets Using Chart Techniques*. John Wiley, Chichester (2009)
5. Gencay, R.: Linear, non-linear and essential foreign exchange rate prediction with simple technical trading rules. *Journal of International Economics* 47, 91–107 (1999)
6. Musiol, G., Mulig, H., Bronsztejn, I.N., Siemiendajew, K.A.: *Nowoczesne Kompendium Matematyki*. Wydawnictwo Naukowe PWN, Warszawa, Polska (2004)
7. Kłęsk, P.: O Ukrytych Modelach Markowa i ich zastosowaniu do rozpoznawania ciągów znaków pisma odręcznego. *Metody Informatyki Stosowanej* 13(1/2008), 175–190 (2008); *Kwartalnik Komisji Informatyki Polskiej Akademii Nauk Oddział w Gdańsku*
8. Krustinger, J.: *Systemy transakcyjne. sekrety mistrzów*. WIG Press (1999); Title translation: *Trading systems. Secrets of the Masters*
9. Rabiner, L.R.: A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, vol. 77, pp. 257–286. IEEE, Los Alamitos (1989)
10. Ross, J.: *Trading the Ross Hook*. John Wiley, Chichester (2001)
11. Satchwell, C.: *Pattern Recognition and Trading Decisions*. McGraw Hill, New York (2005)
12. Schwager, J.D.: *Analiza techniczna rynków terminowych*. WIG Press (2002); Title translation: *Technical analysis of forward markets*
13. Tian, G.G., Wan, G.H., Guo, M.: Market efficiency and the returns to simple technical trading rules: New evidence from u.s. equity markets and chinese equity markets. *Asia-Pacific Financial Markets* 9, 241–288 (2002)
14. Wiliński, A.: *GMDH — metoda grupowania argumentów w zadaniach zautomatyzowanej predykcji zachowań rynków finansowych*. Instytut Badań Systemowych PAN (2009)

# Do We Need Whatever More Than k-NN?

Mirosław Kordos<sup>1</sup>, Marcin Blachnik<sup>2</sup>, and Dawid Strzempa<sup>1</sup>

<sup>1</sup> University of Bielsko-Biała, Department of Mathematics and Computer Science,  
Bielsko-Biała, Willowa 2, Poland  
mkordos@ath.bielsko.pl

<sup>2</sup> Silesian University of Technology, Electrotechnology Department,  
Katowice, Krasinskiego 8, Poland  
mblachnik@polsl.pl

**Abstract.** Many sophisticated classification algorithms have been proposed. However, there is no clear methodology of comparing the results among different methods. According to our experiments on the popular datasets, k-NN with properly tuned parameters performs on average best. Tuning the parameters include the proper k, proper distance measure and proper weighing functions. k-NN has a zero training time and the test time can be significantly reduced by prior reference vector selection, which needs to be done only once or by applying advanced nearest neighbor search strategies (like KDtree algorithm). Thus we propose that instead of comparing new algorithms with an author's choice of old ones (which may be especially selected in favour of his method), the new method would be rather compared first with properly tuned k-NN as a gold standard. And based on the comparison the author of the new method would have to answer the question: "Do we really need this method since we already have k-NN?"

## 1 Introduction

K-nearest-neighbor classification was developed from the need to perform discriminant analysis when reliable parametric estimates of probability densities are unknown or difficult to determine. In an unpublished US Air Force School of Aviation Medicine report in 1951, Fix and Hodges introduced a non-parametric method for pattern classification that has since become known the k-nearest neighbor (k-NN) [1].

Since k-NN classification is one of the most fundamental and simple classification methods, it should be one of the first choices for a classification study when there is little or no prior knowledge about the distribution of the data [13]. Our paper investigates if k-NN should be not only the first but in most cases also the only classification algorithm applied to a given problem.

An approach to comparison of the accuracy of various classification algorithms depends on the author of the given paper. In general many authors try to show that their method is the best by comparing it with other methods that performs poorer on the given data. Frequently one of the compared algorithms is k-NN, however with  $k=1$ , rarely  $k=3$ . As our experiments shown, a good choice for k is rather any number between 10 and 20 than between 1 (or even 3), what changes the k-NN performance dramatically. The fact is rarely mentioned by the authors. There can be two reasons of that: first, they are unaware of this, because they do not perform the experiments themselves, but rather

take the results from other sources and second, they are aware of this, but deliberately use 1-NN (sometimes 3-NN) in order to do better in comparison with k-NN. A proper selection of  $k$  is most important. The weighing schemes mentioned below also improve classification results but in a lesser degree than setting a reasonable  $k$  value.

The next issue is that the vectors that lie closer to the test vector should be paid more attention to by applying rather a weighted k-NN than the raw form of k-NN. The simplest weighting system is to make the weights inversely proportional to the distance between the given point and the test point. However, other schemes may be used as well.

And finally we should take into account that different features have different prediction ability measured e.g. by feature ranking. Some classifications, as neural networks or decision trees have already in various ways embedded the feature weighting mechanism. Since the raw k-NN does not, it is advised to add the weighing scheme while determining the class of the test vector.

Another great impact on the accuracy of kNN classifier has appropriate reference vector selection. In many real problem storing whole data matrix is difficult, and considering distance calculation which has  $O(n^2)$  complexity and large memory requirements may radically restrict kNN's usability, unless some means are undertaken as discussed in the following sections.

The k-NN algorithm is really much simpler than many other methods and on a whole spectrum of datasets performs very well in comparison to them. Thus, we propose to establish a gold standard of comparing any new classification algorithm to k-NN with optimally tuned parameters. To enable this task in an easy way we are currently creating a web page at [www.kordos.com/knn](http://www.kordos.com/knn), where the user can submit any data set for classification and compare the result of their algorithm to that of (almost) optimally tuned k-NN.

## 2 k-Nearest Neighbors Algorithm

**The Basis of k-NN.** In k-NN a vector is classified by a majority voting of its neighbors, and assigned to the class most common among its  $k$  nearest neighbors. For example in 5-NN if two nearest neighbors belong to class A and three to class B, the vector is assigned to class B. If  $k = 1$ , then the vector is simply assigned to the class of its nearest neighbor. The same method can be used for regression, by simply assigning the output value of a vector to be the average of the values of its  $k$  nearest neighbors.

Both in classification and regression it can be useful to weight the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. K-NN is a lazy algorithm; it has a zero training time, but all the calculations are performed at the predictions time. For big datasets that can be long if the distance is to be calculated to each training vector. However, the simple solution is to use tree based search strategies instead of typically done linear search. The most common choices are KDTree or ballTree algorithms. Another solution, however not so smart as the previous one, is to cluster the data first (what needs to be done only once) and then perform k-NN twice. Once on the cluster centers to find  $k$  nearest clusters and the second one only on the vectors within the  $k$  nearest clusters (the second run is needed only if vectors in the  $k$  nearest clusters belong to different classes).



**Distance Matrices.** The distance between two vectors can be calculated using different distance measures. Most frequently Minkovski distance is used:

$$d_{x,y} = \sqrt[\lambda]{\sum_{i=1}^f (|x_i - y_i|)^\lambda} \quad (1)$$

Where  $d_{x,y}$  is the distance between the vector  $x$  and  $y$  in  $f$ -dimensional feature space. For the exponent  $\lambda = 1$ , the above becomes the Manhattan norm, for  $\lambda = 2$  Euclidean norm and for  $\lambda = \infty$  Chebyshev norm. We performed some experiments to determine the distance measure influence on k-NN classification accuracy.

**Weighting.** We discuss two possible weighting schemes: one based on the distance and one on the feature ranking. In the case of the distance-based weighting, the class of a given vector  $y$  is determined by the greatest sum  $W_c$  of weighed distances of  $k$  nearest neighbors from class  $c$ . In the simplest case the weight can be proportional to the inverse of the distance between the two vectors (eq. 2-left). In the experiments, we however used the exponential weighting scheme (eq. 2-right).

$$W_c = \sum_{i=1}^n d_{x_i,y}^{-1} \quad W_c = \sum_{i=1}^n 2^{-0.2d_{x_i,y}} \quad (2)$$

In the weighting by feature ranking, first a correlation between each single feature and class (or output value in case of regression) is calculated and then the distance in each feature dimension is multiplied by this correlation (eq. 3a).

$$d_{x,y} = \sqrt[\lambda]{\sum_{i=1}^f (|c_i(x_i - y_i)|)^\lambda} \quad (3)$$

not the only possible weighting by feature, but this is the simplest one and that one we used in the experiments. The aim of that weighting is to limit the noise that can be contained in the less related features.

As the experiments showed using the distance-based weighing improves the results and using also the feature-based weighting improves the results even more. However, both improvements are relatively small.

**Limitations of k-NN.** The limitations of k-NN are results of the shape of decision borders produced by the algorithm, that is turning the corners and a tendency to suppress a narrow passages, as shown in the figures below. That is however not a frequent case in a real world datasets. To prevent these effects one can increase the exponent in the Minkovski distance measure, but that however can lead to an overall decrease in accuracy (see the experiment section), because only the distance in one feature direction will have a practical influence on the decision making.

Another option is to utilize Mahalanobis distance function, or other data-dependent matrices like probabilistic measures (VDM or MRM) that depends on data distribution. These last two metrics can be also utilized in case of nominal attributes, or in non homogeneous datasets that consists of diferent feature types. [22]

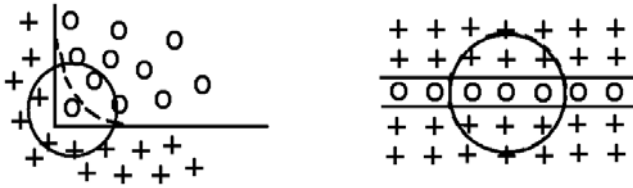


Fig. 1. Limitations of k-NN algorithm

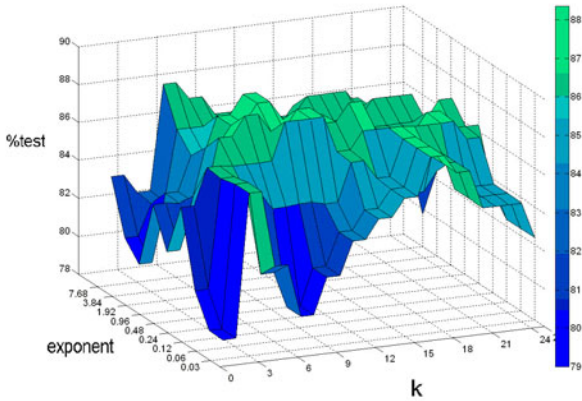


Fig. 2. Influence of the number of neighbor (k) and the power in the Minkovski distance metrics on the classification accuracy of unweighted k-NN in 10-fold crossvalidation on the Appendicitis dataset

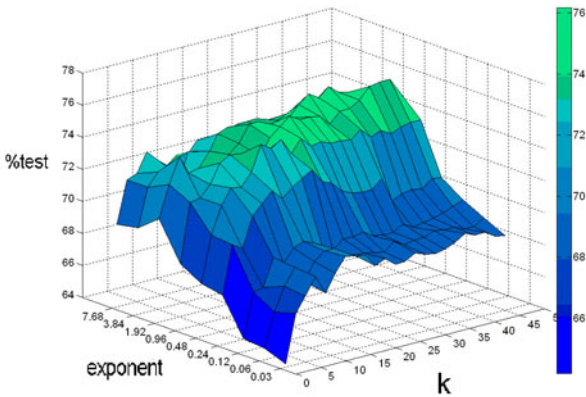


Fig. 3. Influence of the number of neighbor (k) and the power in the Minkovski distance metrics on the classification accuracy of unweighted k-NN in 10-fold crossvalidation on the Diabetes dataset

**Table 1.** Classification accuracy on the Wisconsin Breast Cancer (WBC) dataset

| algorithm                               | test | source         |
|-----------------------------------------|------|----------------|
| 5-NN                                    | 97.4 | this work, [1] |
| 5-NN weighted by distance               | 97.3 | this work, [1] |
| 5-NN weighted by distance and attribute | 97.6 | this work      |
| C4.5                                    | 94.7 | [7]            |
| LVQ                                     | 96.6 | [17]           |
| FDA                                     | 96.7 | [17]           |
| SSV                                     | 96.3 | [12]           |
| CART                                    | 93.5 | [17]           |
| LDA                                     | 96.0 | [17]           |
| MLP+BP                                  | 96.7 | [17]           |
| SVM lin, opt C                          | 96.7 | [17]           |
| incNet (3000 epochs, 40 neurons)        | 97.1 | [13]           |
| SMLP                                    | 97.1 | [11]           |

**Table 2.** Classification accuracy on the Appendicitis dataset

| algorithm                                | test | source         |
|------------------------------------------|------|----------------|
| 11-NN                                    | 88.4 | this work, [1] |
| 11-NN weighted by distance               | 88.9 | this work, [1] |
| 11-NN weighted by distance and attribute | 89.4 | this work      |
| SMLP                                     | 88.2 | [20]           |
| Nad' ve Bayes                            | 83.0 | [17]           |
| SVM                                      | 88.1 | [18]           |
| SSV                                      | 87.8 | [12]           |
| CART                                     | 84.9 | [18]           |
| SMLP                                     | 88.2 | [11]           |
| FSM                                      | 87.6 | [14]           |
| incNet (1100 epochs, 30 neurons)         | 90.1 | [13]           |
| MLP+BP                                   | 85.8 | [19]           |

### 3 Experimental Results

We performed experiments with k-NN on the popular benchmark datasets from the UCI Machine Learning repository [3]: Iris, Appendicitis, WBC, Diabetes and Ionosphere. We compared the following algorithms: standard k-NN, k-NN weighted by distance [4], k-NN weighted by distance and feature, MLP [5], CART [6], C4.5 [7], RBF [5], LVQ [8], LDA [9], FDA [10], SMLP [11], SSV [12], IncNet [13], FSM [14], SVM [15], Naive Bayes [16]. There are algorithms that on a particular dataset performs slightly better than k-NN weighted by distance and feature, however first, the difference is very little and second there is no other algorithms than on average performs as well as k-NN. We performed the experiments only with k-NN and presented the other algorithm results as reported either by their authors or by the comparison projects [17][18][19].

**Table 3.** Classification accuracy on the Diabetes dataset

| algorithm                                | test | source          |
|------------------------------------------|------|-----------------|
| 23-NN                                    | 76.2 | this work, [11] |
| 23-NN weighted by distance               | 76.6 | this work, [11] |
| 23-NN weighted by distance and attribute | 77.0 | this work       |
| SVM, Gauss, C, sigma opt                 | 77.4 | [20]            |
| RBF                                      | 75.7 | [17]            |
| LVQ                                      | 76.0 | [17]            |
| CART                                     | 74.7 | [17]            |
| C4.5                                     | 73.0 | [17]            |
| SSV                                      | 74.8 | [12]            |
| MLP+BP                                   | 75.2 | [19]            |

**Table 4.** Classification accuracy on the Iris dataset

| algorithm                                | test | source          |
|------------------------------------------|------|-----------------|
| 13-NN                                    | 96.7 | this work, [11] |
| 13-NN weighted by distance               | 96.7 | this work, [11] |
| 13-NN weighted by distance and attribute | 96.8 | this work       |
| Naive Bayes                              | 97.3 | [17]            |
| NEFCLASS                                 | 96.7 | [18]            |
| trainable fuzzy system                   | 96.0 | [12]            |

**Table 5.** Classification accuracy on the Ionosphere dataset

| algorithm                                | test | source          |
|------------------------------------------|------|-----------------|
| 13-NN                                    | 96.7 | this work, [11] |
| 13-NN weighted by distance               | 96.8 | this work, [11] |
| 13-NN weighted by distance and attribute | 97.1 | this work       |
| IB3                                      | 96.7 | [21]            |
| C4.5                                     | 94.9 | [17]            |
| SVM                                      | 93.2 | [17]            |
| CART                                     | 88.9 | [17]            |
| FSM                                      | 92.8 | [14]            |
| MLP+BP                                   | 96.0 | [18]            |

## 4 Conclusions

As the experimental results shows in most cases the simple k-NN is not only sufficient but also one of the best and most universal algorithm.

So do we need whatever more than k-NN? The obvious answer is yes, because as no free lunch theorem says there is no best method that would beat all the others, but as our results proved, when k-NN is correctly tuned its accuracy can be comprehensive to other even much more sophisticated algorithms. Of course k-NN is not suitable for all possible problems, specially for very large datasets and real time prediction processes

it requires some preprocessing of the data (see section 2) to reduce its computational complexity and memory requirements.

Another problem is the flexibility of k-NN. It may easily overfit the data, so tuning k-NN classifier requires good accuracy prediction methodology that would overcome such limitation like bootstrapping or cross validation. We believe that the conclusion that derives from our experiments is that the k-NN algorithm should become a reference model to all other algorithms developed at any research groups. We have begun creating a simple web page at [www.kordos.com/knn](http://www.kordos.com/knn) that allows fine tuning of the k-NN classifier and performing experiments on any uploaded data. Results of the experiments can then be used for comparison with other methods.

## References

1. Fix, E., Hodges, J.L.: Discriminatory analysis, nonparametric discrimination: Consistency properties. USAF School of Aviation Medicine, Randolph Field, Texas (1951)
2. [http://www.scholarpedia.org/article/K-nearest\\_neighbor](http://www.scholarpedia.org/article/K-nearest_neighbor)
3. UCI Machine Learning Repository, <http://archive.ics.uci.edu/ml>
4. Strzempa, D.: Internet System for Data Classification (in Polish), MSc Thesis, The Silesian University of Technology, Katowice (2008), <http://www.ath.bielsko.pl/~mkordos/mgr/ds2008.pdf>
5. Duda, R.O., et al.: Pattern Classification. Wiley, New York (2001)
6. Breiman, L., et al.: Classification and Regression Trees. Wadsworth, CA (1984)
7. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Francisco (1993)
8. Kohonen, T.: Statistical Pattern Recognition Revisited. Elsevier, Amsterdam (1990)
9. Schalkoff, R.: Pattern Recognition: Statistical, Structural and Neural Approaches. Wiley, Chichester (1992)
10. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Wiley, Chichester (1950)
11. Kordos, M.: Search-based Algorithms for Multilayer Perceptrons, PhD Thesis, The Silesian University of Technology, Gliwice (2005), <http://www.fizyka.umk.pl/~kordos/pdf/MKordos-PhD.pdf>
12. Grabczewski, K.: Application of SSV Criterion for generating classification rules, PhD Thesis, Nicolaus Copernicus University, Torun (2003) (in Polish)
13. Jankowski, N.: Ontogenic Neural Networks for Medical Data Classification, PhD Thesis, Nicolaus Copernicus University, Torun (1999) (in Polish)
14. Adamczak, R.: Neural networks application for experimental data classification, PhD Thesis, Nicolaus Copernicus University, Torun (1999, 2001) (in Polish)
15. Scholkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond. MIT Press, Cambridge (2001)
16. Michie, D., Spiegelhalter, D.J., Taylor, C.C.: Machine Learning, neural and statistical classification. Elis Horwood, London (1994)
17. Ster, B., Dobnikar, A.: Neural Networks in medical diagnosis: Comparison with other methods. In: EANN 1996, pp. 427–430 (1996)
18. Zarndt, F.: A comprehensive case study: An examination of machine learning and connectionists algorithms, MSc Thesis, Department of Computer Science, Brigham Young University (1995)

19. Weiss, S.M., Kapouleas, I.: An empirical comparison of pattern Recognition, neural nets and machine learning classification methods. In: Reading in Machine Learning. Morgan Kaufman Publ., CA (1990)
20. [http://www.fgs.pl/business\\_intelligence/products/ghostminer/product\\_overview](http://www.fgs.pl/business_intelligence/products/ghostminer/product_overview)
21. <http://www.is.umk.pl/projects/datasets.html>
22. Blachnik, M., Duch, W., Wieczorek, T.: Probabilistic distance measures for prototype-based rules. In: Proc. of the 12th Int. Conference on Neural Information Processing (ICONIP 2005), Taipei, Taiwan, pp. 445–450 (2005)

# Pattern Recognition with Linearly Structured Labels Using Recursive Kernel Estimator

Adam Krzyżak<sup>1</sup> and Ewaryst Rafajłowicz<sup>2</sup>

<sup>1</sup> Department of Computer Science and Software Engineering, Concordia University, Montreal, Canada and on sabbatical leave with the Department of Electrical Engineering, ZUT, Szczecin, Poland

`krzyzak@cs.concordia.ca`

<sup>2</sup> Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50 370 Wrocław, Poland

`ewaryst.rafajlowicz@pwr.wroc.pl`

**Abstract.** We consider pattern recognition problem when classes and their labels are linearly structured (or ordered). We propose the loss function based on the squared differences between the true and the predicted class labels. The optimal Bayes classifier is derived and then estimated by the recursive kernel estimator. Its consistency is established theoretically. Its RBF-like realization of the classifier is also proposed together with a recursive learning algorithm, which is well suited for on-line applications. The proposed approach was tested in real life example involving classification of moving vehicles.

## 1 Introduction

We consider pattern recognition problem in the bayesian setting (see, e.g., [1] or [3]) with the loss function  $L(i, j)$ , which assigns loss  $L(i, j)$  to a pattern from  $i$ -th class classified to  $j$ -th class, where  $i, j$  are class labels from a finite set  $\mathcal{I} \stackrel{def}{=} \{1, 2, \dots, I\}$ . In the main stream of the pattern recognition theory the so called 0-1 loss function is considered, i.e.,  $L(i, j) = 0$ , if  $i = j$  and  $L(i, j) = 1$  otherwise. Selecting 0-1 loss function immediately implies that class labels are unordered in the sense that we can attach them to classes in an arbitrary way, without changing the problem. In fact, we must not even use numbers as labels and one can select, e.g., words attached to classes.

Unlike the main stream of research, we treat labels as linearly structured (ordered) in the sense that classes, which are – in some sense closer – to each other have closer labels. For example, if our task is to classify moving vehicles to the classes: "car", "van", "truck", we attach labels 1, 2, 3, respectively, according to increasing size and weight of vehicles to be classified. By selecting an appropriate loss function, we want to reflect the intuitive feeling that when a small car is classified as truck then our loss should be higher than in the case when it is classified as van. Similarly, in quality control of goods it is reasonable to attach less loss when a product from the first class is classified to the second class than

when it is classified as useless. These intuitions can be formalized by selecting  $L(i, j) = (i - j)^2$  or its generalizations (see Section 2).

Similar considerations can lead to selection of the absolute deviation loss  $L(i, j) = |i - j|$  (see [8], [9]) or to bivariate labels [10]. In [11] we have considered  $L(i, j) = (i - j)^2$  but in a non-recursive setting. The recursive estimator proposed here seems to be better suited both for learning from a stream of data as well as in the recognition phase, while the version proposed in [11] is also appropriate in the recognition phase but at the learning stage one has to store a whole learning sequence. This can be rather expensive when our learning data are extracted from large data sets of video sequence.

## 2 Quadratic Loss with Weights

For the reasons explained in the Introduction, we adopt loss function

$$L(i, j) = w(i) (i - j)^2, \quad i, j \in \mathcal{I}, \tag{1}$$

where  $w(i) > 0$ ,  $i \in \mathcal{I}$  is a sequence of weights. Typical choices of weights that we have in mind are the following:

1.  $w(i) = i^\alpha$ ,  $\alpha > 0$  – which should be used when we would to give higher priority to the correct classification to classes with larger labels,
2.  $w(i) = \frac{1}{i^\alpha}$ ,  $\alpha > 0$  – which emphasizes correct classification to classes with smaller labels.

Clearly, the choice  $w(i) \equiv 1$  is of our main interest.

The rest of ingredients, which are necessary for problem statement are defined below.

- 1) Let  $X \in R^d$  be a random vector, representing a pattern, which is drawn from one of the classes in  $\mathcal{I}$ .
- 2) Pair  $(X, i)$  is a random vector representing a pattern and its correct classification  $i$ , which is unknown for a new pattern  $X$  to be classified.
- 3) For theoretical considerations we assume that probability distribution of  $(X, i)$  is known.
- 4) When an empirical versions of classifiers are considered, we assume that we have a learning sequence  $(X^{(k)}, i^{(k)})$ ,  $k = 1, 2, \dots, n$  of observed patterns  $X_k \in R^d$  and their correct classifications  $i^{(k)} \in \{1, 2, \dots, I\}$ . We assume that  $(X^{(k)}, i^{(k)})$ 's are independent, identically distributed random vectors with the same probability distribution as  $(X, i)$ .
- 5) Denote by  $0 \leq q(i) \leq 1$ , a priori probability that  $X$  comes from  $i$ -th class,  $i = 1, 2, \dots, I$ ,  $\sum_{i=1}^I q(i) = 1$ .

The goal is to find (or to approximate from a learning sequence) a decision function  $\Psi(X)$ , which specifies a label of the class for  $X$  and such that it minimizes the expected loss given by:

$$R(\Psi) = E_X \left[ \sum_{i=1}^I w(i) (i - \Psi(X))^2 P(i|X) \right], \tag{2}$$



where  $E_X$  denotes the expectation w.r.t.  $X$ , while  $P(i|X)$  is the a posteriori probability that observed pattern  $X$  comes from  $i$ -th class. In other words,  $P(i|X = x)$  is the conditional probability of the event that label  $i$  is the correct classification of a given pattern  $X = x$ . Our aim is to minimize the risk  $R(\Psi)$ , provided that the minimizer  $\Psi^*(x)$ , say, is a measurable function.

Note that minimizing (2) we have to ensure that  $\Psi^*(x)$  is a positive integer.  $\Psi^*(x)$  is called the Bayes classifier.

### 3 Optimal Classifier and Empirical Decision Rule

The result below characterizes the Bayes classifier.

**Theorem 1.** *If weights  $w(i)$  form a strictly increasing sequence, then the Bayes classifier has the form:*

$$\Psi^*(x) = \text{ROUND}(\tilde{\Psi}(x)), \tag{3}$$

where

$$\tilde{\Psi}(x) = \frac{\sum_{i=1}^I i w(i) P(i|X = x)}{\sum_{i=1}^I w(i) P(i|X = x)}. \tag{4}$$

**Proof.** To prove Theorem 1 let us note that in order to minimize  $R(\Psi)$  it suffices to minimize the conditional risk

$$r(\psi, x) \stackrel{\text{def}}{=} \sum_{i=1}^I w(i) (i - \psi)^2 P(i|X = x) \tag{5}$$

with respect to  $\psi$ , while  $x$  is treated as a parameter. According to the above statement, the optimal decision rule  $\Psi^*(x)$  is obtained as

$$\Psi^*(x) = \underset{\psi}{\text{arg min}} r(\psi, x) \tag{6}$$

for all  $x \in R^d$  in the range of  $X$ . Let us "forget" for a while that  $\psi$  should take values from  $\mathcal{I}$  and treat them as a real variable. Then, after differentiation of (5) w.r.t.  $\psi$  we obtain: a relaxed minimizer, denoted by  $\tilde{\Psi}(x)$ , of the form (4). Now, it remains to justify that rounding  $\tilde{\Psi}(x)$  to the nearest integer we obtain the optimal solution. To this end let us note that for nonnegative and strictly increasing weights  $w(i)$  function  $r(\psi, x)$  is a strictly convex function of  $\psi$  for every  $x$  (its second derivative w.r.t.  $\psi$  is positive). Consider two neighbors of  $\tilde{\Psi}(x)$  among integers, namely,  $\text{Ceiling}(\tilde{\Psi}(x))$  and  $\text{Floor}(\tilde{\Psi}(x))$ . The minimizer of (5) among integers must be the one of  $\text{Ceiling}(\tilde{\Psi}(x))$  and  $\text{Floor}(\tilde{\Psi}(x))$ , which is closer to  $\tilde{\Psi}(x)$ , since otherwise we obtain the contradiction with the strict convexity of  $r(\psi, x)$ .

Let us consider special cases.

**Constant weights.** If  $w(i) \equiv \text{const}$ , then

$$\tilde{\Psi}(x) = \sum_{i=1}^I i P(i|X = x). \tag{7}$$

This is the main case considered below and for constant weights we shall denote  $\tilde{\Psi}(x)$  by  $\eta(x)$ . Note also that

$$\eta(x) = \sum_{i=1}^I i P(i|X = x) = E\{Y|X = x\}. \tag{8}$$

**Linear weights.** If  $w(i) = i$ , then

$$\tilde{\Psi}(x) = \frac{\sum_{i=1}^I i^2 P(i|X = x)}{\sum_{i=1}^I i P(i|X = x)}. \tag{9}$$

**Decreasing weights.** Selecting  $w(i) = 1/i$ , we obtain

$$\tilde{\Psi}(x) = \left[ \sum_{i=1}^I \frac{1}{i} P(i|X = x) \right]^{-1}. \tag{10}$$

In this case  $\tilde{\Psi}$  resembles the harmonic mean. Note however that Theorem [11](#) does not cover this case, since  $w(i)$  are not increasing. In fact, it will be clear from the proof presented below that [\(10\)](#) is a suboptimal solution.

We can estimate classification rule  $\Psi^*(x)$  from [\(8\)](#) as follows

$$\hat{\Psi}(x) = \text{ROUND} \left[ \frac{\sum_{k=1}^n i^{(k)} K \left( \frac{\|x - X^{(k)}\|}{h(i)} \right)}{\sum_{k=1}^n K \left( \frac{\|x - X^{(k)}\|}{h(i)} \right)} \right] = \eta_n(x), \tag{11}$$

where  $K(t) \geq 0$ ,  $t \in R$  is a non-negative kernel and  $h(i) > 0$  is a sequence of numbers, which play the role of smoothing parameters. More restrictions on  $h(i)$  will be imposed below. Let us note that [\(11\)](#) can be equivalently rewritten in the following recursive form:

$$\eta_n(x) = \text{ROUND} \left[ \frac{L_n(x)}{M_n(x)} \right], \tag{12}$$

where  $L_0(x) \equiv 0$ ,  $M_0(x) \equiv 0$  and, by definition,  $\eta_0(x) \equiv 0$ , while for  $n = 1, 2, \dots$

$$L_n(x) = L_{n-1}(x) + i^{(n)} K \left( \frac{\|x - X^{(n)}\|}{h(n)} \right) \tag{13}$$

$$M_n(x) = M_{n-1}(x) + K \left( \frac{\|x - X^{(n)}\|}{h(n)} \right). \tag{14}$$

We will now state and prove the convergence result for the classification rule [\(11\)](#).

**Theorem 2.** *Let kernel  $K$  satisfy the condition*

$$\alpha H(\|x\|) \leq K(x) \leq \beta H(\|x\|), \quad x \in \mathcal{R}^d$$

for some  $0 < \alpha < \beta < \infty$  and nonincreasing  $H : \mathcal{R}_+ \rightarrow \mathcal{R}_+$  with  $H(+0) > 0$ ,  $\|\cdot\|$  is a norm in  $\mathcal{R}^d$  and assume that

$$h_n \downarrow 0, \quad \sum_{i=1}^n h_i^d \rightarrow \infty \quad \text{as } n \rightarrow \infty. \tag{15}$$

Then rule [\(11\)](#) is universally consistent, i.e.,

$$R(\hat{\Psi}) - R(\Psi^*) \rightarrow 0 \quad \text{as } n \rightarrow \infty$$

for all distributions of  $X$ .

**Proof.** We have

$$\begin{aligned} R(\hat{\Psi}) - R(\Psi^*) &= E \sum_{i=1}^I \left[ (i - \hat{\Psi}(X))^2 - (i - \Psi^*(X))^2 P(i|X) \right] \\ &= \sum_{i=1}^I E[(\Psi^*(X) - \hat{\Psi}(X))(2i - \hat{\Psi}(X) - \Psi^*(X))P(i|X)] \\ &\leq 2I \sum_{i=1}^I E[|\Psi^*(X) - \hat{\Psi}(X)|P(i|X)] \\ &= 2I \sum_{i=1}^I E[|ROUND(\eta(X)) - ROUND(\eta_n(X))|P(i|X)]. \end{aligned}$$

Let  $c_i = i + 1/2$  be a center of the interval  $[i, i + 1)$ ,  $i = 1, \dots, I - 1$ . It is clear that  $R(\hat{\Psi}) - R(\Psi^*) = 0$  if  $\eta(X)$  and  $\eta_n(X)$  belong to the interval  $[i + 1/2, i + 3/2)$ , otherwise  $R(\hat{\Psi}) - R(\Psi^*) \neq 0$ . Choose arbitrary  $\epsilon > 0$ . We have

$$\begin{aligned} &2I \sum_{i=1}^I E[|ROUND(\eta(X)) - ROUND(\eta_n(X))|P(i|X)] \\ &= 2I \sum_{i=1}^I E \left[ |ROUND(\eta(X)) - ROUND(\eta_n(X))|P(i|X) \right. \\ &\quad \cdot \left. \left( I_{\{\eta(X) \in \cup_{i=1}^I S_{c_i, \delta}\}} + I_{\{\eta(X) \notin \cup_{i=1}^I S_{c_i, \delta}\}} \right) \right] \\ &\leq 2IP \left( \eta(X) \in \cup_{i=1}^I S_{c_i, \delta} \right) \\ &+ 2I \sum_{i=1}^I E \left[ |ROUND(\eta(X)) - ROUND(\eta_n(X))|P(i|X) I_{\{\eta(X) \notin \cup_{i=1}^I S_{c_i, \delta}\}} \right] \\ &\leq 2I \sum_{i=1}^I P(\eta(X) \in S_{c_i, \delta}) \\ &\quad + (2I)^2 \max_{1 \leq i \leq I} P(\eta(X) \notin S_{c_i, \delta}, |\eta(X) - \eta_n(X)| > \delta/2) \\ &\leq 2I^2 \max_{1 \leq i \leq I} \int_{S_{c_i, \delta}} \eta(x) \mu(dx) + 4I^2 P(|\eta(X) - \eta_n(X)| > \delta/2) \\ &\leq 2I^3 \max_{1 \leq i \leq I} \mu(S_{c_i, \delta}) + 4I^2 \frac{2}{\delta} E|\eta(X) - \eta_n(X)|, \end{aligned}$$

where  $\mu$  is the distribution of  $X$  and we have used Markov inequality in last line above. Now choose  $\delta$  such that  $2T^3 \sup_x \mu(S_{x,\delta}) < \epsilon$  (it is possible as  $\mu$  is a finite measure). To complete the proof it suffices to show that  $E|\eta(X) - \eta_n(X)| \rightarrow 0$  as  $n \rightarrow \infty$ , but the latter follows from Theorem 24.2 in [5].

## 4 Neural Network for Structured Decision Making with Application to Recognition from Video Streams

Estimator  $\eta_n(x)$  itself is not recursive, but (13) and (14) are sufficient for applications in processing intensive data streams, since they allow for updating  $L_n$  and  $M_n$  without storing long learning sequences.

This advantage can be boosted by reducing the number of kernels appearing in the sums in (11). Mimicking the ideas from radial basis functions (RBF) neural networks (see [1], [5], [6], [7], [12]) we approximate decision rule (11) as follows.

**Step 1.** From initial observations of a data stream select centers  $c_j \in R^d$ ,  $j = 1, 2, \dots, J$  as cluster points of  $X^{(k)}$ 's.

**Step 2.** Form two RBF like networks and their ratio as follows:

$$\tilde{L}_n(x) = \sum_{j=1}^J \alpha_j^{(n)} K\left(\frac{\|x - c_j\|}{h(n)}\right), \tag{16}$$

$$\tilde{M}_n(x) = \sum_{j=1}^J \beta_j^{(n)} K\left(\frac{\|x - c_j\|}{h(n)}\right). \tag{17}$$

$$\tilde{\eta}_n(x) = \text{ROUND} \left[ \tilde{L}_n(x) / \tilde{M}_n(x) \right]. \tag{18}$$

**Step 3.** Train RBF nets: set  $\alpha_j^{(0)} = 0$ ,  $\beta_j^{(n)} = 0$ , when new pair  $(X^{(n)}, i^{(n)})$  is available, find

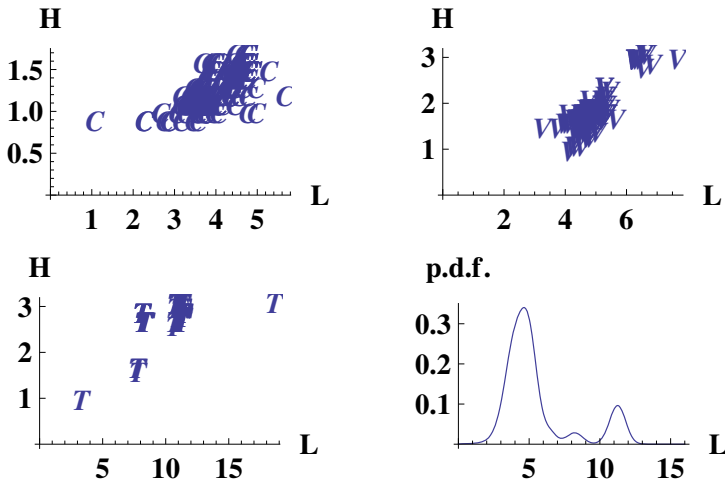
$$j^* = \text{arg} \min_{j=1,2,\dots,J} \|c_j - X^{(n)}\|.$$

Clearly,  $j^*$  depends on  $n$  but we skip this dependence in the notation. For  $n = 1, 2, \dots$  update weights as follows::

$$\alpha_{j^*}^{(n)} = \alpha_{j^*}^{(n-1)} + i^{(n)}, \quad \beta_{j^*}^{(n)} = \beta_{j^*}^{(n-1)} + 1. \tag{19}$$

In order to reveal in what sense the above RBF nets approximate (11), let us assume for a while that  $X^{(n)}$ 's can assume only finite number of values, which are equal to  $c_j$ ,  $j = 1, 2, \dots, J$ . Then, formulas (16), (17), (18) together with (19) provide exactly (11). In general  $X^{(n)}$ 's do not coincide exactly with  $c_j$ 's, but the approximation errors can be reduced by a proper choice of  $c_j$ 's. Additionally, the approximation errors are largely reduced by ROUND operation.

Our aim in this section is to illustrate the performance of the recognizer, proposed in Section 3, when applied to classification of moving vehicles from a video sequence. Only two features, namely vehicle length and height, were



**Fig. 1.** Learning sequence – lengths (L) and heights (H) (in meters) of moving vehicles: C – cars, V – vans, T – trucks and estimated p.d.f. of their lengths

selected. Vehicles are classified as: cars (C), vans (V), which include different kinds of vans, and trucks (T). A natural structure between these classes is C-V-T and we use  $L(i, j) = (i - j)^2$  loss function. The learning sequence is depicted in Fig. 1. As one can notice, the classes are heavily mixed, not only because larger cars and smaller vans are similar in size, but also because on-line extraction of information on length and height of a moving vehicle is distorted by many factors. For these reasons in many cases, which are shown in Fig. 1 markers "T" and "V" coincide. In our studies we have selected  $K(t)$  to be the Gaussian kernel. As  $h(n)$  we took  $c/\sqrt{(n+100)}$ , where  $c$  was selected as 5.5 for vehicle lengths and 4 for their heights. Available data were divided into the learning sequence of length 212, which was used for training, and the testing sequence of the same length. The quality of recognition was estimated as follows:  $Q = \left[212^{-1} \sum_{n=1}^{212} (i^{(n)} - j^{(n)})^2\right]^{1/2}$ , where  $i^{(n)}$  is the proper classification of  $n$ -th vehicle in the testing sequence, while  $j^{(n)}$  is the output of the classifier obtained when the length and the height of  $n$ -th vehicle was fed up as its input.  $Q = 0.039$  was obtained, which seems to be satisfactory, taking into account that classes are mixed to large extent. It is somewhat risky to compare performance a classifier, which is built using one optimality criterion with its performance according to another optimality criterion, but it can be of interest to count the percentage of misclassifications committed by our classifier. This number is 14.15%.

## 5 Concluding Remarks

The problem of pattern recognition with linearly structured labels is stated in nonparametric setting and solved by the recursive empirical classifier. Its

approximation by a double RBF neural network is also proposed, which is well suited for on-line processing intensive streams of data – video sequences in our example. This paper can also be viewed as our contribution to never-ending discussion on what is more difficult: nonparametric regression estimation or pattern recognition. One can easily notice similarities between our recursive recognizer and the nonparametric, recursive regression estimator. The main difference is that in the classical pattern recognition setting we assume that there are no errors in class membership labels. However, if we relax this assumption, e.g., in the spirit proposed in [4], then the differences between nonparametric pattern recognition and regression estimation are even less apparent.

**Acknowledgement.** This work was supported by the grant of the Polish Ministry of Science and Higher Education under a grant ranging from 2006 to 2009 and by the Natural Sciences and Engineering Research Council of Canada.

## References

1. Bishop, C.: *Neural Networks for Pattern Recognition*. Oxford Univ. Press, Cambridge (1995)
2. Devroye, L., Györfi, L.: *Nonparametric Density Estimation. The  $L_1$  View*. Wiley, New York (1985)
3. Devroye, L., Györfi, L., Lugosi, G.: *Probabilistic Theory of Pattern Recognition*. Springer, New York (1996)
4. Greblicki, W.: Learning to recognize patterns with a probabilistic teacher. *Pattern Recognition* 12, 159–164 (1980)
5. Györfi, L., Kohler, M., Krzyżak, A., Walk, H.: *A Distribution-Free Theory of Nonparametric Regression*. Springer, New York (2002)
6. Karayiannis, N.B., Randolph-Gips, M.M.: On the construction and training of reformulated radial basis function neural networks. *IEEE Trans. on Neural Networks* 14, 835–846 (2003)
7. Krzyżak, A., Skubalska-Rafajłowicz, E.: Combining space-filling curves and radial basis function networks. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) *ICAISC 2004. LNCS (LNAI)*, vol. 3070, pp. 229–234. Springer, Heidelberg (2004)
8. Rafajłowicz, E., Skubalska-Rafajłowicz, E.: MAD loss in pattern recognition and RBF learning. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008. LNCS (LNAI)*, vol. 5097, pp. 671–680. Springer, Heidelberg (2008)
9. Rafajłowicz, E.: Improving the efficiency of counting defects by learning RBF nets with MAD loss. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008. LNCS (LNAI)*, vol. 5097, pp. 146–153. Springer, Heidelberg (2008)
10. Rafajłowicz, E.: RBF nets in faults localization. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) *ICAISC 2006. LNCS (LNAI)*, vol. 4029, pp. 113–122. Springer, Heidelberg (2006)
11. Rafajłowicz, E., Krzyżak, A.: Pattern recognition with ordered labels. *Nonlinear Analysis* 71, 1437–1441 (2009)
12. Xu, L., Krzyżak, A., Yuille, A.: On radial basis function nets and kernel regression: statistical consistency, convergence, rates and receptive field size. *Neural Networks* 4, 609–628 (1994)

# Canonical Correlation Analysis for Multiview Semisupervised Feature Extraction

Olcay Kursun<sup>1</sup> and Ethem Alpaydin<sup>2</sup>

<sup>1</sup> Department of Computer Engineering, Istanbul University, 34320, Avcilar, Istanbul, Turkey  
okursun@istanbul.edu.tr

<sup>2</sup> Department of Computer Engineering, Bogazici University, 34342, Bebek, Istanbul, Turkey

**Abstract.** Hotelling's Canonical Correlation Analysis (CCA) works with two sets of related variables, also called views, and its goal is to find their linear projections with maximal mutual correlation. CCA is most suitable for unsupervised feature extraction when given two views but it has been also long known that in supervised learning when there is only a single view of data given, the supervision signal (class-labels) can be given to CCA as the second view and CCA simply reduces to Fisher's Linear Discriminant Analysis (LDA). However, it is unclear how to use this equivalence for extracting features from multiview data in semisupervised setting (i.e. what modification to the CCA mechanism could incorporate the class-labels along with the two views of the data when labels of some samples are unknown). In this paper, a CCA-based method supplemented by the essence of LDA is proposed for semi-supervised feature extraction from multiview data.

**Keywords:** Semisupervised Learning; Feature Extraction; Multiview Learning; LDA; CCA.

## 1 Introduction

Fisher's Linear Discriminant Analysis (LDA) [1] is one of the most popular linear dimensionality reduction methods; it seeks to find discriminatory projections of the data (i.e. those, which maximize the between class scatter and minimize the within class scatter). Whereas, Hotelling's Canonical Correlation Analysis (CCA) [2] works with two sets of related variables and its goal is to find maximally correlated linear projections of the two sets of variables. While LDA works completely in supervised setting (e.g. computationally, it needs to compute the within and between-class scatter matrices), CCA works completely in unsupervised manner (i.e. it ignores the class-labels and looks for correlated functions between the two views of data samples). Finding such correlated functions of the two views of the same phenomenon by discarding the representation-specific details (noise) is expected to reveal the underlying hidden yet influential semantic factors responsible for the correlation [3]. In this work, we extend the CCA setup so that it can take into account the class-label information into account as well. There are various ways of extending CCA to work

with more than two views [4]; however, considering the class-label information as a third view is not directly applicable in the semisupervised setting.

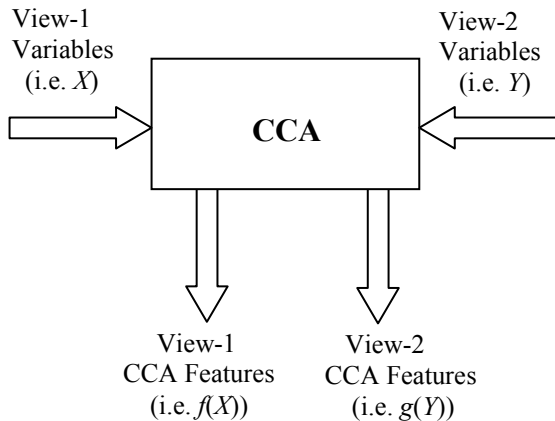
We propose to accommodate the class-labels in the CCA setup in a rather indirect way, through the class centers of the other view. Thus, if all the samples were labelled, this setup reduces to the classical samples versus class-labels setup, which has been long known to be equivalent to LDA with the slight change of representation for the class-labels by representing them using the mean of the samples of that class in the other view rather than a kind of 1-of- $C$  coding [5]. On the other hand, if all the samples were unlabelled this setup is the plain CCA itself. However, when there are both labelled and unlabelled samples, our method extracts CCA-like features with preference for LDA-like discriminatory ones.

## 2 Canonical Correlation Analysis (CCA)

Canonical Correlation Analysis (CCA) is introduced by Hotelling (1936) to describe the linear relations between two multidimensional (or two sets of) variables as the problem of finding basis vectors for each set such that the projections of the two variables on their respective basis vectors are maximally correlated (Figure 1). These two sets of variables, for example, may correspond to different views of the same semantic object (e.g. audio versus video of a person speaking, two cameras viewing the same object as in binocular vision, text versus links or images in webpages, etc). Let  $u$ -dimensional  $X$  and  $v$ -dimensional  $Y$  denote corresponding two sets of real-valued random variables (i.e.,  $X \in \mathbb{R}^u$  and  $Y \in \mathbb{R}^v$ ), the canonical correlation is defined as:

$$\rho(X;Y) = \sup_{f,g} \text{corr}(f^T X; g^T Y) \tag{1}$$

where,  $\text{corr}(X;Y)$  stands for Pearson's correlation coefficient.



**Fig. 1.** CCA-based Feature Extraction. Correlated features are extracted from the two views. The class-labels are not utilized.



The problem of finding the orthogonal projections that achieve the top correlations reduces to a generalized eigenproblem, where the projection  $f$  (and the projection  $g$  can be solved for similarly) corresponds to the top eigenvector of the following [6]:

$$\mathbf{C}_{XX}^{-1} \mathbf{C}_{XY} \mathbf{C}_{YY}^{-1} \mathbf{C}_{YX} f = \lambda_{CCA} f \tag{2}$$

and

$$\rho(X;Y) = \sqrt{\lambda_{CCA}}, \tag{3}$$

where

$$\mathbf{C}(X, Y) = \mathbf{E} \left\{ \begin{pmatrix} X \\ Y \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix}^T \right\} = \begin{bmatrix} \mathbf{C}_{XX} & \mathbf{C}_{XY} \\ \mathbf{C}_{YX} & \mathbf{C}_{YY} \end{bmatrix}. \tag{4}$$

### 3 Fisher Linear Discriminant Analysis (LDA)

Fisher Linear Discriminant Analysis (LDA) is a variance preserving approach with the goal of finding the optimal linear discriminant function [1, 7]. To utilize the categorical class label information in finding informative projections, LDA considers maximizing an objective function that involves the scatter properties of every class as well as the total scatter [7]. The objective function is designed to be maximized by a projection that maximizes the between class (or equivalently total scatter as in PCA) and minimize the within class scatter:

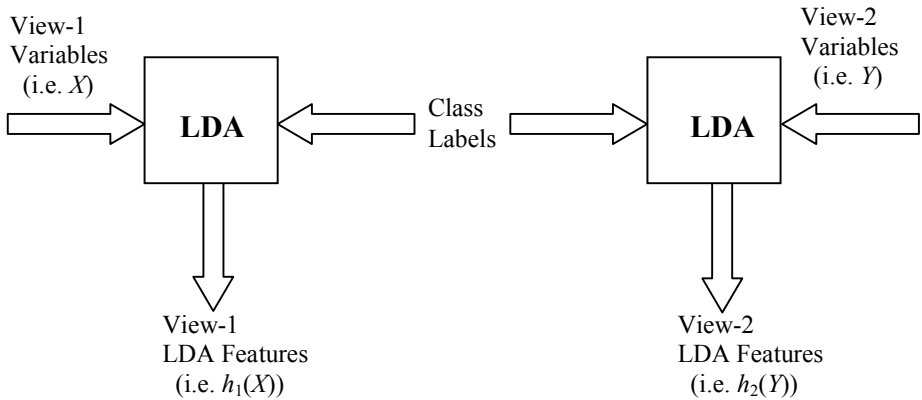
$$J = \sup \frac{h^T \mathbf{S}_B h}{h^T \mathbf{S}_W h}. \tag{5}$$

The optimization can be shown to be accomplished by computing the solution of the following generalized eigenproblem for the eigenvectors corresponding to the largest eigenvalues:

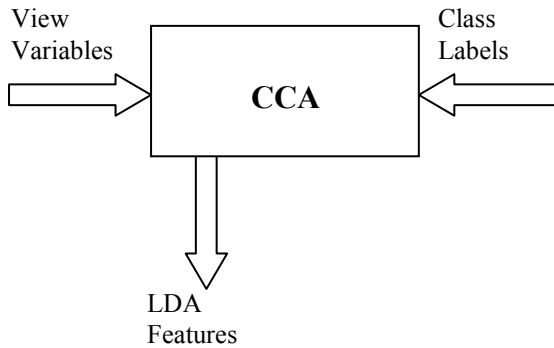
$$\mathbf{S}_B h = \lambda_{LDA} \mathbf{S}_W h. \tag{6}$$

LDA is originally designed for single view datasets, therefore when we have two views of the same objects (as the case for CCA), one straightforward approach would be to use the views separately as shown in Figure 2 and use both feature sets together for the subsequent classification task.

A direct connection between LDA and CCA can be obtained by showing that LDA is exactly what is accomplished by applying CCA between the set of all variables (of a view) and the corresponding class labels (0/1 for binary, 1-of- $C$  coding for multiclass classification). Searching for the maximal correlations between the variables and the class-labels via CCA (Figure 3), yields the LDA projections as solutions [5, 8, 9].



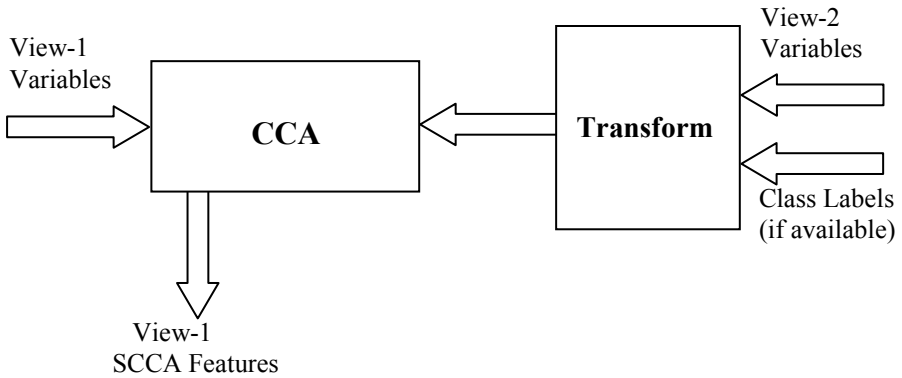
**Fig. 2.** LDA-based Feature Extraction. Features are extracted from the two views independently only for the labelled samples.



**Fig. 3.** CCA-based implementation of LDA. Correlated functions of the (single) view and the class-labels correspond to features extracted by LDA.

### 4 Proposed Architecture for Semisupervised CCA (SCCA)

One key observation to the method we propose is that in the architecture presented in Figure 3, the class-labels are not required to be hard labels in discrete format (e.g. class-0 and class-1 represented as 0 and 1 respectively). In fact, class-centers can be presented as class-labels [10]. Our proposal is simply to keep the other view when the class-label is absent; and otherwise, represent the class-labels by replacing the other view by the class-center of the samples in that other view. For example, to extract such SCCA features for View-1, we use View-1 variables versus View-2 variables in a regular CCA setup but we change View-2 feature vector to the respective class-center vector for the labelled samples (Figure 4). The procedure can be repeated in a similar fashion in order to extract SCCA features for View-2. Thus, SCCA features are expected to represent the view to view relations (akin to CCA) as well as view to class relations (akin to LDA) because for the unlabelled samples SCCA works like CCA and for the labelled samples it works like LDA.



**Fig. 4.** The proposed semisupervised version of CCA-based Feature Extraction (View-2 features can be extracted similarly). When dealing with a labelled data sample, View-2 variables are replaced by the View-2 prototype (class center) of the class of that sample.

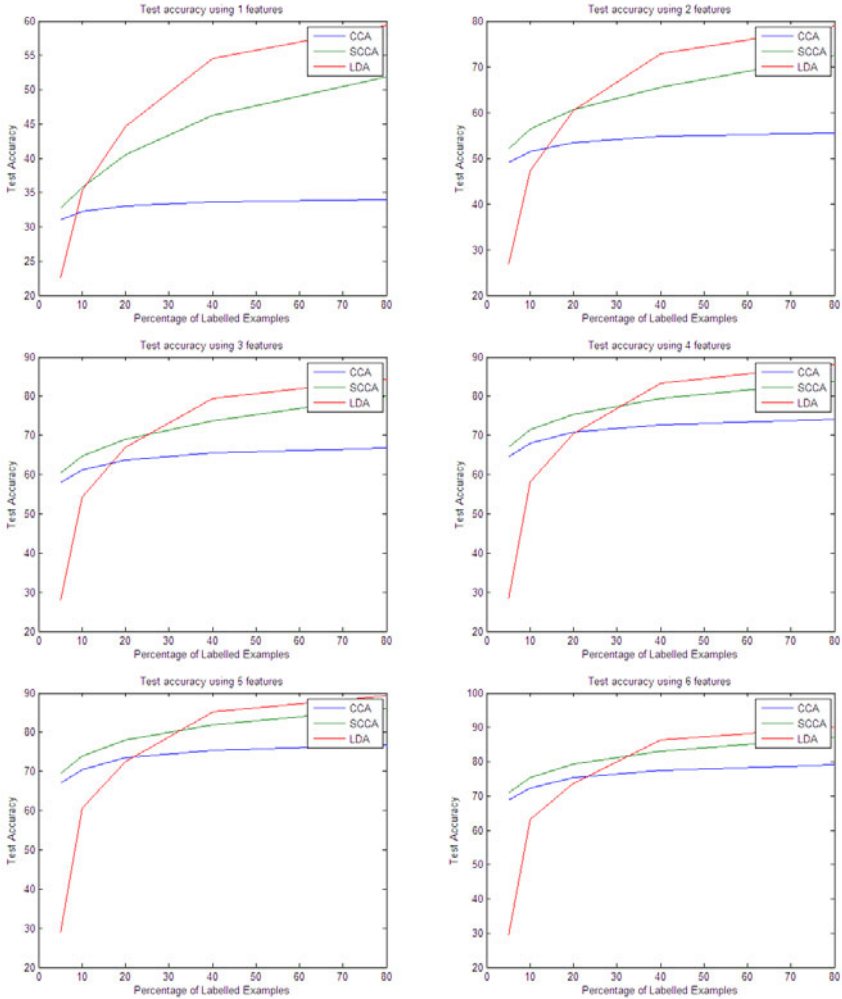
## 5 Experimental Results

For our experiments, we have used “Multi-feature digit dataset” [11] available from UCI machine learning repository [12]. This dataset consists of features of handwritten numerals (digits from ‘0’ to ‘9’) extracted from a collection of Dutch utility maps. 200 samples per class have been digitized and then represented in terms of the following six feature sets:

1. mfeat-fou: 76 Fourier coefficients of the character shapes;
2. mfeat-fac: 216 profile correlations;
3. mfeat-kar: 64 Karhunen-Loève coefficients;
4. mfeat-pix: 240 pixel averages in 2 x 3 windows;
5. mfeat-zer: 47 Zernike moments;
6. mfeat-mor: 6 morphological features.

Among the 200 samples per-class, we used the first 100 for the training (to account for both labelled and unlabelled) and the remaining 100 samples for the testing. We varied the number of labelled/unlabelled samples in the training set to evaluate the contribution of the unlabelled samples to plain LDA that only uses the labelled samples and also to evaluate the contribution of the labelled samples to plain CCA that uses all the available training samples but without benefiting from the class information of the labeled ones. We used CCA implementation in [13].

As some pairs of views can better complement weaknesses of each other than some others, we have avoided picking a particular pair of views; instead, we applied SCCA to all the 15 pairwise combinations of these six views. In Figure 5, we show the test accuracies averaged over  $750 = 15 \times 50$  classification runs (15 pairs of views and 50 random splits of the training set into labelled and unlabelled groups for each view-pair). We can see that for low ratio of labelled samples SCCA achieves the highest accuracy levels and LDA performs poorly. However, as the number of labelled samples increase relative to the unlabelled ones, LDA performs better because the use of unlabelled samples introduce noise and simply shifts the optimal decision boundary



**Fig. 5.** SVM classification accuracies using various number of features extracted (per view) by CCA, LDA, and the proposed SCCA methods

unnecessarily. For the training and testing we used LIBSVM [14] implementation of linear SVM-classifiers and as inputs to the SVM we extracted the same number of features from both views (shown as the title at the top of plots in each panel). The fact that we used linear SVM for classification shows that SCCA features are clearly superior to LDA and CCA features when there are abundance of unlabelled samples.

## 6 Conclusions

In this paper, we proposed a method called SCCA for semisupervised multiview feature extraction. We propose to use CCA with a modification to accommodate the

class-labels through the class centers of the other view. Even though, we limited ourselves to two-view (plus the class-labels) scenario, the results can be generalized to more views [4]. To extract SCCA features of a view, we use that view and also the unlabelled samples of the other view as is; but we transform the labelled samples of the other view by replacing them with their corresponding class-centers in that (other) view. Thus, labelled samples are replaced by their prototypes and provide a form of LDA-like supervision to the proposed CCA-like setup. Thus, if all the samples were labelled, this setup reduces to LDA; and if all the samples were unlabelled, it simply is the plain CCA. However, when there are both labelled and unlabelled samples, our method extracts CCA-like features with preference for LDA-like discriminatory ones. The experimental results on a benchmark, multi-feature digit dataset, shows that SCCA features are clearly more advantageous than both LDA and CCA features when the number of labelled samples are small and there are a large number of unlabelled ones.

## References

1. Fisher, R.A.: The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics* 7, 179–188 (1936)
2. Hotelling, H.: Relations between two sets of variates. *Biometrika* 28, 321–377 (1936)
3. Favorov, O.V., Ryder, D.: SINBAD: a neocortical mechanism for discovering environmental variables and regularities hidden in sensory input. *Biological Cybernetics* 90, 191–202 (2004)
4. Kettenring, J.R.: Canonical analysis of several sets of variables. *Biometrika* 58, 433–451 (1971)
5. Bartlett, M.S.: Further aspects of the theory of multiple regression. *Proc. Camb. Philos. Soc.* 34, 33–40 (1938)
6. Hardoon, D., Szedmak, S., Shawe-Taylor, J.: Canonical correlation analysis: an overview with application to learning methods. *Neural Computation* 16, 2639–2664 (2004)
7. Alpaydin, E.: *Introduction to Machine Learning (Adaptive Computation and Machine Learning Series)*. The MIT Press, Cambridge (2004)
8. Loog, M., van Ginneken, B., Duin, R.P.W.: Dimensionality reduction of image features using the canonical contextual correlation projection. *Pattern Recognition* 38, 2409–2418 (2005)
9. Barker, M., Rayens, W.: Partial least squares for discrimination. *Journal of Chemometrics* 17, 166–173 (2003)
10. Sun, T., Chen, S.: Class label versus sample label-based CCA. *Applied Mathematics and Computation* 185, 272–283 (2007)
11. van Breukelen, M., Duin, R.P.W., Tax, D.M.J., den Hartog, J.E.: Handwritten digit recognition by combined classifiers. *Kybernetika* 34(4), 381–386 (1998)
12. Asuncion, A., Newman, D.J.: *UCI Machine Learning Repository*. University of California, Department of Information and Computer Science, Irvine (2007)
13. Borga, M.: *Learning Multidimensional signal processing*, PhD thesis, Department of Electrical Engineering, Linköping University, Linköping, Sweden (1998)
14. Hsu, C.W., Lin, C.J.: A Comparison of Methods for Multi-Class Support Vector Machines. *IEEE Trans. Neural Networks* 13, 415–425 (2002)

# Evaluation of Distance Measures for Multi-class Classification in Binary SVM Decision Tree

Gjorgji Madzarov and Dejan Gjorgjevikj

Department of Computer Science and Engineering, Ss. Cyril and Methodius  
University, Karpos 2 bb Skopje, Macedonia  
{madzarovg,dejan}@feit.ukim.edu.mk  
[www.feit.ukim.edu.mk](http://www.feit.ukim.edu.mk)

**Abstract.** Multi-class classification can often be constructed as a generalization of binary classification. The approach that we use for solving this kind of classification problem is SVM based Binary Decision Tree architecture (SVM-BDT). It takes advantage of both the efficient computation of the decision tree architecture and the high classification accuracy of SVMs. The hierarchy of binary decision subtasks using SVMs is designed with a clustering algorithm. In this work, we are investigating how different distance measures for the clustering influence the predictive performance of the SVM-BDT. The distance measures that we consider include Euclidian distance, Standardized Euclidean distance and Mahalanobis distance. We use five different datasets to evaluate the performance of the SVM based Binary Decision Tree architecture with different distances. Also, the performance of this architecture is compared with four other SVM based approaches, ensembles of decision trees and neural network. The results from the experiments suggest that the performance of the architecture significantly varies depending of applied distance measure in the clustering process.

**Keywords:** Support Vector Machines, Binary tree architecture, Euclidian distance, Standardized Euclidean distance and Mahalanobis distance.

## 1 Introduction

The recent results in pattern recognition have shown that support vector machine (SVM) [1][2][3] classifiers often have superior recognition rates in comparison to other classification methods. However, the SVM was originally developed for binary decision problems, and its extension to multi-class problems is not straightforward. The popular methods for applying SVMs to multiclass classification problems usually decompose the multi-class problems into several two-class problems that can be addressed directly using several SVMs. Similar to these methods, we have developed an architecture of SVM classifiers utilizing binary decision tree (SVM-BDT) for solving multiclass problems [4]. This architecture uses hierarchy clustering algorithm to convert the multi-class problem into binary tree. The binary decisions in the non-leaf nodes of the binary tree are

made by the SVMs. The SVM-BDT architecture [4] uses Euclidean distance in the clustering process for measuring the classes similarity. Here, we consider two additional distance measures (Standardized Euclidean distance and Mahalanobis distance).

The remainder of this paper is organized as follows: Section 2 describes the SVM-BDT algorithm and the proposed distance measures. The experimental results in section 3 are presented to compare the performance of the SVM-BDT architecture with different distance measures and with traditional multi-class approaches based on SVM, ensemble of decision trees and neural network. Finally, conclusions are presented in Section 4.

## 2 Methodology

### 2.1 Support Vector Machines Utilizing a Binary Decision Tree

SVM-BDT (Support Vector Machines utilizing Binary Decision Tree) [4] is tree based architecture which contains binary SVM in the non leaf nodes. It takes advantage of both the efficient computation of the tree architecture and the high classification accuracy of SVMs. Utilizing this architecture,  $N-1$  SVMs are needed to be trained for an  $N$  class problem, but only  $\log_2 N$  SVMs in average are required to be consulted to classify a sample. This lead to a dramatic improvement in recognition speed when addressing problems with big number of classes.

The hierarchy of binary decision subtasks should be carefully designed before the training of each SVM classifier. There exist many ways to divide  $N$  classes into two groups, and it is critical to have proper grouping for the good performance of SVM-BDT.

The SVM-BDT method is based on recursively dividing the classes in two disjoint groups in every node of the decision tree and training a SVM that will decide in which of the groups the incoming unknown sample should be assigned. The groups are determined by a clustering algorithm according to their class membership and their interclass distance in kernel space.

SVM-BDT method starts with dividing the classes in two disjoint groups  $g_1$  and  $g_2$ . This is performed by calculating  $N$  gravity centres for the  $N$  different classes and the interclass distance matrix. Then, the two classes that have the biggest (in the first case Euclidean, in the second case Standardized Euclidean and in the third case Mahalanobis) distance from each other are assigned to each of the two clustering groups. After this, the class with the smallest distance from one of the clustering groups is found and assigned to the corresponding group. The gravity center of this group and distance matrix are then recalculated to represent the addition of the samples of the new class to the group. The process continues by finding the next unassigned class that is closest to either of the clustering groups, assigning it to the corresponding group and updating the group's gravity center and distance matrix, until all classes are assigned to one of the two possible groups. This defines a grouping of all the classes in two disjoint groups of classes. This grouping is then used to train a SVM classifier

in the root node of the decision tree, using the samples of the first group as positive examples and the samples of the second group as negative examples. The classes from the first clustering group are being assigned to the first (left) sub-tree, while the classes of the second clustering group are being assigned to the (right) second sub-tree. The process continues recursively (dividing each of the groups into two subgroups applying the procedure explained above), until there is only one class per group which defines a leaf in the decision tree.

The recognition of each sample starts at the root of the tree. At each node of the binary tree a decision is being made about the assignment of the input pattern into one of the two possible groups represented by transferring the pattern to the left or to the right sub-tree. This is repeated recursively downward the tree until the sample reaches a leaf node that represents the class it has been assigned to.

An example of SVM-BDT that solves a 7 - class pattern recognition problem utilizing a binary tree, in which each node makes binary decision using a SVM is shown on Fig. 1.a, while Fig. 1.b illustrates grouping of 7 classes.

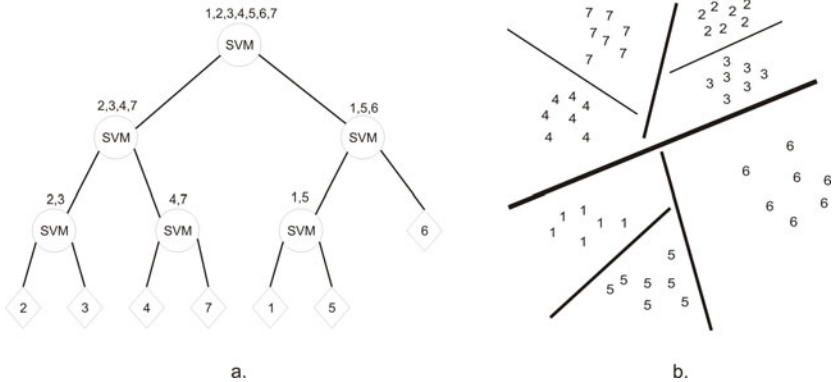


Fig. 1. a. SVM-BDT architecture; b. SVM-BDT divisions of seven classes

### 2.2 Euclidean Distance

Euclidean Distance is the most common used distance measure. In most cases when people said about distance, they will refer to Euclidean distance. Euclidean distance or simply “distance” examines the root of square differences between coordinates of a pair of objects.

$$d_{ij} = \left( \sum (\hat{x}_i - \hat{x}_j)^2 \right)^{\frac{1}{2}}. \tag{1}$$

The gravity centers of the two groups that are obtained by the clustering algorithm in the non leaf nodes of the tree are represented by  $\hat{x}_i$  and  $\hat{x}_j$ .



### 2.3 Standardized Euclidean Distance

The contribution of each feature is different if the distance between two groups is measured by Euclidean Distance. Some form of standardization is necessary to balance out these contributions. The conventional way to do this is to transform the features so they all have the same variance of one. Euclidean Distance calculated on standardized data is called Standardized Euclidean Distance. This distance measure between two groups of samples can be written as:

$$d_{ij} = \left( \Sigma \left( \frac{\hat{x}_i}{\hat{s}_i} - \frac{\hat{x}_j}{\hat{s}_j} \right)^2 \right)^{\frac{1}{2}}, \quad (2)$$

where  $\hat{s}_i$  and  $\hat{s}_j$  are the group  $i$  and the group  $j$  standard deviation vectors respectively. The  $\hat{x}_i$  and  $\hat{x}_j$  are the gravity centers of the group  $i$  and group  $j$ , that are obtained by the clustering algorithm in the non leaf nodes of the tree.

### 2.4 Mahalanobis Distance

Mahalanobis distance [5] is also called quadratic distance. It measures the separation of two groups of samples. It differs from Euclidean distance in that it takes into account the correlations of the data set and is scale-invariant. Suppose we have two groups with means  $\hat{x}_i$  and  $\hat{x}_j$ , Mahalanobis distance is given by

$$d_{ij} = \left( (\hat{x}_i - \hat{x}_j)^T S^{-1} (\hat{x}_i - \hat{x}_j) \right)^{\frac{1}{2}}, \quad (3)$$

where  $S^{-1}$  is an inverse pooled covariance matrix. This matrix is computed using weighted average of covariance matrices of both of the groups.

## 3 Experimental Results

In this section, we present the results of our experiments with several multi-class problems. The performance was measured on the problem of recognition of digits, letters and medical images.

Here, we compare the results obtained by the SVM-BDT method with three different distance measures (Euclidean Distance - SVM-BDT<sub>E</sub>, Standardized Euclidean Distance - SVM-BDT<sub>SE</sub> and Mahalanobis Distance - SVM-BDT<sub>M</sub>) that are used in the clustering process. Also, the performance of this architecture is compared with the one-against-all (OvA) [6], one-against-one (OvO) [7][8], DAGSVM [9], BTS [10], Bagging [11], Random Forests [11], Multilayer Perceptron (MLP, neural network).

The training and testing of the SVMs based methods (SVM-BDT<sub>E</sub>, SVM-BDT<sub>SE</sub>, SVM-BDT<sub>M</sub>, OvO, OvA, DAGSVM and BTS) was performed using a custom developed application that uses the Torch library [13]. For solving the partial binary classification problems, we used SVMs with Gaussian kernel. In these methods, we had to optimize the values of the kernel parameter  $\sigma$  and penalty  $C$ . For parameter optimization we used experimental results.

We also developed an application that uses the same (Torch) library for the neural network classification. One hidden layer with 25 units was used by the neural network. The number of hidden units was determined experimentally.

The classification based on ensembles of decision trees [11] (Bagging and Random Forest) was performed by Clus, a popular decision tree learner based on the principles stated by Blockeel et al. [12]. There were 100 models in the ensembles. The pruning method that we used was C4.5. The number of selected features in the Random Forest method was  $\log_2 K$  where  $K$  is the number of features in the dataset.

In our experiments, five different multi-class classification problems were addressed by each classifying methods. The training and testing time and the recognition performance were recorded for every method.

The first problem was recognition of isolated handwritten digits (10 classes) from the MNIST database [14]. The MNIST database contains grayscale images of isolated handwritten digits. From each digit image, after performing a slant correction, 40 features were extracted. The features are consisted of 10 horizontal, 8 vertical and 22 diagonal projections [15]. The second and the third problem are 10 class problems from the UCI Repository [16] of machine learning databases: Optdigit (64 features) and Pendigit (16 features). The fourth problem was recognition of isolated handwritten letters, a 26-class problem from the Statlog (16 features) collection [17]. The fifth problem was recognition of medical images, a 197-class problem from the IRMA2008 collection [18]. The medical images were described with 80 features obtained by the edge histogram descriptor from the MPEG7 standard [19].

Table 1 through Table 3 show the results of the experiments using 10 different approaches (7 approaches based on SVM, two based on ensembles of decision trees and one neural network) on each of the 5 data sets. Primary we focused on the results achieved from SVM-BDT methods with Euclidean, Standardized Euclidean and Mahalanobis distance. Table 1 gives the prediction error rate of each method applied on each of the datasets. Table 2 and Table 3 shows the testing and training time of each algorithm, for the datasets, measured in seconds, respectively.

The results in the tables show that SVM based methods outperform the other approaches, in terms of classification accuracy. In terms of speed, SVM based methods are faster, with different ratios for different datasets. Overall, the SVM based algorithms were significantly better compared to the non SVM based methods.

The results in Table 1 show that for the MNIST, Pendigit and Optdigit datasets, the SVM-BDT<sub>M</sub> method achieved the best prediction accuracy comparing to SVM-BDT<sub>E</sub> and SVM-BDT<sub>SE</sub> methods. The results in Table 1, also show that for all datasets, the OvA method achieved the lowest error rate, except in the case of Pendigit dataset. It can be noticed that for the 197-class classification problem the prediction error rates, testing and training times of the SVM-BDT<sub>M</sub>, OvO, DAGSVM and BTS are left. These methods are uncompetitive to the other methods for this classification problem because of their long

**Table 1.** The prediction error rate % of each method for every dataset

|                       | 10-class |          |          | 26-class | 197-class |
|-----------------------|----------|----------|----------|----------|-----------|
|                       | MNIST    | Pendigit | Optdigit | Statlog  | IRMA2008  |
| SVM-BDT <sub>E</sub>  | 2.45     | 1.94     | 1.61     | 4.54     | 55.80     |
| SVM-BDT <sub>SE</sub> | 2.43     | 1.90     | 1.65     | 4.55     | 55.00     |
| SVM-BDT <sub>M</sub>  | 2,15     | 1,63     | 1,55     | 4.54     | /         |
| OvO                   | 2.43     | 1.94     | 1.55     | 4.72     | /         |
| OvA                   | 1.93     | 1.70     | 1.17     | 3.20     | 48.50     |
| DAGSVM                | 2.50     | 1.97     | 1.67     | 4.74     | /         |
| BTS                   | 2.24     | 1.94     | 1.51     | 4.70     | /         |
| R. Forest             | 3.92     | 3.72     | 3.18     | 4.98     | 60.80     |
| Bagging               | 4.96     | 5.38     | 7.17     | 8.04     | 64.00     |
| MLP                   | 4.25     | 3.83     | 3.84     | 14.14    | 64.00     |

**Table 2.** Testing time of each method for every dataset measured in seconds

|                       | 10-class |          |          | 26-class | 197-class |
|-----------------------|----------|----------|----------|----------|-----------|
|                       | MNIST    | Pendigit | Optdigit | Statlog  | IRMA2008  |
| SVM-BDT <sub>E</sub>  | 25.33    | 0.54     | 0.70     | 13.10    | 6.50      |
| SVM-BDT <sub>SE</sub> | 24.62    | 0.55     | 0.71     | 13.08    | 6.45      |
| SVM-BDT <sub>M</sub>  | 20.12    | 0.61     | 0.67     | 12.90    | /         |
| OvO                   | 26.89    | 3.63     | 1.96     | 160.50   | /         |
| OvA                   | 23.56    | 1.75     | 1.63     | 119.50   | 19.21     |
| DAGSVM                | 9.46     | 0.55     | 0.68     | 12.50    | /         |
| BTS                   | 26.89    | 0.57     | 0.73     | 17.20    | /         |
| R. Forest             | 39.51    | 3.61     | 2.76     | 11.07    | 34.45     |
| Bagging               | 34.52    | 2.13     | 1.70     | 9.76     | 28.67     |
| MLP                   | 2.12     | 0.49     | 0.41     | 1.10     | 0.60      |

training time. In the first case the SVM-BDT<sub>M</sub> method took several hundred times longer training time. This appeared as a result of the calculation of the inverse pooled covariance matrix in the clustering process, because of the huge number of classes and the big number of features (80), which are characteristic for this classification problem. In the second case the one-against-one methods (OvO, DAGSVM and BTS) took long training and testing time, because of the large number of classifiers that had to be trained (19306) and the large number of classifiers that had to be consulted in the process of classification.

Of the non SVM based methods, the Random Forest method achieved the best recognition accuracy for all datasets. The prediction performance of the MLP method was comparable to the Random Forest method for the 10-class problems and the 197-class problem, but noticeably worse for the 26-class problem. The MLP method is the fastest one in terms of training and testing time, which is evident in Table 2 and Table 3.

**Table 3.** Training time of each method for every dataset measured in seconds

|                       | 10-class |          |          | 26-class | 197-class |
|-----------------------|----------|----------|----------|----------|-----------|
|                       | MNIST    | Pendigit | Optdigit | Statlog  | IRMA2008  |
| SVM-BDT <sub>E</sub>  | 304.25   | 1.60     | 1.59     | 63.30    | 75.10     |
| SVM-BDT <sub>SE</sub> | 285.14   | 1.65     | 1.63     | 64.56    | 73.02     |
| SVM-BDT <sub>M</sub>  | 220.86   | 1.80     | 5.62     | 62.76    | /         |
| OvO                   | 116.96   | 3.11     | 2.02     | 80.90    | /         |
| OvA                   | 468.94   | 4.99     | 3.94     | 554.20   | 268.34    |
| DAGSVM                | 116.96   | 3.11     | 2.02     | 80.90    | /         |
| BTS                   | 240.73   | 5.21     | 5.65     | 387.10   | /         |
| R. Forest             | 542.78   | 17.08    | 22.21    | 50.70    | 92.79     |
| Bagging               | 3525.31  | 30.87    | 49.4     | 112.75   | 850.23    |
| MLP                   | 45.34    | 2.20     | 1.60     | 10.80    | 42.43     |

The results in Table 2 show that the DAGSVM method achieved the fastest testing time of all the SVM based methods for the MNIST dataset. For the other datasets, the testing time of DAGSVM is comparable with BTS and SVM-BDT methods and their testing time is noticeably better than the OvA and OvO methods.

In terms of training speeds, it is evident in Table 3 that among the SVM based methods, SVM-BDT<sub>E</sub> is the fastest one in the training phase except for the MNIST dataset. Due to the huge number of training samples in the MNIST dataset (60000), SVM-BDT<sub>E</sub>'s training time was longer compared to other one-against-one SVM methods. The huge number of training samples increases the nonlinearity of the hyperplane in the SVM, resulting in an increased number of support vectors and increased training time. Also, it is evident that the SVM-BDT<sub>M</sub> method is slower than the SVM-BDT<sub>E</sub> and the SVM-BDT<sub>SE</sub> methods in the training phase for the Optdigit classification problems. This appears as a result of the size of the feature vector (64) which is longer than the feature vectors of the other classification problems.

## 4 Conclusion

In this work, we have reviewed and evaluated several distance measures that can be applied in the clustering process of building the SVM-BDT architecture. In particular, we compared the Euclidean Distance, Standardized Euclidean Distance and Mahalanobis Distance. The predictive accuracy as a criterion of the performance of the classifiers shows that Mahalanobis Distance is the most suitable distance measure for measuring the similarity between classes in the clustering process of constructing the classifier architecture comparing to the other distance measures of the SVM-BDT methods. But, its training time complexity rapidly grows with the number of features of the classification problem and makes it uncompetitive to the other distance measure techniques like Euclidean and Standardized Euclidean

Distances. The SVM-BDT<sub>E</sub> and the SVM-BDT<sub>SE</sub> show similar results for the predictive accuracy and also similar speed in the training and testing phase. Their complexities linearly depend from the characteristics of the classification problems. Comparing to the other SVM and non SVM based methods the SVM-BDT methods with different distance measure show comparable results or offer better recognition rates than the other multi-class methods. The speed of training and testing is improved when we used Euclidean Distance and Standardized Euclidean Distance for measuring the similarity between classes in the clustering process of constructing the classifier architecture.

## References

1. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1999)
2. Burges, C.J.C.: A tutorial on support vector machine for pattern recognition. *Data Min. Knowl. Disc.* 2, 121 (1998)
3. Joachims, T.: Making large scale SVM learning practical. In: Scholkopf, B., Bruges, C., Smola, A. (eds.) *Advances in kernel methods-support vector learning*. MIT Press, Cambridge (1998)
4. Madzarov, G., Gjorgjevikj, D., Chorbev, I.: A multi-class SVM classifier utilizing binary decision tree. *An International Journal of Computing and Informatics, Informatica* 33(2), 233–241 (2009)
5. Mahalanobis, P.: On tests and measures of group divergence I. *Theoretical formulae, J. and Proc. Asiat. Soc. of Bengal* 26, 541–588 (1930)
6. Vapnik, V.: *Statistical Learning Theory*. Wiley, New York (1998)
7. Friedman, J.H.: Another approach to polychotomous classification. Technical report, Department of Statistics, Stanford University (1997)
8. Xu, P., Chan, A.K.: Support vector machine for multi-class signal classification with unbalanced samples. In: *Proceedings of the IJCNN 2003, Portland*, pp. 1116–1119 (2003)
9. Platt, J., Cristianini, N., Shawe-Taylor, J.: Large margin DAGSVMs for multiclass classification. *Advances in Neural Information Processing Sys.* 12, 547–553 (2000)
10. Fei, B., Liu, J.: Binary Tree of SVM: A New Fast Multiclass Training and Classification Algorithm. *IEEE Transaction on neural net.* 17(3) (May 2006)
11. Kocev, D., Vens, C., Struyf, J., Dzeroski, S.: Ensembles of multi-objective decision trees. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) *ECML 2007. LNCS (LNAI)*, vol. 4701, pp. 624–631. Springer, Heidelberg (2007)
12. Blockeel, H., Struyf, J.: Efficient Algorithms for Decision Tree Cross-validation. *Journal of Machine Learning Research* 3, 621–650 (2002)
13. Collobert, R., Bengio, S., Mariethoz, J.: Torch: a modular machine learning software library, Technical Report IDIAP-RR 02-46, IDIAP (2002)
14. MNIST, MiniNIST, USA, <http://yann.lecun.com/exdb/mnist>
15. Gorgevik, D., Cakmakov, D.: An Efficient Three-Stage Classifier for Handwritten Digit Recognition. In: *Proceedings of 17th ICPR 2004, August 23-26, vol. 4*, pp. 507–510. IEEE Computer Society, Cambridge (2004)
16. Blake, C., Keogh, E., Merz, C.: *UCI Repository of Machine Learning Databases* (1998), <http://archive.ics.uci.edu/ml/datasets.html>
17. Statlog, <http://archive.ics.uci.edu/ml/datasets/Letter+Recognition>
18. <http://www.imageclef.org/2008/medaat>
19. Martinez, J.M. (ed.) MPEG Requirements Group, ISO/MPEG N4674, Overview of the MPEG-7 Standard, v 6.0, Jeju (March 2002)

# Triangular Visualization

Tomasz Maszczyk and Włodzisław Duch

Department of Informatics, Nicolaus Copernicus University, Toruń, Poland

tmaszczyk@is.umk.pl, Google: W. Duch

<http://www.is.umk.pl>

**Abstract.** The *TriVis* algorithm for visualization of multidimensional data proximities in two dimensions is presented. The algorithm preserves maximum number of exact distances, has simple interpretation, and unlike multidimensional scaling (MDS) does not require costly minimization. It may also provide an excellent starting point significantly reducing the number of required iterations in MDS.

## 1 Introduction

Almost all datasets in real applications have many input variables that may be inter-related in subtle ways. Such datasets can be analyzed using conventional methods based on statistic, providing a numerical indication of the contribution of each feature to a specific category. Frequently exploratory data analysis is more informative when visual analysis and pattern recognition is done rather than direct analysis of numerical data. The visual interpretation of datasets is limited by human perception to two or three-dimensions. Projection methods and non-linear mapping methods that show interesting aspects of multidimensional data are therefore highly desirable [1].

Methods that represent topographical proximity of data are of great importance. They are usually variants of multidimensional scaling (MDS) techniques [2]. Here a simpler and more effective approach called *TriVis* is introduced, based on the growing structures of triangles that preserve exactly as many distances as possible. Mappings obtained in this way are easy to understand, and may also be used for MDS initialization. Other methods do not seem to find significantly better mappings. In the next section a few linear and non-linear visualization methods are described, and *TriVis* visualization based on sequential construction of triangles is introduced. Illustrative examples for several datasets are presented in section 3. Conclusions are given in the final section.

## 2 Visualization Algorithms

First a short description of two most commonly used methods, principal component analysis (PCA) [1] and multidimensional scaling (MDS) [2], is given. After this, our *TriVis* approach is presented. Comparison of these 3 methods is presented in the next section.

## 2.1 Principal Component Analysis

PCA is a linear projection method that finds orthogonal combinations of input features  $\mathbf{X} = \{x_1, x_2, \dots, x_N\}$  preserving most variation in the data. Principal component directions  $\mathbf{P}_i$  result from diagonalization of data covariance matrix [3]. They can be ordered from most to the least important, according to the size of the corresponding eigenvalues. Directions with maximum variability of data points guarantee minimal loss of information when position of points are recreated from their low-dimensional projections. Visualization can be done taking the first two principal components and projecting the data into the space defined by these components,  $y_{ij} = \mathbf{P}_i \cdot \mathbf{X}_j$ . PCA complexity is dominated by covariance matrix diagonalization, for two highest eigenvalues it is at least  $O(N^2)$ . For some data distributions this algorithm shows informative structures.

Kernelized version of the standard PCA may be easily formulated [4], finding directions of maximum variance for vectors mapped to an extended space. This space is not constructed in an explicit way, the only condition is that the kernel mapping  $K(\mathbf{X}, \mathbf{X}')$  of the original vectors should be a scalar product  $\Phi(\mathbf{X}) \cdot \Phi(\mathbf{X}')$  in an extended space  $\Phi(\mathbf{X})$ . This enables interesting visualization of data, although interpretation is rather difficult.

## 2.2 Multidimensional Scaling

MDS is perhaps the most popular non-linear technique of proximity visualization. The main idea how to decrease dimensionality while preserving original distances in high-dimensional space has been rediscovered several times [5,6,7] and is done either by minimization of specific cost functions [2] or by solving a system of cubic equations [7]. MDS methods need only similarities between objects as inputs, so explicit vector representation of objects is not necessary. Qualitative information about pairwise similarities is sufficient for non-metric MDS, but here only quantitative evaluation of similarity calculated by numerical functions is used. MDS methods differ by their cost functions, optimization algorithms, the type of similarity functions and the use of feature weighting. There are many measures of topographical distortions due to the reduction of dimensionality, most of them weighted variants of the simple stress function [2]:

$$S(\mathbf{d}) = \sum_{i>j}^n W_{ij} (D_{ij} - d_{ij})^2 \quad (1)$$

where  $d_{ij}$  are distances (dissimilarities) in the target (low-dimensional) space, and  $D_{ij}$  are distances in the input space. Weights  $W_{ij} = 1$  for simple stress function, or to reduce effect of large distances  $W_{ij} = 1/D_{ij}$  or  $W_{ij} = 1/D_{ij}^2$  are used. The sum runs over all pairs of objects and thus contributes  $O(n^2)$  terms. In the  $k$ -dimensional target space there are  $kn - k$  parameters for minimization. For visualization purposes the dimension of the target space is  $k = 1, 2$  or  $3$  (usually  $k = 2$ ).

MDS cost functions are not easy to minimize, with multiple local minima representing different mappings. Initial configuration is either selected randomly or

is based on PCA projection. Dissimilar objects are represented by points that are far apart, and similar objects are represented by points that are close, showing clusters in the data. Orientation of axes in the MDS mapping is arbitrary, and the values of coordinates do not have any meaning, as only relative distances are preserved. Kohonen networks [8] are also a popular tool combining clusterization with visualization, but they do not minimize directly any measure of topographical distortion for visualization, therefore their visualization is not as good as that provided by MDS.

### 2.3 Triangular Visualization

*TriVis* algorithm creates representation of data points in two-dimensional space that exactly preserves as many distances between points as it is possible. Distances between any 3 vectors forming a triangle may always be correctly reproduced; a new point is iteratively added relatively to one side of existing triangle, forming a new triangle that exactly preserves two original distances. There are many possibilities of adding such points in relation to the existing triangle sides. To preserve the overall cluster structure 3 furthest points are selected for the start (an alternative is to use centers of 3 clusters), and the new point is chosen to minimize the MDS stress function  $S(\mathbf{d}) = \sum_{i>j}^n (D_{ij} - d_{ij})^2$ . This gives mapping that preserves exactly  $2n - 3$  out of  $n(n-1)/2$  original distances, minimizing overall stress.

---

#### Algorithm 1

---

- 1: Find three farthest vectors and mark them (preserving original distances) as points of the initial triangle.
  - 2: Mark segments (pairs of points) forming triangle sides as available.
  - 3: **for**  $i = 1$  to  $n - 3$  **do**
  - 4: Find the segment AB for which vector  $\mathbf{X}_i$  added as the point  $C=C(\mathbf{X}_i)$  forms a triangle ABC preserving two original distances —AC— and —BC—, and gives the smallest increase of the stress  $S_i = \sum_{j=1}^m (D_{ij} - d_{ij})^2$ .
  - 5: Delete the AB segment from the list of available segments, and add to it segments AC and BC.
  - 6: **end for**
- 

Complexity of this algorithm grows like  $O(n^2)$ , but MDS has to perform minimization over positions of these points while *TriVis* simply calculates positions. To speed up visualization process this algorithm could be applied first to  $K$  vectors selected as the nearest points to the centers of  $K$  clusters (for example using the K-means or dendrogram clusterization). Plotting these points should preserve the overall structure of the data, and applying *TriVis* to points within each cluster decomposes the problem into  $K$  smaller  $O(n_k^2)$  problems. For large number of vectors the use of jitter technique may be sufficient, plotting the vectors that belong to one specific cluster near the center of this cluster, with dispersion equal to the mean distance between these points and the center.



To measure what can be gained by full MDS minimization *TriVis* mapping should be used as a starting configuration for MDS. This should provide much lower stress at the beginning reducing the number of iterations.

### 3 Illustrative Examples

The usefulness of the *TriVis* sequential visualization method has been evaluated on four datasets downloaded from the UCI Machine Learning Repository [9] and from [10]. A summary of these datasets is presented in Tab. 1; their short description follows:

1. **Iris** the most popular dataset, it contains 3 classes of 50 instances each, where each class refers to a type of the Iris flowers.
2. **Heart** disease dataset consists of 270 samples, each described by 13 attributes, 150 cases belongs to group “absence” and 120 to “presence of heart disease”.
3. **Wine** wine data are the results of a chemical analysis of wines, grown in the same region in Italy, but derived from three different cultivars. 13 features characterizing each wine are provided, the data contains 178 examples.
4. **Leukemia**: microarray gene expressions for two types of leukemia (ALL and AML), with a total of 47 ALL and 25 AML samples measured with 7129 probes [10]. Visualization is based on 100 best features from simple feature ranking using FDA index [1].

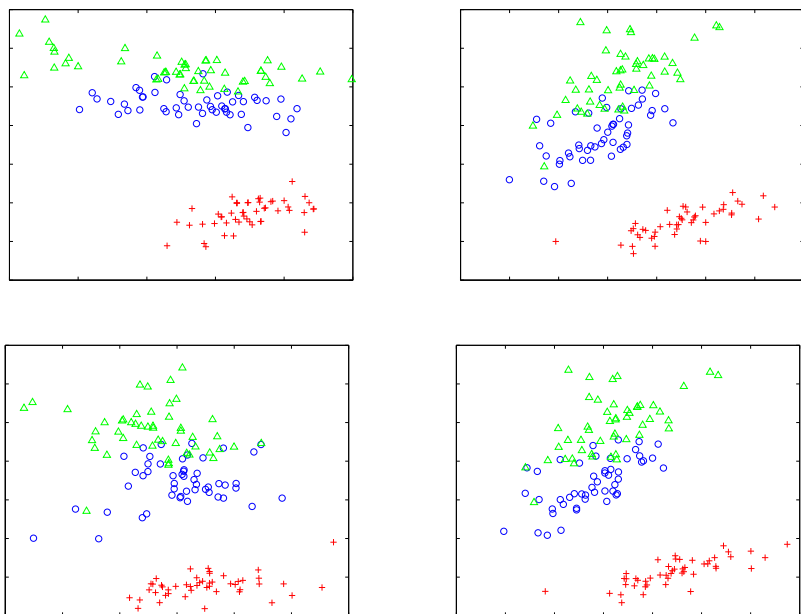
For each dataset two-dimensional mappings have been created using PCA, *TriVis*, MDS starting from random configuration and MDS starting from *TriVis* configuration (Figs. 1-4). Visualization is sensitive to feature selection and weighting and frequently linear projections discovered through supervised learning may be more informative [11,12]. Since our goal here is not the best visualization but rather comparison of *TriVis* algorithm with PCA and MDS methods all features have been used.

Mappings of both Iris and Cleveland Heart datasets are rather similar for all 4 techniques (selecting only relevant features will show a better separation between classes); PCA shows a bit more overlap and MDS optimization of *TriVis* configuration does not provide more information than the initial configuration.

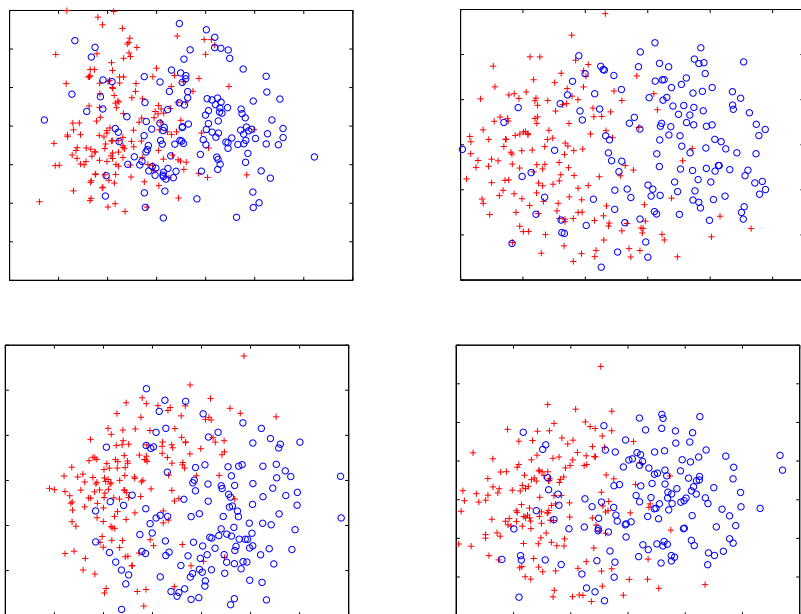
Wine dataset does not map well with PCA and different classes are somehow better separated using MDS with *TriVis* initialization. This is a good example

**Table 1.** Summary of datasets used for illustrations

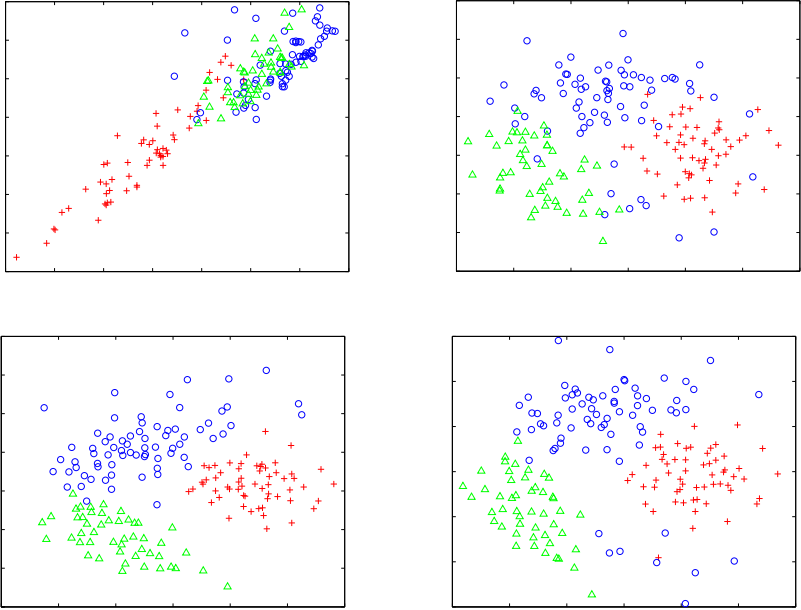
| Title    | #Features | #Samples | #Samples per class   |                      |                      | Source |
|----------|-----------|----------|----------------------|----------------------|----------------------|--------|
| Iris     | 4         | 150      | 50 “Setosa”          | 50 “Virginica”       | 50 “Versicolor”      | [9]    |
| Heart    | 13        | 303      | 164 “absence”        | 139 “presence”       |                      | [9]    |
| Wine     | 13        | 178      | 59 “C <sub>1</sub> ” | 71 “C <sub>2</sub> ” | 48 “C <sub>3</sub> ” | [9]    |
| Leukemia | 100       | 72       | 47 “ALL”             | 25 “AML”             |                      | [10]   |



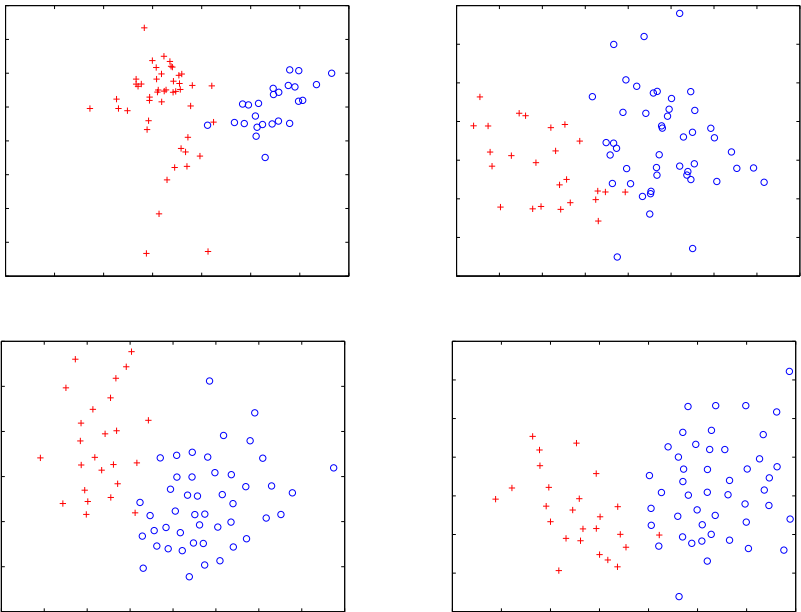
**Fig. 1.** Iris dataset, top row: PCA and *TriVis*, bottom row: typical (randomly initialized) MDS and MDS initialized by *TriVis*



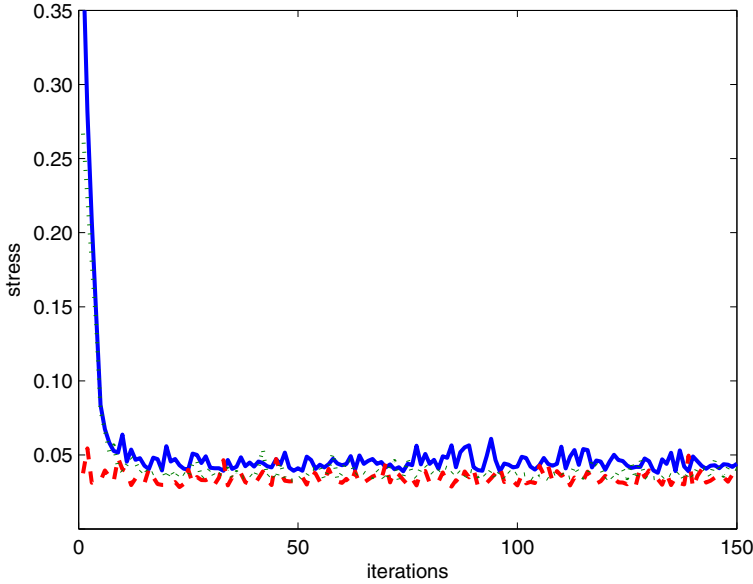
**Fig. 2.** Cleveland Heart dataset, top row: PCA and *TriVis*, bottom row: typical MDS and MDS initialized by *TriVis*



**Fig. 3.** Wine dataset, top row: PCA and *TriVis*, bottom row: typical MDS and MDS initialized by *TriVis*



**Fig. 4.** Leukemia dataset, top row: PCA and *TriVis*, bottom row: typical MDS and MDS initialized by *TriVis*



**Fig. 5.** Comparison of 3 types of MDS initialization (Wine dataset): solid blue line - random, dotted green line - PCA, dashed red - *TriVis*

showing that using *TriVis* configuration as the start for MDS leads to faster and better convergence.

Leukemia shows good separation using *TriVis* projection (Fig. 4), providing a bit more interesting projection than other methods.

To show the influence of *TriVis* initialization on MDS comparison of the convergence starting from random, PCA and *TriVis* configurations is presented in Fig. 5. *TriVis* initialization clearly works in the best way leading to a convergence in a few iterations and achieving the lowest stress values. This type of initialization may prevent MDS from getting stuck in poor local minimum.

## 4 Conclusions

In exploratory data analysis PCA and MDS are the most popular methods for data visualization. Visualization based on proximity helps to understand the structure of the data, to see the outliers and to place interesting cases in their most similar context, it may also help to understand what black box classifiers really do [13,14]. In safety-critical areas visual interpretation may be crucial for acceptance of proposed methods of data analysis.

The *TriVis* algorithm presented in this paper has several advantages: it enables visualization of proximities in two dimensions, preserves maximum number of exact distances reducing distortions of others, has simple interpretation, allows for simple assessment of various similarity functions and feature selection

and weighting techniques, it may unfold various manifolds [15] (hypersurfaces embedded in high-dimensional spaces). For large datasets it may be coupled with hierarchical dendrogram clusterization methods to represent with high accuracy relations between clusters. PCA does not preserve proximity information, while MDS is much more costly and does not seem to have advantages over *TriVis*. If MDS visualization is desired *TriVis* gives an excellent starting point significantly reducing the number of required iterations.

## References

1. Webb, A.: Statistical Pattern Recognition. J. Wiley & Sons, Chichester (2002)
2. Cox, T., Cox, M.: Multidimensional Scaling, 2nd edn. Chapman and Hall, Boca Raton (2001)
3. Jolliffe, I.: Principal Component Analysis. Springer, Berlin (1986)
4. Schölkopf, B., Smola, A.: Learning with Kernels. In: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge (2001)
5. Torgerson, W.: Multidimensional scaling. I. Theory and method. *Psychometrika* 17, 401–419 (1952)
6. Sammon, J.: A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers* C18, 401–409 (1969)
7. Duch, W.: Quantitative measures for the self-organized topographical mapping. *Open Systems and Information Dynamics* 2, 295–302 (1995)
8. Kohonen, T.: Self-organizing maps. Springer, Heidelberg (1995)
9. Asuncion, A., Newman, D.: UCI machine learning repository (2007), <http://www.ics.uci.edu/~mllearn/MLRepository.html>
10. Golub, T.: Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science* 286, 531–537 (1999)
11. Maszczyk, T., Duch, W.: Support vector machines for visualization and dimensionality reduction. In: Kůrková, V., Neruda, R., Koutník, J. (eds.) ICANN 2008, Part I. LNCS, vol. 5163, pp. 346–356. Springer, Heidelberg (2008)
12. Maszczyk, T., Grochowski, M., Duch, W.: Discovering Data Structures using Meta-learning, Visualization and Constructive Neural Networks. *Studies in Computational Intelligence*, vol. 262. Springer, Heidelberg (in print 2010)
13. Duch, W.: Visualization of hidden node activity in neural networks: I. visualization methods. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) ICAISC 2004. LNCS (LNAI), vol. 3070, pp. 38–43. Springer, Heidelberg (2004)
14. Duch, W.: Coloring black boxes: visualization of neural network decisions. In: Int. Joint Conf. on Neural Networks, Portland, Oregon, vol. I, pp. 1735–1740. IEEE Press, Los Alamitos (2003)
15. Tenenbaum, J., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* 290, 2319–2323 (2000)

# Recognition of Finite Structures with Application to Moving Objects Identification

Ewaryst Rafajłowicz and Jerzy Wietrzych

Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50 370 Wrocław, Poland  
ewaryst.rafajlowicz@pwr.wroc.pl

**Abstract.** Our aim is to discuss problems of structure recognition in the Bayesian setting, treating structures as special cases of relations. We start from a general problem statement, which is solvable by dynamic programming for linear structures. Then, we consider splitting the problem of structure recognition into a series of pairwise relations testing, which is applicable when on-line processing of intensive data streams is necessary. An appropriate neural network structure is also proposed and tested on a video stream.

## 1 Introduction

In recent years the recognition of structures has been an area of intensive research in many disciplines. Problems of structure recognition arise in chemistry and biology [16], [17], [24], physics [23], document processing [7], [11], bibliographic references [19], recognition of hand-written texts (the oldest area of applications, dating for sixties) and mathematical formulas [2]. Mathematical tools that are used for structure recognition include: graphs, trees that are derived either from a set of rules or from formal grammars [20], [9], [21], hierarchical multi-level knowledge representation scheme, neural or neuro-fuzzy systems [15], Bayesian inference [17], [8], Markov random fields [18], statistical mechanics [1], Delaunay triangulation with partial least squares projection [24], computational intelligence [6] and the bibliography cited therein.

## 2 Structures as Relations and Their Recognition

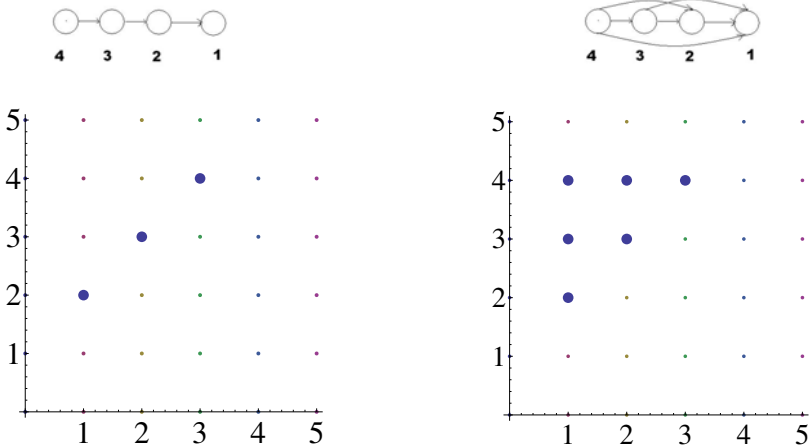
The notion of structure is very broad. This term is used in a number of disciplines such as anatomy, botany, geology, chemistry, psychology, linguistics, mathematics, economics, engineering, among others. Trying to recognize structures we have to define them more precisely, bearing in mind that it should still be a notion with a large capacity. The idea is to use the well known notion of a relation in order to describe structures in a way, which is convenient for further considerations. For two sets  $A$  and  $B$  a relation  $R$  is any subset of  $A \times B$ , including empty relation  $R = \emptyset$  and full relation  $R = A \times B$ . We shall write  $a R b$  if  $a$  and

$b$  are in relation  $R$ , i.e., if  $(a, b) \in R$ . Similarly,  $a \neg R b$  if  $(a, b) \notin R$ , i.e., they are not related. In the following definition we impose additional restrictions on relations, which seem to coincide with our intuition what is a structure. Note that we admit  $a R a$  in structures.

**Definition 1.** Let  $R \subset A \times A$  be a relation on the cartesian product of  $A \times A$ . We say that  $R$  is a structure if

- 1)  $R$  is antisymmetric, i.e.,  $a R a'$  and  $a' R a$ , implies  $a = a'$ ,  $a, a' \in A$ ,
- 2)  $R$  is transitive, i.e., if  $a R a'$  and  $a' R a''$ , then  $a R a''$ .

Additionally, we concentrate on relations, which are defined on finite sets  $A$ . Without further loss of generality, we can identify discrete sets  $A$  with finite sets of positive integers. Thus,  $A = \{1, 2, \dots, n_A\}$ . We emphasize that the description of structures as special relations is formally equivalent to other descriptions, e.g., as graphs, but a relational description is more convenient for our purposes and sometimes more can be seen when we look at figures representing relations. It is expedient to consider two examples.



**Fig. 1.** Structures and their representations as relations (left panel – linear structure, right panel – quasi-linear structure)

**Linear structure** is shown in Fig. 1 (left panel). Even in this simple case the numbering of objects (dots) entering into relations is important in the sense that for different numberings of the objects we obtain different patterns on the right hand side figures.

**Quasi-linear structure** is shown in Fig. 1 (right panel). It differs from the previous one, since all direct relations between objects are present.

It is important for further considerations that structures with different numbering of nodes are treated as different. This point of view is justified by the fact that we would like to treat labels of the nodes as related to real objects.

**Definition 2.** Given a finite set  $A$  and relation  $R$ , for which conditions listed in Definition 1 hold. The following set  $S = \{(a, b) : a R b, a, b \in A\}$  is called a finite structure.

The advantage of the above representation of structures is that it allows us to compose a structure from pairs (dots) and even when not all of them are recognized, we obtain a meaningful description of substructures.

If for each pair  $a, b$  we know whether they are related or not, then the problem of structure recognition does not arise. It suffices to find a convenient representation of this structure. We are interested in cases where it is not certain whether  $a R b$  or  $a \neg R b$  is true for each pair  $a, b \in A$  and we have to make decisions, which are based on vector  $X(a, b) \in \mathbf{R}^d, d \geq 1$  of observations. Vector  $X(a, b)$  may contain features of  $a$  and  $b$  objects and information, possibly vague, concerning their relations. For a finite structure  $S$  the set of observations has the following form:  $\mathbf{X} \stackrel{def}{=} \{X(a, b) : (a, b) \in S\}$ . Taking into account that information contained in  $\mathbf{X}$  is not sufficient for unequivocal decisions, which structure corresponds to  $\mathbf{X}$  we want to design a recognizer  $\Psi(\mathbf{X}) \in \mathcal{S}$ , minimizing possible misclassifications, where  $\mathcal{S}$  denotes the class of all structures  $S$  that may appear in the problem at hand. We shall consider two cases.

- 1) When we a priori know all the probabilities of events  $a R b, (a, b) \in S$ , conditioned on  $\mathbf{X}$ , then one can try to build the Bayesian classifier. We shall discuss this case briefly in the next section.
- 2) However, these probabilities are rarely known and we have to build a classifier, which incorporates knowledge from a learning sequence.

**Optimal classifier for linear structures.** Consider the class of all linear structures having length  $L \geq 2$  and denote it by  $\mathcal{S}_L$  (see Fig. 1). It contains  $L!$  different structures, which can be "numbered" by all the permutations of labels  $1, 2, \dots, n$ . Denote by  $\Pi(n)$  the set of all permutations of  $1, 2, \dots, n$ , while by  $\pi(n) \in \Pi(n)$  we denote its elements, i.e.,  $\pi(n) = (j_1, j_2, \dots, j_n)$ , where  $j_i \in \{1, 2, \dots, n\}$ . Let us assume for a moment that we have all the following conditional probabilities

$$P(j_1 R j_2, j_2 R j_3, \dots, j_{n-1} R j_n | \mathbf{X}), j_i \in \{1, 2, \dots, n\}, i = 1, 2, \dots, n, \tag{1}$$

at our disposal, where the conditioning is on the vector of observations  $\mathbf{X}$ , which corresponds to structure  $S$  from  $\mathcal{S}_L$  and  $S$  is unknown. Let us also assume that we adopt the 0-1 loss function in the Bayesian problem of classification (see 5). Then, the optimal classifier  $\Psi^*(\mathbf{X})$  provides as its output permutation  $\pi^*(n) = (j_1^*, j_2^*, \dots, j_n^*)$  of labels  $1, 2, \dots, n$  for which the maximum

$$\max_{\pi(n) \in \Pi(n)} P(j_1 R j_2, j_2 R j_3, \dots, j_{n-1} R j_n | \mathbf{X}) \tag{2}$$

is attained. In other words, the optimal classifier is the maximum a posteriori probability rule (see 5).

The maximization problem stated by (2) is a formidable and – in general – non-tractable task for the following reasons.



1) From the computational point of view (2) is a discrete (combinatorial) optimization problem. In order to illustrate its complexity, assume  $n = 20$  and suppose that our computer calculates  $10^9$  probabilities in (2) per second. Then, a brute force look up would take about 28 000 days.

2) Probabilities in (2) are not only unknown, but it almost impossible to estimate  $n!$  functions of  $\mathbf{X}$  from empirical data.

Therefore, we have to incorporate additional knowledge as assumptions concerning probabilities in (2). Later on we assume the following

$$P(j_1 R j_2, j_2 R j_3, \dots, j_{n-1} R j_n | \mathbf{X}) = \prod_{i=1}^{n-1} P(j_i R j_{i+1} | \mathbf{X}), \tag{3}$$

which means that objects forming a linear structure enter into relations in the stochastically independent way, conditionally on vector of observations. Probabilities  $P(j_i R j_{i+1} | \mathbf{X})$  are formally conditioned on all the observations in  $\mathbf{X}$ . Fortunately, in practice one may expect that  $P(j_i R j_{i+1} | \mathbf{X})$ 's are conditioned only on sub-vectors of  $\mathbf{X}$ .

**Bayesian recognizer by dynamic programming.** Under assumption (3) we can equivalently restate problem (2) as follows

$$Q_n^* \stackrel{def}{=} \max_{\pi(n) \in \Pi(n)} \sum_{i=1}^{n-1} \log[P(j_i R j_{i+1} | \mathbf{X})] = \tag{4}$$

$$= \max_{\pi(n-1) \in \Pi(n-1)} \left[ \max_{j_1 \in \mathcal{N}(n)} \log[P(j_1 R j_2 | \mathbf{X})] + \sum_{i=2}^{n-1} \log[P(j_i R j_{i+1} | \mathbf{X})] \right],$$

since  $\log(\cdot)$  is strictly increasing function, while the decomposition in the second row of (4) follows from the fact that  $j_1$  enters only into the first term of the sum.  $j_1^*$ , which maximizes  $[\max_{j_1 \in \mathcal{N}(n)} \log[P(j_1 R j_2 | \mathbf{X})]]$  is a function of  $j_2$  and  $\mathbf{X}$ , which is further denoted by  $j_1^* = \phi_1^*(j_2, \mathbf{X})$ . Denote by  $\Phi_1^*(j_2, \mathbf{X}) = \log[P(\phi_1^*(j_2, \mathbf{X}) R j_2 | \mathbf{X})]$ . Then,  $Q_n^*$  can be rewritten as follows

$$Q_n^* = \max_{\pi(n-2) \in \Pi(n-2)} \left[ \max_{j_2 \in \mathcal{N}(n)} [\Phi_1^*(j_2, \mathbf{X}) + \log[P(j_2 R j_3 | \mathbf{X})] + \tag{5}$$

$$+ \sum_{i=3}^{n-1} \log[P(j_i R j_{i+1} | \mathbf{X})] \right].$$

Analogously as above, denote

$$j_2^* = \phi_2(j_3, \mathbf{X}) = arg \max_{j_2 \in \mathcal{N}(n)} [\Phi_1^*(j_2, \mathbf{X}) + \log[P(j_2 R j_3 | \mathbf{X})]]$$

$$\Phi_2^*(j_3, \mathbf{X}) = \Phi_1^*(\phi_2(j_3, \mathbf{X}), \mathbf{X}) + \log[P(\phi_2(j_3, \mathbf{X}) R j_3 | \mathbf{X})],$$

which lead to

$$Q_n^* = \max_{\pi(n-3) \in \Pi(n-3)} \left[ \max_{j_3 \in \mathcal{N}(n)} [\Phi_2^*(j_3, \mathbf{X}) + \log[P(j_3 R j_4 | \mathbf{X})] + \sum_{i=4}^{n-1} \log[P(j_i R j_{i+1} | \mathbf{X})]] \right]. \tag{6}$$

Now, it is clear that we can repeat analogous steps by induction. The last one is as follows  $Q_n^* = \max_{j_n \in \mathcal{N}(n)} [\Phi_{n-1}^*(j_n, \mathbf{X})]$  and  $j_n^* = \phi_n(\mathbf{X})$  depends only on  $\mathbf{X}$ . Substituting it back we obtain

$$j_{n-1}^* = \phi_{n-1}(\phi_n(\mathbf{X}), \mathbf{X}), \dots, j_1^* = \phi_1^*(\phi_2^*(\dots, \phi_n(\mathbf{X}), \mathbf{X}) \dots), \tag{7}$$

which is – formally – the solution of our problem. Two remarks are in order concerning the above scheme of calculating the optimal recognizer.

1) We were able to use the dynamic programming scheme due to the assumption that a structure to be recognized is linear. It is possible to extend, although not easy, to extend this way of reasoning to tree-like structures. However, it is not applicable to all structures, as those represented by graphs with cycles.

2) Even if probabilities  $P(j_i R j_{i+1} | \mathbf{X})$  are known, functions  $j_k^* = \phi_k(j_{k-1}, \mathbf{X})$  have to be calculated and approximated numerically, except very special cases.

### 3 Recognizing Separate Relations in Structures

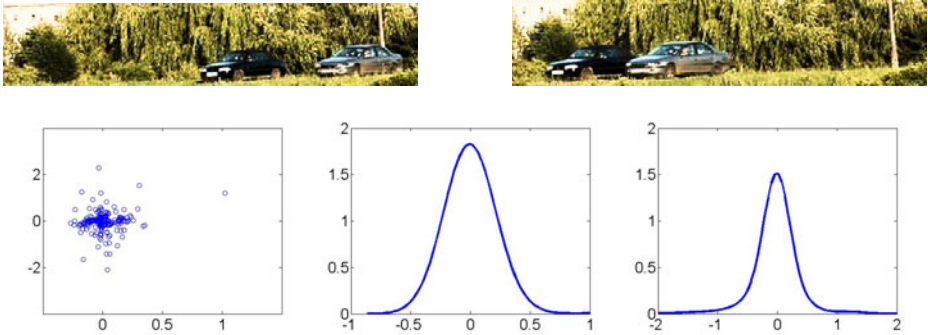
The conclusion from the above remarks is that we have to make further simplifications in stating problems of structure recognition in order to make them solvable in a reasonable time and less data demanding. The proposed simplification is to divide the problem of structure recognition to the separated problems of recognizing pairwise relations  $a R b$ . This approach is motivated by our discussion in Section 2, where structures are embedded into relations. The additional advantages of this approach are that it is not limited to linear or tree-like structures and the corresponding probabilities are conditioned on observations  $X(a, b)$  directly related to  $a R b$ , which reduces the difficulties in estimating them from observations. Thus, we assume that the conditional probabilities  $P(a R b | X(a, b))$ ,  $a, b \in A$  are available or they can be estimated from available observations.

According to the maximum a posteriori probability rule we propose to recognize structure  $S$  as follows:

**Step 1.** Calculate (or estimate) all the probabilities  $P(a R b | X(a, b))$ ,  $a, b \in A$  (their number is at most  $n_A^2$ ).

**Step 2.** Find all the pairs  $(a, b)$ , for which  $P(a R b | X(a, b)) > \beta$ , where  $0 < \beta < 1$  is a prescribed level, which specifies the level of meaningful probability that  $a R b$  is true.

**Step 3.** Mark the pairs selected in Step 2 as a point on a figure representing relations and interpret the obtained set of relations as a structure.



**Fig. 2.** Upper row – two cars in two positions. Lower row: left panel – scatter plot of the learning sequence, middle panel – estimated p.d.f. of log position ratio, right panel – estimated p.d.f. of log areas ratio (see Section 4) for more explanations)

When probabilities  $P(a R b | X(a, b))$  are not known, then one may choose one of the following ways, depending on the kind of available observations.

**Case 1.** If our learning sequence is so rich that it contains repeated observations of random vector  $X(a, b)$  for each pair  $(a, b)$ , then one can estimate the probability density function (p.d.f.), denoted as  $f_{ab}(x)$ , of observations  $X(a, b)$ , conditioned on  $a R b$  and a priori probability  $p_{ab} > 0$  that  $a R b$ ,  $\sum_{a, b \in A} p_{ab} = 1$ . Then, estimated densities and a priori probabilities can be plugged-in into  $P(a R b | X(a, b) = x) = p_{ab} f_{ab}(x) / f_{unc}(x)$ , where  $f_{unc}(x) \stackrel{def}{=} \sum_{a, b \in A} f_{ab}(x)$ .

**Case 2.** If our learning sequence is not so rich as in Case 1 and if objects entering into relations in a structure are of the same kind, then we can consider the decision whether  $a R b$ , given  $X(a, b)$ , as one-class classification problem. In such a case we have only positive examples in a learning sequence, i.e., only those  $X(a, b)$  for which  $a R b$ . Let us denote by  $f(x)$  the probability density function of all these cases, assuming that it does not depend on a particular pair  $a, b$ . For a prescribed confidence level  $0 < \beta < 1$  select set  $\Omega$  such that  $\int_{\Omega} f(x) dx \geq \beta$ . The choice of  $\Omega$  is not unique and usually a kind of symmetry is imposed on  $\Omega$ .

In Case 2 Steps 1 and 2 should be replaced by the following.

**Step 1’.** For a given or estimated  $f$  and  $\beta$  choose confidence region  $\Omega$ .

**Step 2’.** Select all pairs  $(a, b)$  such that the corresponding  $X(a, b) \in \Omega$ .

If one can assume that features in  $X(a, b)$  are stochastically independent, then the choice of  $\Omega$  is easy. Indeed, independence of features implies that  $f(x) = \prod_{k=1}^d f_k(x_k)$ , where  $f_k$ ’s are p.d.f.’s of features  $x_k$ ’s, which form  $X(a, b)$ . Thus, we may select  $\Omega = \omega_1 \times \dots \times \omega_d$ , where  $\omega_k$  is an interval such that  $\int_{\omega_k} f_k(x_k) dx_k = \beta^{1/d}$ . If  $x_k \in \omega_k$  for all  $k$ , then we claim for the corresponding  $a R b$  with the probability  $\beta$ . If  $f$  is estimated from observations then it can be difficult to find  $\Omega$ . In such a case, especially when an on-line application is necessary, it is expedient to implement the following neural network structure,

## Neural network structure

**An input layer** divides  $X(a, b)$  into separate features, which are feed as inputs of parallel branches described below.

**A hidden layer** consists of parallel branches, which estimate  $f_k$ 's as univariate radial basis functions (or – in other words – as Parzen-Rosenblatt estimates (see [4] with the kernels placed over selected centers)

**An output layer** multiplies outputs from the hidden layer, producing the estimate  $f(X(a, b))$ . If  $f(X(a, b)) > \gamma$ , then we claim  $a R b$ .

Selecting  $\gamma > 0$  we define a certain level set, which – implicitly – defines  $\Omega$ . However, if  $f$  is estimated, then it is advisable to select  $\gamma$  experimentally. We refer the reader to [3], [14], [13], [25] for RBF networks, its variants and learning and to [22] for alternative approach, based on orthogonal expansions.

**Identifying moving objects from a video stream.** Our aim is to illustrate the above method of pairwise relations finding by example of identifying the structure of the motion of cars on a road. In the upper row of Fig. 2 two subsequent frames are shown. Our task is to relate "old" copies of cars with new ones. We skip irrelevant details of low level image processing, which provide us the following data:  $x_{new}(l)$ ,  $x_{old}(l)$ ,  $a_{new}(l)$ ,  $a_{old}(l)$ ,  $l = 1, 2, \dots, L$ , where  $L$  is the number of cars (assume the same for subsequent frames for simplicity), their old and new horizontal positions and areas of the smallest rectangle containing their silhouettes, respectively. As input data to our recognition system ( $X_1(k, l)$ ,  $X_2(k, l)$ ,  $k, l = 1, 2 \dots, L$  we have selected the following ratios:  $X_1(k, l) = \log(x_{new}(l)/x_{old}(k))$ ,  $X_2(k, l) = \log(a_{new}(l)/a_{old}(k))$ . For the learning sequence only ratios corresponding to the same car were selected, i.e.,  $X_1(k, k)$  and  $X_2(k, k)$ . The scatter plot of the learning sequence extracted from 352 frames is shown in Fig. 2. Our first step in learning was an attempt to built branches of the neural net described earlier. However, it occur that it is not necessary, since  $X_1(k, k)$ 's and  $X_2(k, k)$ 's perfectly fits normal distributions  $N(0, 0.1)$  and  $N(-0.02, 0.35)$ , respectively and they can be treated as independent (for simultaneous testing normality and independence one can use the test proposed in [12]). Thus, it suffices to select a rectangle  $\Omega = [-0.3, 0.3] \times [-1.15, 1.15]$  and check whether new pairs  $(X_1(k, l), X_2(k, l))$ 's fall into  $\Omega$ . Selecting edges of  $\Omega$  by using  $3\sigma$  rule of thumb, we assure that  $\beta \approx 0.99$ . This simple recognizer was tested on a sequence containing 115 frames. The percentage of correct classifications was 85.3. It can be increased by introducing additional features of cars, since the area of their silhouettes does not always discriminate small cars properly.

**Acknowledgement.** This work was supported by a grant from the Polish Ministry of Science and Higher Education under a grant ranging from 2006 to 2009.

## References

1. Biehl, M., Mietzner, A.: Statistical mechanics of unsupervised structure recognition. *J. Phys. A: Math. Gen.* 27, 1885–1897 (1994)
2. Blostein, D., Grabavec, A.: Recognition of mathematical notation. In: Wang, P.S.P., Bunke, H. (eds.) *Handbook on Optical Character Recognition and Document Image Analysis*. World Sci., Singapore (1996)
3. Bishop, C.: *Neural Networks for Pattern Recognition*. Oxford Univ. Press, Oxford (1995)
4. Devroye, L., Györfi, L.: Nonparametric Density Estimation. In: *The  $L_1$  View*. Wiley, New York (1985)
5. Devroye, L., Györfi, L., Lugosi, G.: *Probabilistic Theory of Pattern Recognition*. Springer, New York (1996)
6. Duch, W., Setiono, R., Żurada, J.: Computational Intelligence Methods for Rule-Based Data Understanding. *Proc. IEEE* 92, 771–805 (2004)
7. Fankhauser, P., Xu, Y.: An incremental approach to document structure recognition. *Electronic Publishing* 6(4), 1–12 (1993)
8. Friedman, N., Koller, D.: Being Bayesian about network structure: A Bayesian approach to structure discovery in Bayesian networks. *Machine Learning* 50, 95–126 (2003)
9. Gonzalez, R., Thomason, M.: *Syntactic pattern recognition: an introduction*. Advanced Book Program. Addison-Wesley Pub. Co, Reading (1978)
10. Hastie, T., Tibshirani, R.: Classification by Pairwise Coupling. *The Annals of Statistics* 26, 451–471 (1998)
11. Hu, T., Ingold, R.: A mixed approach toward an efficient logical structure recognition from document images. *Electronic Publishing* 6, 457–468 (1993)
12. Kallenberg, W., Ledwina, T., Rafajłowicz, E.: Testing bivariate independence and normality. *Sankhya. Ser. A* 59, 42–59 (1997)
13. Karayiannis, N.B., Randolph-Gips, M.M.: On the Construction and Training of Reformulated Radial Basis Function Neural Networks. *IEEE Trans. on Neural Networks* 14, 835–846 (2003)
14. Krzyżak, A., Skubalska-Rafajłowicz, E.: Combining Space-Filling Curves and Radial Basis Function Networks. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) *ICAISC 2004. LNCS (LNAI)*, vol. 3070, pp. 229–234. Springer, Heidelberg (2004)
15. Kwok, T.Y., Yeung, D.Y.: Constructive algorithms for structure learning in feed-forward neural networks for regression problems. *IEEE Trans. Neural Networks* 8, 630–645 (1997)
16. Lathrop, R.H., Webster, T.A., Smith, T.F.: ARIADNE: pattern-directed inference and hierarchical abstraction in protein structure recognition. *Communications of the ACM* 30(11), 909–921 (1987)
17. Lathrop, R.H., Rogers, R.G., Smith, T.F., White, V.J.: A Bayes-optimal Sequence-structure Theory that Unifies Protein Sequence-structure Recognition and Alignment. *Bulletin of Mathematical Biology* 60, 1039–1071 (1998)
18. Li, S.Z.: Markov Random Filed Models in Computer Vision. In: *Proc. European Conf. on Computer Vision*, Stockholm, May, vol. B, pp. 361–370 (1994); bibitemp3 Michalski, R.S.: Discovering classification rules using variable valued logic system, VL1. In: *Third International Joint Conference on Artificial Intelligence*, pp. 162–172 (1973)

19. Parmentier, F., Belaid, A.: Logical Structure Recognition of Scientific Bibliographic References. In: ICDAR 1997, Ulm, Germany, August 18-20 (1997)
20. Pavlidis, T.: Structural pattern recognition. Springer, Berlin (1977)
21. Quinlan, J.R.: Learning logical definitions from relations. *Machine Learning* 5, 239–266 (1990)
22. Skubalska-Rafajłowicz, E.: Pattern Recognition Algorithms Based on Space-Filling Curves and Orthogonal Expansions. *IEEE Trans. Inf. Th.* 47, 1915–1927 (2001)
23. Stankovic, I., Kroger, M., Hess, S.: Recognition and analysis of local structure in polycrystalline configurations. *Comp. Physics Com.* 145, 371–384 (2002)
24. Wen, Z., Li, M., Li, Y., Guo, Y., Wang, K.: Delaunay triangulation with partial least squares projection to latent structures: a model for G-protein coupled receptors classification and fast structure recognition. *Amino Acids* (2006)
25. Xu, L., Krzyżak, A., Yuille, A.: On Radial Basis Function Nets and Kernel Regression: Statistical Consistency, Convergence Rates and Receptive Field Size. *Neural Networks* 4, 609–628 (1994)
26. Ye, Y., Godzik, A.: Multiple flexible structure alignment using partial order graphs. *Bioinformatics* 21(10), 2362–2369 (2005)
27. Zhong, P., Wang, R.: Using Combination of Statistical Models and Multilevel Structural Information for Detecting Urban Areas From a Single Gray-Level Image. *IEEE Trans. Geoscience and Remote Sensing* 45, 1469–1482 (2007)

# Clustering of Data and Nearest Neighbors Search for Pattern Recognition with Dimensionality Reduction Using Random Projections

Ewa Skubalska-Rafajłowicz

Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50–370 Wrocław, Poland  
ewa.rafajlowicz@pwr.wroc.pl

**Abstract.** The dimensionality and the amount of data that need to be processed when intensive data streams are classified may occur prohibitively large. The aim of this paper is to analyze Johnson-Lindenstrauss type random projections as an approach to dimensionality reduction in pattern classification based on  $K$ -nearest neighbors search. We show that in-class data clustering allows us to retain accuracy recognition rates obtained in the original high-dimensional space also after transformation to a lower dimension.

## 1 Introduction

In this paper we develop and examine some new properties of the random projections when applied to pattern recognition algorithms based on nearest neighbors search.

The nearest neighbors search problem is of major and growing importance to a variety of applications such as data compression, statistical data analysis, pattern recognition, data mining, etc [3]. In many applications (image or video processing, analysis of data from sensors arrays, CCD and CMOS cameras, speech recordings and spectral analysis) the dimensionality of the feature space can attain hundreds or even several thousands.

Several dimension reduction techniques, linear and non-linear, have been proposed in many fields, including principal components analysis, projection pursuit, principal curves, independent component analysis, self-organizing maps, space-filling curves [14], neural network implementations of some statistical models and random linear projections, among others. It is reported in the literature that random projections work better than other simplistic dimension reduction schemes [4] when original dimensionality of the data is very large.

In the random projection method, the original high-dimensional observations are projected onto a lower-dimensional space using a suitably scaled random matrix with independent, typically, normally distributed entries.

Random projections have been found to be a computationally efficient, yet sufficiently accurate method for dimensionality reduction. Promising experimental results are reported in many papers [3], [5], [7], [13], [15]. There has been an increasing interest in avoiding the curse of dimensionality by resorting to approximate nearest neighbor searching [8], [9] and clustering methods [3], [4]. However, in pattern recognition methods based on voting among  $K$ -nearest neighbors, the true class-membership of the  $K$ -nearest neighbors is of major importance. Thus, we propose the method of verifying when the nearest neighbor found in lower-dimensional space (space obtained by random projections) is with high probability the true nearest neighbor (the nearest neighbor in the original high-dimensional space). On this bases we propose the adaptive method of neighbors number selection by voting among neighbors. Furthermore, we examine the role of data compression, showing that in-class data clustering allows us to retain accuracy recognition rates (obtained in the original high-dimensional space) after projection to lower-dimensional space.

## 2 Johnson-Lindenstrauss Lemma and Linear Random Projections

In random projections, we can estimate the original pairwise Euclidean distances directly using the corresponding Euclidean distances in smaller dimension. Furthermore, the Johnson-Lindenstrauss lemma [11], [6] provides the performance guarantee.

The Johnson-Lindenstrauss Lemma can be formulated as follows: A  $d$ -dimensional point set  $C$  can be embedded into a  $k$ -dimensional space, where  $k$  is logarithmic in cardinality of  $C$  and independent of  $d$ . Distances between points in  $C$  are preserved within a factor  $(1 \pm \varepsilon)$ , i.e., there exists a mapping  $F : R^d \rightarrow R^k$  such that for all  $x, y \in C$

$$(1 - \varepsilon)\|x - y\|^2 \leq \|F(x) - F(y)\|^2 \leq (1 + \varepsilon)\|x - y\|^2. \quad (1)$$

A random projection from  $d$  dimensions to  $k$  dimensions is a linear transformation represented by a  $d \times k$  matrix  $S \in R^{k \times d}$  - a matrix whose entries are i.i.d. samples of some random variable. Let  $u \in R^d$  be an original data point. The projected data is a point in  $k$ -dimensional space ( $R^k$ ) given by  $v = S.u$

We can estimate the distance in  $d$ -dimensional space  $\|u_m - u_l\|^2$  from the sample squared distances as follows:

$$\hat{D}^2 = \frac{1}{k} \sum_{j=1}^k (v_{mj} - v_{lj})^2 = \|v_m - v_l\|^2. \quad (2)$$

Note that  $\hat{D}^2$  is estimated using the elements of projected vectors and every projection contains some information about the length of the projected vector  $(u_m, u_l)$ .



It is easy to see that  $E\{\hat{D}^2\} = D^2$ , while its variance, decreases to zero as  $k \rightarrow \infty$ , since  $var(\hat{D}^2) = \frac{2}{k}D^4$ .

Using chi-squared tail Chernoff bounds (see [6] for details) we can obtain the bound on the probability when the relative error exceeds  $\varepsilon$  ( $1 > \varepsilon > 0$ )

$$Pr \left\{ \frac{|\hat{D}^2 - D^2|}{D^2} \geq \varepsilon \right\} \leq 2 \exp \left( -\frac{k}{4}\varepsilon^2 + \frac{k}{6}\varepsilon^3 \right). \tag{3}$$

The Bonferroni union bound leads to a very rough (too conservative) approximation of  $k$ , because it ignores the correlations. In practice, one can use much smaller values of  $k$  with acceptable distance distortion (see [3], [5], [7], [12], [13], [15] among others).

**Table 1.** Comparison of M-NN method and probable M-NN method for two Gaussians problems

| $M$ | $k$  | Example A |           | Example B |             |      | Example C |   |      |
|-----|------|-----------|-----------|-----------|-------------|------|-----------|---|------|
|     |      | M-NN      | P M-NN    | M-NN      | P           | M-NN | M-NN      | P | M-NN |
| 1   | 1000 | 1.00      | 1.0-0.99  | 1.00      | 0.94 - 0.74 | 0.73 | 0.69-0.49 |   |      |
| 1   | 300  | 1.0       | 1.0-0.96  | 1.00      | 0.83-0.69   | 0.88 | 0.63-0.48 |   |      |
| 1   | 100  | 1.0       | 0.95-0.80 | 1.00      | 0.76-0.56   | 0.74 | 0.63-0.45 |   |      |
| 10  | 1000 | 1.0       | 1.0-0.99  | 1.00      | 1.0-0.94    | 0.88 | 0.72-0.54 |   |      |
| 10  | 300  | 1.0       | 1.0-0.99  | 1.00      | 0.95-0.78   | 0.87 | 0.65-0.48 |   |      |
| 10  | 100  | 1.0       | 1.0-0.94  | 1.00      | 0.84-0.57   | 0.87 | 0.71-0.46 |   |      |

**Table 2.** Accuracy rates using PM-NN method for AM problem (left table) and BM (right table) for two mixtures of Gaussian problems

| $N$ | $M$ | $k$ | Example AM |           | $N$ | $M$ | $k$  | Example BM |       |           |
|-----|-----|-----|------------|-----------|-----|-----|------|------------|-------|-----------|
|     |     |     | mean       | range     |     |     |      | M-NN       | mean  | range     |
| 10  | 1   | 100 | 0.87       | 0.9-0.79  | 100 | 1   | 100  | 0.932      | 0.572 | 0.69-0.48 |
| 10  | 3   | 100 | 0.93       | 0.97-0.81 | 100 | 3   | 100  | 0.975      | 0.577 | 0.64-0.45 |
| 10  | 5   | 100 | 0.93       | 0.97-0.87 | 100 | 5   | 100  | 0.99       | 0.579 | 0.66-0.47 |
| 100 | 1   | 100 | 0.944      | 0.99-0.91 | 100 | 1   | 300  | 0.930      | 0.61  | 0.70-0.52 |
| 100 | 3   | 100 | 0.971      | 1.0-0.93  | 100 | 3   | 300  | 0.973      | 0.66  | 0.72-0.55 |
| 100 | 5   | 100 | 0.98       | 1.0-0.94  | 100 | 5   | 300  | 0.986      | 0.67  | 0.72-0.61 |
|     |     |     |            |           | 100 | 1   | 1000 | 0.944      | 0.688 | 0.77-0.60 |
|     |     |     |            |           | 100 | 3   | 1000 | 0.979      | 0.735 | 0.83-0.65 |
|     |     |     |            |           | 100 | 5   | 1000 | 0.988      | 0.765 | 0.85-0.71 |

**Table 3.** Accuracy rates using probable M-NN method for AM problem (left table) and BM problem (right table) with adaptively chosen M (two mixtures of Gaussian problems)

| $\varepsilon$ | $M \geq$ | $k$ | Example AM |           |
|---------------|----------|-----|------------|-----------|
|               |          |     | mean       | range     |
| 0.02          | 1        | 100 | 0.964      | 0.99-0.89 |
| 0.01          | 1        | 100 | 0.960      | 0.99-0.91 |
| 0.005         | 1        | 100 | 0.945      | 0.95-0.89 |
| 0.01          | 3        | 100 | 0.975      | 1.00-0.93 |
| 0.01          | 5        | 100 | 0.97       | 1.00-0.92 |

| $\varepsilon$ | $M \geq$ | $k$  | Example BM |       |            |
|---------------|----------|------|------------|-------|------------|
|               |          |      | M-NN mean  | range |            |
| 0.01          | 1        | 100  | 0.957      | 0.585 | 0.697-0.47 |
| 0.1           | 1        | 100  | 0.957      | 0.578 | 0.67-0.48  |
| 0.01          | 3        | 100  | 0.99       | 0.595 | 0.67-0.48  |
| 0.01          | 5        | 100  | 0.993      | 0.615 | 0.73-0.49  |
| 0.01          | 1        | 300  | 0.973      | 0.655 | 0.73-0.60  |
| 0.01          | 3        | 300  | 0.986      | 0.666 | 0.75-0.61  |
| 0.01          | 5        | 300  | 0.944      | 0.684 | 0.77-0.57  |
| 0.01          | 1        | 1000 | 0.979      | 0.715 | 0.72-0.67  |
| 0.01          | 3        | 1000 | 0.988      | 0.762 | 0.85-0.68  |
| 0.01          | 5        | 1000 | 0.998      | 0.789 | 0.86-0.61  |

**Table 4.** Accuracy rates obtained using probable M-NN method for AM problem (left table) and BM problem (right table) with centers of Gaussian as class prototypes

| $\varepsilon$ | $M \geq$ | $k$ | Example AM |           |
|---------------|----------|-----|------------|-----------|
|               |          |     | mean       | range     |
| 0.01          | 1        | 100 | 0.996      | 1.00-0.98 |
| 0.01          | 3        | 100 | 0.997      | 1.00-0.98 |
| 0.01          | 5        | 100 | 0.996      | 1.00-0.98 |
| 0.01          | 1        | 300 | 1.00       | 1.00-1.00 |
| 0.01          | 3        | 300 | 1.00       | 1.00-1.00 |
| 0.01          | 5        | 300 | 1.00       | 1.00-1.00 |

| $\varepsilon$ | $M \geq$ | $k$  | Example BM |           |
|---------------|----------|------|------------|-----------|
|               |          |      | mean       | range     |
| 0.01          | 1        | 100  | 0.776      | 0.84-0.68 |
| 0.01          | 3        | 100  | 0.802      | 0.85-0.74 |
| 0.01          | 5        | 100  | 0.804      | 0.86-0.74 |
| 0.01          | 1        | 300  | 0.877      | 0.92-0.84 |
| 0.01          | 3        | 300  | 0.915      | 0.96-0.84 |
| 0.01          | 5        | 300  | 0.912      | 0.95-0.85 |
| 0.01          | 1        | 1000 | 0.985      | 1.00-0.97 |
| 0.01          | 3        | 1000 | 0.992      | 1.00-0.97 |
| 0.01          | 5        | 1000 | 0.970      | 1.00-0.94 |

**Table 5.** Accuracy rates obtained using probable M-NN method for AM problem (left table) and BM problem (right table) with class means as prototypes

| $M \geq$ | $k$  | Example AM |           |
|----------|------|------------|-----------|
|          |      | mean       | range     |
| 1        | 10   | 0.809      | 0.92-0.76 |
| 1        | 50   | 0.971      | 1.00-0.93 |
| 1        | 100  | 0.995      | 1.00-0.98 |
| 1        | 300  | 1.00       | 1.00-0.99 |
| 1        | 1000 | 1.00       | 1.00-0.99 |

| $M$ | $k$  | Example BM |       |           |
|-----|------|------------|-------|-----------|
|     |      | 1-NN mean  | range |           |
| 1   | 100  | 1.00       | 0.796 | 0.84-0.73 |
| 1   | 300  | 1.00       | 0.89  | 0.95-0.83 |
| 1   | 1000 | 1.00       | 0.977 | 0.99-0.95 |

### 3 Probable Nearest Neighbors - Preservation of Closeness Relation under Space Embedding Using Normal Random Projections

Let  $x \in X$  be a given query point. The nearest neighbors search problem is defined as follows. Given a set of  $M$  points  $C = \{c_1, c_2, \dots, c_M\}$  in a metric space  $(X, d)$ . Order the points in  $C$  in such a way that  $c_{(j)}$  is the  $j$ -th closest point among  $C$  to a query point  $x \in X$ . Without loss of generality we can assume that  $D_i(x) \leq D_{i+1}(x)$ ,  $i = 1, \dots, M - 1$ , where  $D_i(x) = d(c_i, x)$ . Thus  $D_i(x)$  is the  $i$ -th nearest neighbor of  $x$  among the points from set  $C$ ,  $|C| = M$ .

According to Johnson-Lindenstrauss lemma

$$Pr\{1 - \varepsilon \leq \frac{\hat{D}_i^2(x)}{D_i^2(x)} \leq 1 + \varepsilon, i = 1, \dots, M\} \geq 1 - 2M \exp(-\frac{k}{4}\varepsilon^2 + \frac{k}{6}\varepsilon^3). \quad (4)$$

It is easy to see that with probability exceeding the right side of [4](#) we have the following result. If  $\varepsilon \leq \frac{D_j^2(x) - D_i^2(x)}{D_j^2(x) + D_i^2(x)}$  then

$$(1 - \varepsilon)D_i^2(x) \leq \hat{D}_i^2(x) \leq \hat{D}_j^2(x) \leq (1 + \varepsilon)D_j^2(x), \quad (5)$$

for any  $i, j = 1, \dots, M$  and  $i < j$ . This means, that if squared distances from  $x$  to neighbors  $c_i$  and  $c_j$  differ at least  $\varepsilon(D_j^2(x) + D_i^2(x))$ , then  $\hat{D}_i^2(x) \leq \hat{D}_j^2(x)$ . It is obvious, that we want to know when we can conclude from  $\hat{D}_i^2(x) \leq \hat{D}_j^2(x)$  that the original squared distances are in the same relation:  $D_i^2(x) \leq D_j^2(x)$  with probability at least  $1 - \delta$ ,  $\delta \in (0, 1)$ .

Note that for every  $i = 1, \dots, M$

$$Pr\{\frac{\hat{D}_i^2(x)}{D_i^2(x)} \geq 1 + \varepsilon\} = Pr\{\frac{D_i^2(x)}{\hat{D}_i^2(x)} \leq \frac{1}{1 + \varepsilon}\} \geq Pr\{\frac{D_i^2(x)}{\hat{D}_i^2(x)} \leq 1 - \varepsilon\}.$$

Similarly,

$$Pr\left\{\frac{\hat{D}_i^2(x)}{D_i^2(x)} \leq 1 - \varepsilon\right\} = Pr\left\{\frac{D_i^2(x)}{\hat{D}_i^2(x)} \geq \frac{1}{1 - \varepsilon}\right\} \geq Pr\left\{\frac{D_i^2(x)}{\hat{D}_i^2(x)} \geq 1 + 2\varepsilon\right\},$$

provided that  $\varepsilon \leq 1/2$ .

Thus,

$$Pr\{1 - \varepsilon \leq \frac{D_i^2(x)}{\hat{D}_i^2(x)} \leq 1 + 2\varepsilon, i = 1, \dots, M\} \geq$$

$$Pr\{1 - \varepsilon \leq \frac{\hat{D}_i^2(x)}{D_i^2(x)} \leq 1 + \varepsilon, i = 1, \dots, M\} \geq 1 - 2M \exp(-\frac{k}{4}\varepsilon^2 + \frac{k}{6}\varepsilon^3), \varepsilon \in (0, 1/2].$$

So, if  $\hat{D}_i^2(x) < \hat{D}_j^2(x)$ , and the relative distance between two neighbors is

$$\frac{\hat{D}_j^2(x) - \hat{D}_i^2(x)}{\hat{D}_j^2(x) + 2\hat{D}_i^2(x)} \geq \varepsilon, \quad (6)$$

then with probability exceeding  $1 - 2M \exp(-\frac{k}{4}\varepsilon^2 + \frac{k}{6}\varepsilon^3)$  we obtain  $D_j^2(x) \geq (1 - \varepsilon)\hat{D}_j^2(x) \geq (1 + 2\varepsilon)\hat{D}_i^2(x) \geq D_i^2(x)$ , for any  $i, j \in \{1, \dots, M\}$ .

Now we can easily prove the following theorem.

**Theorem 1.** Let  $\hat{D}_1(x) < \dots < \hat{D}_M(x)$ . Choose  $\delta > 0$  and  $\varepsilon \in (0, 1/2)$  such that  $\varepsilon < \frac{\hat{D}_j^2(x) - \hat{D}_i^2(x)}{\hat{D}_j^2(x) + 2\hat{D}_i^2(x)}$ ,  $i, j = 1, \dots, M$ ,  $i < j$ . If the dimensionality after reduction  $k$  is selected in such a way that

$$k \geq 4 \frac{\log 2 + \log M - \log \delta}{\varepsilon^2 - \frac{2}{3}\varepsilon^3}, \quad (7)$$

then the probability that for any pair  $(i, j)$ ,  $D_i(x) > D_j(x)$  ( $i, j = 1, \dots, M, i < j$ ) is smaller than  $\delta$ .

Note that this theorem is formulated for only one query point. However, as in the case of the probabilistic version of Johnson-Lindenstrauss theorem, Theorem [1](#) is also too restrictive for practical purposes. Nevertheless, we can use inequality [\(6\)](#) for adaptive selection of a number of nearest neighbors. We propose the following algorithm:

**Adaptive selection of number of neighbors:** Let  $x$  be a query point. Assume that we have chosen  $M$  nearest neighbors of  $x$  from a given prototype set  $P$  (consisting of  $N \gg M$  elements). Choose  $\varepsilon > 0$ .

Step 1. Find  $(M + 1)$ -th nearest neighbor of  $x$  from set  $P$  and check inequality [\(6\)](#) for  $i = M$  and  $j = M + 1$ .

Step 2. If [\(6\)](#) fails, set  $M = M + 1$  and go to Step 1. Otherwise, end the procedure.

## 4 Experimental Results

In the first part of the experiments, we have used the following three data sets. Two equiprobable classes described by normal distributions with vector means:  $(0, 0, \dots, 0)$  and  $(m, m, \dots, m)$ , respectively, and the same covariance matrix  $I$ . Dimension of the original data  $d$  is equal to  $d = 10000$  and  $m = 0.5$  (set A with Bayes error close to 0),  $m = 0.25$  (set B with Bayes error also close to 0) and  $m = 0.05$  (set C with Bayes error  $10^{-12}$ ).

We have compared the performance of the  $M$ -NN nearest neighbor method based on data taken from the original space and the probable  $M$ -NN (PM-NN) method based on projected data (using normal LRP).  $M$  denotes the number of neighbors ( $M = 1$  or  $M = 10$ ). The length of a learning sequence  $N$  was fix as 100. Note that it is very small learning sequence in comparison to the dimensionality of the space. The dimension after projection  $k$  was established as 1000, 300 and 100. In each case we run 20 experiments with different random projections. Table 1 contains the accuracy rates (accuracy is 1 minus error rate) for both methods.

Note that in each classification problem the centers of two Gaussians are  $m\sqrt{d}$  apart. It is known, that two Gaussian  $N(\mu_1, I_d)$  and  $N(\mu_2, I_d)$  are  $c$ -separated if  $\|m\mu_1 - \mu_2\| \geq c\sqrt{d}$ . In high dimensional spaces if  $c$  is between 0.5 – 1 then Gaussians have an almost negligible overlap. Thus, only for  $c = 0.05$  (Example C) larger recognition errors occur.

In the second part of the experiments, we concentrate on more complicated problems. Namely, a class-distribution for each of two equiprobable classes consists of a mixture of five Gaussians (with the same covariance matrix  $I$ ) and concentrated around  $(0, 0, \dots, 0)$  and  $(m, m, \dots, m)$ , respectively. The dimension of the original data  $d$  is equal to  $d = 10000$  and  $m = 0.5$  (problem AM – almost negligible overlap),  $m = 0.1$  (problem BM – slightly overlapping classes). In both cases Bayes error is close to 0.000.  $N$  denotes the length of the learning sequence. The results are summarized in Table 2.

The same problems have been further examined using adaptively chosen number of nearest neighbors. The results are given in Table 3. Table 4 contains accuracy rates for decisions made on the basis of distances of query points to  $N = 10$  class prototypes (five for each class). We assume that prototypes are obtained using some kind of cluster formation [7, 13, 12]. For the sake of simplicity and in order to avoid cross-interferences between methods, prototypes are taken as centers of Gaussians forming the mixtures. Similar experiments have been performed for 2 class-prototypes (one for each class). This time, the approximate distribution means  $(0, 0, \dots, 0)$  and  $(m, m, \dots, m)$  form the prototypes set.

## 5 Conclusions

Random projections methods outperform other dimensionality reduction methods from the view point of computational complexity. If data clusters are only slightly mixed, then  $1 - NN$  recognition error grows very rapidly, when a dimension of the projection space decreases, since clusters after projection are

strongly mixed. Thus, using the many neighbors approach with an adaptively chosen number of neighbors allows us to improve the classification efficiency. Furthermore, in many cases, carefully designed class prototypes enable us to retain almost the same level of recognition accuracy even if dimensionality reduction is very large.

In this paper we have analyzed some aspects of Euclidean nearest neighbors classification methods. Inner product based distance measures should be investigated in a similar vein. The results presented in [16], [12] can be a good starting point.

**Acknowledgments.** This work was supported by a grant of the Ministry of Science and Higher Education for 2006-2009.

## References

1. Achlioptas, D.: Database friendly random projections. In: Proc. Principles of Database Systems (PODS), pp. 274–281 (2001)
2. Arriaga, R., Vempala, S.: An algorithmic theory of learning: Robust concepts and random projection. In: Proc. of FOCS, New York, pp. 616–623 (1999)
3. Bertoni, A., Valentini, G.: Random projections for assessing gene expression cluster stability. In: Proceedings IEEE international joint conference on neural networks, vol. 1, pp. 149–154 (2005)
4. Biau, G., Devroye, L., Lugosi, G.: On the Performance of Clustering in Hilbert Spaces. *IEEE Tran. on Information Theory* 54(2), 781–790 (2008)
5. Bingham, E., Mannila, H.: Random projection in dimensionality reduction: Applications to image and text data. In: Proc. Seventh ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD 2001), pp. 245–250 (2001)
6. Dasgupta, S., Gupta, A.: An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Structures and Algorithms* 22(1), 60–65 (2003)
7. Fern, X.Z., Brodley, C.E.: Random projection for high dimensional data clustering: a cluster ensemble approach. In: Proceedings of the 20th international conference on machine learning (ICML 2003), Washington DC, USA (August 2003)
8. Indyk, P., Motwani, R.: Approximate nearest neighbors: Towards removing the curse of dimensionality. In: Proc. of STOC, Dallas, TX, pp. 604–613 (1998)
9. Indyk, P., Naor, A.: Nearest neighbor preserving embeddings. *ACM Transactions on Algorithms (TALG) archive* 3, article no. 31 (2007)
10. Clarkson, K.: Nearest-neighbor searching and metric space dimensions. In: *Nearest-Neighbor Methods for Learning and Vision: Theory and Practice*. MIT Press, Cambridge (2005)
11. Johnson, W.B., Lindenstrauss, J.: Extensions of Lipschitz mapping into Hilbert space. *Contemporary Mathematics* 26, 189–206 (1984)
12. Liu, K., Kargupta, H., Ryan, J.: Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. *IEEE Transactions on Knowledge and Data Engineering* 18, 92–106 (2006)
13. Kaski, S.: Dimensionality reduction by random mapping: fast similarity computation for clustering. In: Proc. IEEE International Joint Conference on Neural Networks, vol. 1, pp. 413–418 (1998)

14. Skubalska-Rafajłowicz, E.: Pattern recognition algorithm based on space-filling curves and orthogonal expansion. *IEEE Trans. on Information Theory* 47, 1915–1927 (2001)
15. Skubalska-Rafajłowicz, E.: Random projection RBF nets for multidimensional density estimation. *International Journal of Applied Mathematics and Computer Science* 18(4), 455–464 (2008)
16. Skubalska-Rafajłowicz E: Neural networks with sigmoidal activation functions – dimension reduction using normal random projection. *Nonlinear Analysis* 71(12), e1255–e1263 (2009)
17. Vempala, S.: *The Random Projection Method*. American Mathematical Society, Providence (2004)

# Noise Detection for Ensemble Methods

Ryszard Szupiluk<sup>1,2</sup>, Piotr Wojewnik<sup>1,2</sup>, and Tomasz Zabkowski<sup>1,3</sup>

<sup>1</sup> Polska Telefonia Cyfrowa Ltd., Al. Jerozolimskie 181, 02-222 Warsaw, Poland

<sup>2</sup> Warsaw School of Economics, Al. Niepodleglosci 162, 02-554 Warsaw, Poland

<sup>3</sup> Warsaw University of Life Sciences, ul. Nowoursynowska 159/34,  
02-787 Warsaw, Poland

{rszupiluk,pwojewnik,tzabkowski}@era.pl

**Abstract.** In this paper we present a novel noisy signal identification method applied in ensemble methods for destructive components classification. Typically two main signal properties like variability and predictability are described by the same second order statistic characteristic. In our approach we postulate to separate measure of the signal internal dependencies and their variability. The validity of the approach is confirmed by the experiment with energy load data.

## 1 Introduction

The problems with noise filtration, reduction and identification are fundamental tasks in signal processing, data analysis or systems modeling [12,20]. This paper presents a novel noisy signal identification and classification approach in ensemble method context. It can be also treated as multivariate filtration postprocessing task where from the set of prediction results we aim to remove common latent noises [16].

The ensemble method is stated as follow. Let us assume we have a set of predictive models. We collect their results together in one multivariate variable  $\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_m]^T$ ,  $\mathbf{X} \in R^{m \times N}$ . Next we assume that each prediction result  $\mathbf{x}_i$  is a linear mixture of the latent components. The latent component can be constructive or destructive for the prediction results. The constructive components  $\hat{\mathbf{s}}_j$  are associated with true predicted value whereas the destructive components  $\tilde{\mathbf{s}}_j$  are responsible for errors. The relation between prediction results and their latent components can be described in matrix form as

$$\mathbf{X} = \mathbf{A}\mathbf{S}, \quad (1)$$

where  $\mathbf{S} = [\hat{\mathbf{s}}_1 \ \hat{\mathbf{s}}_2 \ \dots \ \hat{\mathbf{s}}_k \ \tilde{\mathbf{s}}_{k+1} \ \dots \ \tilde{\mathbf{s}}_m]^T$ ,  $\mathbf{S} \in R^{m \times N}$ ,  $\mathbf{A} \in R^{m \times m}$ , represents the mixing system. The relation (1) means matrix  $\mathbf{X}$  factorisation by the latent components matrix  $\mathbf{S}$  and the mixing matrix  $\mathbf{A}$ . Our aim is to find the latent components and reject the destructive ones (replace them with zero) and next mix the constructive components back to obtain improved prediction results as

$$\hat{\mathbf{X}} = \mathbf{A}\hat{\mathbf{S}} = \mathbf{A}[\hat{\mathbf{s}}_1 \ \hat{\mathbf{s}}_2 \ \dots \ \hat{\mathbf{s}}_k \ \mathbf{0}_{k+1} \ \dots \ \mathbf{0}_m]^T. \quad (2)$$



We face two problems: how to estimate the components, and how to choose the noisy ones. Some solutions to the first problem were proposed by blind signal separation technique. Blind Signal Separation (BSS) methods aim at identification of unknown signals mixed in an unknown system [4,16]. In our context to find the latent variables  $\mathbf{A}$  and  $\mathbf{S}$  we can use a transformation defined by separation matrix  $\mathbf{W} \in R^{m \times m}$ , such that

$$\mathbf{Y} = \mathbf{W}\mathbf{X} = \mathbf{W}\mathbf{A}\mathbf{S} = \mathbf{P}\mathbf{D}\mathbf{S}, \quad (3)$$

where  $\mathbf{P}$  is a permutation matrix and  $\mathbf{D}$  is a diagonal scaling matrix,  $\mathbf{W}$  represents pseudoinverse system to  $\mathbf{A}$  in the sense that relation  $\mathbf{W}\mathbf{A} = \mathbf{P}\mathbf{D}$  holds [4]. The relation (3) means that estimated signals  $\mathbf{S}$  can be rescaled and re-ordered in comparison to original sources. In our case it is not crucial, therefore  $\mathbf{Y} = [\mathbf{y}_1 \mathbf{y}_2 \dots \mathbf{y}_m]^T$  can be treated directly as estimated version of basis components  $\mathbf{S}$ . The process of matrix  $\mathbf{W}$  identification with BSS methods can explore different properties of the data like: independence [3,7], decorrelation [4], sparsity [10], smoothness [4,18], non-negativity [9] etc. Many of those approaches were tested and successfully used in practice.

The open question, still, is proper identification of latent components in  $\mathbf{S}$ . In particular, we eliminate each subset of  $\mathbf{S}$  and check the impact on the final results. Such process of component classification as destructive or constructive is simple and works well but for many components it is time consuming. Therefore, we try to find the general characteristic that can be used to classify the basis latent components. It seems quite natural to associate the destructive components with noises. In next paragraphs we present our approach for noisy signal identification used in above ensemble methods.

## 2 What Is the Random Noise?

The term random noise due to references to such terms like randomness, uncertainty or probability has many definitions, sometimes with deep philosophical interpretations [8,11,13]. In data analysis practice the random signal or noise is when the current values give no precise information about the future values. The most popular example of the noise model is the white noise. The pure white noise typically means that the observations come from the same distribution and are independent from each other [19,20]. We expect that in practise such generated signal should not include any predictable patterns (internal dependencies, correlations, trends) and shouldn't be smooth. It is very convenient case if the analysed model or data include white noise. In practise, the situation is more complex. There are coloured noises with internal dependencies or mixtures of the random noises and deterministic signals [5,14]. The situation is even more difficult for chaotic signals that look like noises but they are generated by deterministic systems [13]. In practice, we don't know what is the noise and how its characteristic looks like. For example, we don't know a priori what input variables improve our prediction results and what input variables worsen prediction results. From modeling perspective some variables can be classified as noises, but

they cannot be treated as noises from statistical point of view. However, being aware of these considerations and ambiguity of noise definition, we will focus on noises that are treated as random signals.

Taking above considerations the random noise detection can be solved in many ways but typically it leads to the analysis of the internal dependencies in signals [12,20]. The standard characteristic investigated in this case is the autocorrelation function or its Fourier transformation called power spectrum [19]. Unfortunately it has some disadvantages like functional form what is difficult for comparison, it appears to be insensitive in some cases and it causes problems with detection of the long memory dependencies due to its exponential decrease [14].

The alternative to autocorrelation function is the Hurst exponent  $H$  and  $R/S$  analysis [6,13,14]. The Hurst exponent can be obtained by dividing signal into parts with  $n$  – observation each and calculate

$$\ln E(R/S) = \ln c + H \ln n, \quad (4)$$

where and  $R$  is the range of the signal,  $S$  is standard deviation,  $n$  is number of observations in each part,  $c$  is some constant, expectation is taken over the parts.

The Hurst exponent can take values from 0 to 1, where for  $H = 0.5$  we have white noise (not necessary Gaussian) and for  $H > 0.5$  we have persistent signal (e.g.  $H = 1$  means pure deterministic signal) and  $H < 0.5$  we have antipersistent signal. It is important to calculate  $H$  only on linear part of the regression identified by individual inspection during analysis [6,13,14].

Taking above consideration we propose a novel approach to noise detection. In our method we separate the volatility measure and internal dependencies measure. For dependency measure we combine autocorrelation analysis and  $R/S$  analysis methodology. For volatility measure we propose the separate smoothness factor.

### 3 Internal Dependencies Measure

Internal dependency analysis aims at investigation how one value of the signal influences other values. Our approach is arranged according to the following requirements:

1. We want to know internal dependencies between different parts of signal specified according to different time domains.
2. The information from 1. should be accessible for different length of the analysed parts.
3. The final synthetic internal dependency value should be easily interpretable and decomposable into the factors that describe dependencies between particular parts of the signal.

To fulfil the above requirements there are several steps to perform in the following procedure.

1. We divide signal  $y$  into  $L$  parts of  $n$  observation each so we have a set of signals  $y_i^{(n)}$ ,  $i = 1 \dots L$ .
2. The signals are stored in one multivariate variable  $\mathbf{Y}^{(n)} = [y_1^{(n)} y_2^{(n)} \dots y_L^{(n)}]$ .
3. We find covariance matrix  $\mathbf{C}^{(n)} = E[y_1^{(n)} y_2^{(n)} \dots y_L^{(n)}]$  and compute

$$\xi(n) = \frac{1}{(n-1)} \left( \frac{\|\mathbf{C}^{(n)}\|_p}{\|diag(\mathbf{C}^{(n)})\|_p} - 1 \right), \tag{5}$$

where

$$\|\mathbf{C}^{(n)}\|_p = \left( \sum_{i=1}^L \sum_{j=1}^L |c_{ij}|^p \right)^{\frac{1}{p}}, \tag{6}$$

is the  $p$ -norm with most popular cases like

- 1-norm is absolute value criterion,
- 2-norm is Frobenius norm,
- $\infty$ - norm is Chebyshev norm.

In the context of our considerations we set the value of  $p$  parameter to 1. The  $diag(\mathbf{C}^{(n)})$  is diagonal matrix with entries  $c_{ii}$ .

4. Make  $K$  iterations with steps 1-3 for different  $n$  to obtain set of  $\xi(n_t)$ ,  $t = 1 \dots K$  and perform the regression  $\xi$  and  $n$ . The simple way to synthesize information from set of  $\xi(n_t)$ ,  $t = 1 \dots K$  is the mean value calculation

$$\xi_C = \frac{1}{K} \sum_{t=1}^K \xi(n_t). \tag{7}$$

The main idea of the proposed measure is to find second order statistical dependencies for different time combinations. It means that we analyze the correlation between parts of the signals close to each other as well as we check long memory effects. The crucial characteristic is given with inspection of the  $\xi_C$  value as the function of the  $n$ . For random noise we expect that such characteristic is flat. Moreover, the introduced characteristic enables the cyclic analysis.

## 4 Variability Measure

It seems intuitive that for data with temporal structure the random noises are not regular or smooth. Unfortunately, the standard variability measure like variance doesn't take into account information from data order. Therefore, for signals with temporal structure we propose a following measure:

$$P(\mathbf{y}) = \frac{\frac{1}{N-1} \sum_{k=2}^N |\mathbf{y}(k) - \mathbf{y}(k-1)|}{\max(\mathbf{y}) - \min(\mathbf{y}) + \delta(\max(\mathbf{y}) - \min(\mathbf{y}))}, \tag{8}$$

where symbol  $\delta(\cdot)$  means a zero indicator -  $\delta(a) = 1$  iff  $a = 0$ , otherwise  $\delta(a) = 0$ ,  $N$  is the number of observations indexed by  $k = 1, \dots, N$ . Measure (8) has simple

interpretation: it is maximal when the changes in each step are equal to the range (maximal change), and is minimal when data are constant. The possible values are from 0 to 1. The indicator  $\delta(\cdot)$  is introduced to avoid dividing by zero. As we expected, the more predictable signal, the less smoothness value.

## 5 Noise Distance and Component Classification

It is an open question how to join optimally the information from signal variability and internal dependencies. From our point of view the both characteristics are equally important. Therefore, we propose the Euclidean distance measure applied to manifold given by signal variability and internal dependencies. For given signal  $\mathbf{y}$  the values of the internal dependency measure and the smoothness factor can be represented in the vector form  $[\xi_C(\mathbf{y}), P(\mathbf{y})] \in \langle 0, 1 \rangle^2$ . Lets observe, the extremely deterministic signal in terms of internal dependency achieves  $\xi_C = 0$ , while in terms of smoothness  $P = 1$ . Therefore, the distance between  $[\xi_C(\mathbf{y}), P(\mathbf{y})]$  and  $[0, 1]$  informs of the signal noisiness.

$$D(\mathbf{y}) = \left\| [\xi_C(\mathbf{y}), 1 - P(\mathbf{y})] \right\|. \tag{9}$$

Let's back to our aggregation problem through noise component elimination. In general, we are looking for such matrix  $\mathbf{W} = [\mathbf{w}_1 \mathbf{w}_2 \dots \mathbf{w}_m]^T$  that for  $\mathbf{y}_i = \mathbf{w}_i \mathbf{X}$  component of the highest noise can be found by minimization of the noise distance

$$\min_{\mathbf{w}} D(\mathbf{w}^T \mathbf{X}) = \min_{\mathbf{w}} \left\| [\xi_C(\mathbf{w}^T \mathbf{X}), 1 - P(\mathbf{w}^T \mathbf{X})] \right\|. \tag{10}$$

Unfortunately, due to complexity of  $\xi_C$  it will be inefficient to solve the optimization problem (10) both with direction and genetic optimization techniques. Therefore, we propose the utilization of the standard decomposition and separation procedures to identify the possible separation matrices  $\mathbf{W}^{(i)} = [\mathbf{w}_1^{(i)} \mathbf{w}_2^{(i)} \dots \mathbf{w}_m^{(i)}]^T, i = 1 \dots k$ . Next we suggest calculation of the noise distance (9) for each possible component identified by  $\mathbf{w}_j^{(i)}, i = 1 \dots k, j = 1 \dots m$ . It means that the problem (10) can be expressed as

$$\min_{i=1 \dots k \ j=1 \dots m} D((\mathbf{w}_j^{(i)})^T \mathbf{X}) = \min_{i=1 \dots k \ j=1 \dots m} \left\| [\xi_C((\mathbf{w}_j^{(i)})^T \mathbf{X}), 1 - P((\mathbf{w}_j^{(i)})^T \mathbf{X})] \right\|. \tag{11}$$

## 6 Practical Experiment

To verify the model aggregation method with the noise elimination we perform an experiment with the Polish energy load data. The problem is to predict the hourly energy consumption in 24 hours basing on the last 48 hourly loads, weekday and holiday indicator - ca. 105k observations. We perform a 100 runs of simulation. In one simulation the date is randomly chosen and 62 days before (1500 obs.) are used for learning and 30 days after (720 obs.) – for testing.

Nine multilayer perceptrons are trained and their quality is measured with mean absolute percentage error, MAPE.

The models results on learning data are decomposed using: PCA, Cholesky factorization, JADE and Pearson ICA [3,4]. The decomposition matrices are applied to models on testing data. The resulting signals are measured for  $[\xi_C(\cdot), P(\cdot)]$  and the Euclidean distance from reference characteristics  $[0, 1]$  is calculated. After elimination of a particular signal we measure MAPE of improved models. For each signal we choose the best model. Each point in Fig 1 presents one signal and appropriate model in terms of noise distance and MAPE value. We can observe dependency between the noise distance of the signal and the quality of the model obtained after signal elimination.

In Fig 2A we can observe the quality of primary models. In Fig 2B we can observe the improvement value after PCA component identification and elimination of the most noisy ones. The interquartile range for the models 4-9 is below the zero-line. Therefore, we can conclude that elimination of the noisy signals leads to reduction in MAPE value.

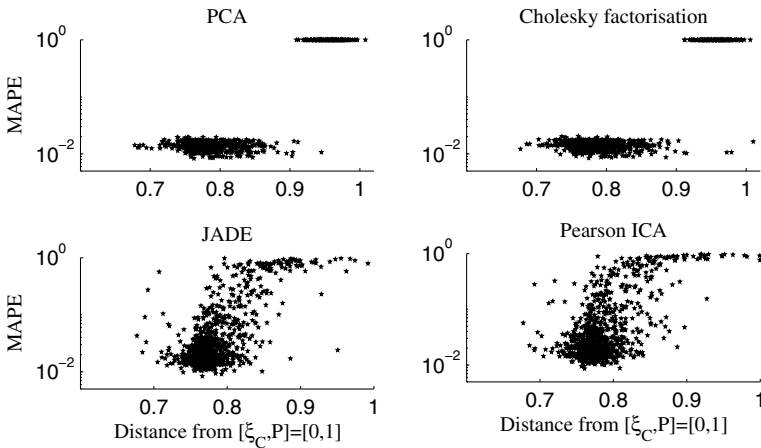


Fig. 1. The quality of the model obtained after elimination of the most noisy signal according to noise distance

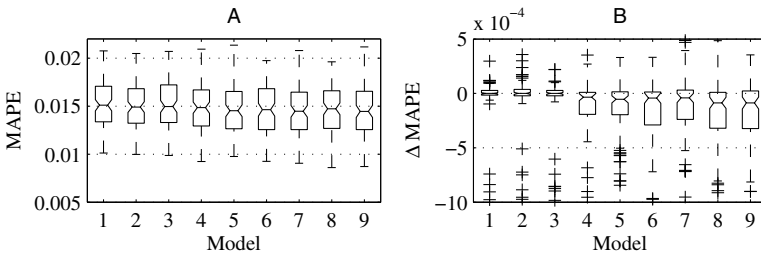


Fig. 2. Quality of primary models and improvement value

## 7 Conclusions

The article presents an approach for the random noise detection applied to the latent components classification in ensemble method context. In our opinion it is difficult to make accuracy measure for signal variability and internal dependencies in one characteristic like autocorrelation or  $R/S$  analysis.

For non-white random noises it seems more adequate to distinguish above signal features. The characteristics proposed in the article are quite natural and associated with our intuitive meaning about the noise. On the other hand they contain all the information given by standard second order statistical measures. Of course, there are many details left for future analysis like form of distance measure. The Euclidean norm is the simplest choice, but it opens the discussion about importance and mutual relation between variability and internal dependency.

However, on the model aggregation background the noise identification method works very well in presented form. The practical experiment on the energy load data confirmed the validity of the proposed approach.

## References

1. Breiman, L.: Bagging predictors. *Machine Learning* 24, 123–140 (1996)
2. Bishop, C.M.: *Neural networks for pattern recognition*. Oxford Univ. Press, Oxford (1996)
3. Cardoso, J.F.: High-order contrasts for independent component analysis. *Neural Computation* 11(1), 157–192 (1999)
4. Cichocki, A., Amari, S.: *Adaptive Blind Signal and Image Processing*. John Wiley, Chichester (2002)
5. Goransson, B.: Direction finding in the presence of spatially correlated noise fields. In: *Proc. European Signal Processing Conf.* (1994)
6. Hurst, H.E.: Long term storage capacity of reservoirs. *Trans. Am. Soc. Civil Engineers* 116, 770–808 (1951)
7. Hyvarinen, A., Karhunen, J., Oja, E.: *Independent Component Analysis*. John Wiley, Chichester (2001)
8. Jaynes, E.T.: *Probability Theory: The Logic of Science*. Cambridge University Press, Cambridge (2003)
9. Lee, D.D., Seung, H.S.: Learning of the parts of objects by non-negative matrix factorization. *Nature* 401 (1999)
10. Li, Y., Cichocki, A., Amari, S.: Sparse component analysis for blind source separation with less sensors than sources. In: *Fourth Int. Symp. on ICA and Blind Signal Separation*, Nara, Japan, pp. 89–94 (2003)
11. Lindley, D.V.: The probability approach to the treatment of uncertainty in artificial intelligence and expert systems. *Statistical Science* 2, 17–24 (1987)
12. MacDonough, R.N., Whalen, A.D.: *Detection of signals in noise*, 2nd edn. Academic Press, San Diego (1995)
13. Mandelbrot, B.: *Multifractals and 1/f noise*. Springer, Heidelberg (1997)
14. Peters, E.: *Fractal market analysis*. John Wiley and Son, Chichester (1996)
15. Samorodnitskij, G., Taqqu, M.S.: *Stable non-Gaussian random processes: stochastic models with infinite variance*. Chapman and Hall, New York (1994)

16. Stone, J.V.: Blind Source Separation Using Temporal Predictability. *Neural Computation* 13(7), 1559–1574 (2001)
17. Szupiluk, R., Wojewnik, P., Zabkowski, T.: Model Improvement by the Statistical Decomposition. In: Rutkowski, L., Siekmann, J.H., Tadeusiewicz, R., Zadeh, L.A. (eds.) *ICAISC 2004. LNCS (LNAI)*, vol. 3070, pp. 1199–1204. Springer, Heidelberg (2004)
18. Szupiluk, R., Wojewnik, P., Zabkowski, T.: Prediction Improvement via Smooth Component Analysis and Neural Network Mixing. In: Kollias, S.D., Stafylopatis, A., Duch, W., Oja, E. (eds.) *ICANN 2006. LNCS*, vol. 4132, pp. 133–140. Springer, Heidelberg (2006)
19. Therrien, C.W.: *Discrete Random Signals and Statistical Signal Processing*. Prentice-Hall, New Jersey (1992)
20. Vaseghi, S.V.: *Advanced signal processing and digital noise reduction*. John Wiley and Sons, Chichester (1997)

# Divergence Based Online Learning in Vector Quantization

Thomas Villmann<sup>1</sup>, Sven Haase<sup>1</sup>, Frank-Michael Schleif<sup>2</sup>,  
and Barbara Hammer<sup>2</sup>

<sup>1</sup> University of Applied Sciences Mittweida

Department of Mathematics/Natural Sciences/Informatics

<sup>2</sup> Clausthal University of Technology, Institute of Computer Science,  
Clausthal-Zellerfeld, Germany

**Abstract.** We propose the utilization of divergences in gradient descent learning of supervised and unsupervised vector quantization as an alternative for the squared Euclidean distance. The approach is based on the determination of the Fréchet-derivatives for the divergences, which can be immediately plugged into the online-learning rules.

**Keywords:** vector quantization, divergence based learning, information theory.

## 1 Introduction

Prototype based vector quantization for clustering and classification usually is based on the Euclidean distance. Prominent widely ranged models are  $k$ -means [19], the self-organizing map (SOM, [15]) or the neural gas vector quantizer (NG, [20]) for unsupervised data modeling and visualization. These methods have in common that the dissimilarity of data is measured by the Euclidean norm, the derivative of which determines the respective adaptation scheme. In the last years these methods were extended to deal with non-standard metrics like functional norms [17, 29], kernelized metrics [22, 12] or general dissimilarities [7].

From the pioneering work of ZADOR it is clear that vector quantization is closely related to information theory [30]. This idea was been further established incorporating information theoretic concepts directly into vector quantization learning paradigms [21, 26]. In particular, J. PRINCIPE ET AL. proposed the aspect of mutual information maximum learning in vector quantization replacing the averaged quantization error by the mutual information between the data distribution and the distribution of prototypes [18]. Recent information theoretic approaches focus on the minimization of the of the averaged divergence between the data and their representing prototypes [2, 13]. However, these approaches are based on the batch mode learning utilizing the expectation-maximization methodology. In this scheme, all data have to be available at hand, which is not assumed in the online learning mode of these algorithms.

Online vector quantization usually follows a stochastic gradient descent learning of the underlying cost function, which itself is based on the measurement of



the dissimilarities between data and prototypes. Thereby, the learning procedure incorporates the derivative of the dissimilarity measure. In this contribution we propose the application of divergences as dissimilarity measure also for online learning. We show that utilizing the concept of *Fréchet-derivatives* it is possible to transfer the batch mode ideas also to online learning. We exemplarily show this for the widely used self-organizing map while generalization to other methods is straight forward.

The paper is organized as follow: first we reconsider the basics of online vector quantization. Thereafter we show, how divergence learning can be plugged in. We demonstrate the effect of the new learning approach for an application in remote sensing data analysis.

## 2 Prototype Based Vector Quantization

Prototype based vector quantization (VQ) is a mapping of data  $\mathbf{v} \in V \subseteq \mathbb{R}^n$ , distributed according to the data density  $P$ , onto a set  $\mathbf{W} = \{\mathbf{w}_{\mathbf{r}} \in \mathbb{R}^n\}_{\mathbf{r} \in A}$  of prototypes. The set  $A$  is an appropriate index set,  $D$  is the input dimension and  $N = \#A$  the number of prototypes. The aim of vector quantization during learning is to distribute the prototypes in the data space such that they represent the data as good as possible. This property is judged by quantization error

$$E_{VQ} = \int \xi(\mathbf{v}, \mathbf{w}_{\mathbf{s}(\mathbf{v})}) P(\mathbf{v}) d\mathbf{v}$$

based on the dissimilarity measure  $\xi$  and

$$\mathbf{s}(\mathbf{v}) = \underset{\mathbf{r} \in A}{\operatorname{argmin}} [\xi(\mathbf{v}, \mathbf{w}_{\mathbf{r}})] \tag{1}$$

being the best matching unit (winner). Hence, the quantization error can be seen as the *expectation value* for the mapping error in the winner determination.

For the NG the above cost function is modified to

$$E_{NG} = \frac{1}{2C(\lambda)} \sum_{\mathbf{r}} \int P(\mathbf{v}) h_{\sigma}(\mathbf{r}) \xi(\mathbf{v}, \mathbf{w}_{\mathbf{r}}) d\mathbf{v}$$

with the so-called neighborhood function  $h_{\sigma}(\mathbf{r}) = \exp\left(\frac{-\operatorname{rank}(\mathbf{r})}{2\sigma^2}\right)$  and is the rank function counting the number of prototypes  $\mathbf{r}'$  for which  $\xi(\mathbf{v}, \mathbf{w}_{\mathbf{r}'}) \leq \xi(\mathbf{v}, \mathbf{w}_{\mathbf{r}})$  holds [20]. For SOM a cost function can be defined by

$$E_{SOM} = \int P(\mathbf{v}) \sum_{\mathbf{r}} \delta_{\mathbf{r}}^{\mathbf{s}(\mathbf{v})} \cdot le(\mathbf{v}, \mathbf{r}) d\mathbf{v}$$

with local errors  $le(\mathbf{v}, \mathbf{r}) = \sum_{\mathbf{r}'} h_{\sigma}(\mathbf{r}, \mathbf{r}') \xi(\mathbf{v}, \mathbf{w}_{\mathbf{r}'})$  and  $\delta_{\mathbf{r}}^{\mathbf{s}(\mathbf{v})}$  is the Kronecker-symbol using HESKES' variant [11]. Here, the neighborhood function  $h_{\sigma}(\mathbf{r}, \mathbf{r}') = \exp\left(\frac{-\xi_A(\mathbf{r}, \mathbf{r}')}{2\sigma^2}\right)$  is the distance measured in the index set  $A$ . For SOMs, the index set  $A$  is equipped with a topological order usually taken as regular

low-dimensional grid. However, compared with standard SOM the winning rule in Heskens-SOM is slightly modified:

$$\mathbf{s}(\mathbf{v}) = \operatorname{argmin}_{\mathbf{r} \in A} [l_e(\mathbf{v}, \mathbf{r})]. \tag{2}$$

For both algorithms learning is realized as a stochastic gradient with respect to the prototypes  $\mathbf{w}_r$ :

$$\Delta \mathbf{w}_r = -\varepsilon \frac{\partial E_{NG/SOM}}{\partial \mathbf{w}_r} \tag{3}$$

which contains as an essential ingredients the derivative  $\frac{\partial \xi(\mathbf{v}, \mathbf{w}_r)}{\partial \mathbf{w}_r}$ .

As mentioned above, frequently the quadratic Euclidean norm is used for  $\xi$ . In the following we will replace it by divergence measures. Yet, the strategy is straight forward: If the derivative is determined it may be plugged into each gradient based vector quantization scheme including SOMs, NG but also supervised approaches like generalized learning vector quantization (GLVQ) [25].

### 3 Divergences as Dissimilarities and Derivatives Thereof

Divergences estimate the similarity between density functions or positive measure functions in a more general sense<sup>1</sup>. In information theory they are related mutual information [16]. According to the classification given in CICHOCKI ET AL. one can distinguish at least *three* main classes of divergences, the *Bregman*-divergences, the *Csiszár's f*-divergences and the  $\gamma$ -divergences [5].

Let  $\Phi$  be a strictly convex real-valued function with the domain  $\mathcal{L}$  (the Lebesgue-integrable functions). Further,  $\Phi$  is assumed to be twice continuously Fréchet-differentiable [14]. Further, we suppose  $p$  and  $\rho$  to be positive measures with  $p(x) \leq 1$  and  $\rho(x) \leq 1$  not necessarily normalized. In the latter case we explicitly refer to these as probability densities. Bregman divergences are defined as  $D_{\Phi}^B : \mathcal{L} \times \mathcal{L} \rightarrow \mathbb{R}^+$  with

$$D_{\Phi}^B(p||\rho) = \Phi(p) - \Phi(\rho) - \frac{\delta\Phi(\rho)}{\delta\rho}(p - \rho) \tag{4}$$

whereby  $\frac{\delta\Phi(\rho)}{\delta\rho}$  is the Fréchet-derivative of  $\Phi$  with respect to  $\rho$ . An important subset are the  $\beta$ -divergences

$$D_{\beta}(p||\rho) = \int p \cdot \frac{p^{\beta-1} - \rho^{\beta-1}}{\beta - 1} d\mathbf{x} - \int \frac{p^{\beta} - \rho^{\beta}}{\beta} d\mathbf{x} \tag{5}$$

with  $\beta \neq 1$  and  $\beta \neq 0$ . In the limit  $\beta \rightarrow 1$  the divergence  $D_{\beta}(p, \rho)$  becomes the generalized Kullback-Leibler-divergence

$$D_{GKL}(p||\rho) = \int p(\mathbf{x}) \log \left( \frac{p(\mathbf{x})}{\rho(\mathbf{x})} \right) d\mathbf{x} - \int p(\mathbf{x}) - \rho(\mathbf{x}) d\mathbf{x}. \tag{6}$$

---

<sup>1</sup> Density functions are functions  $f(x) \geq 0$  with  $\int_{\Omega} f(x) dx = 1$  with  $\Omega$  being the measure space. For positive measure functions the normalization condition is dropped.

Csiszár’s  $f$ -divergences are generated by a *convex* function  $f : [0, \infty) \rightarrow \mathbb{R}$  with  $f(1) = 0$  (without loss of generality) as

$$D_f(p||\rho) = \int \rho(\mathbf{x}) \cdot f\left(\frac{p(\mathbf{x})}{\rho(\mathbf{x})}\right) d\mathbf{x} \tag{7}$$

with the definitions  $0 \cdot f\left(\frac{0}{0}\right) = 0$ ,  $0 \cdot f\left(\frac{a}{0}\right) = \lim_{x \rightarrow 0} x \cdot f\left(\frac{a}{x}\right) = \lim_{u \rightarrow \infty} a \cdot \frac{f(u)}{u}$  [8]. Again, we can identify an important subset the so-called  $\alpha$ -divergences [5]:

$$D_\alpha(p||\rho) = \frac{1}{\alpha(\alpha - 1)} \int [p^\alpha \rho^{1-\alpha} - \alpha \cdot p + (\alpha - 1)\rho] d\mathbf{x} \tag{8}$$

with the generating  $f$ -function

$$f(u) = u \frac{(u^{\alpha-1} - 1)}{\alpha^2 - \alpha} + \frac{1 - u}{\alpha}$$

and  $u = \frac{p}{\rho}$ . In the limit  $\alpha \rightarrow 1$  the generalized Kullback-Leibler-divergence  $D_{GKL}$  (6) is obtained.

The  $\alpha$ -divergences are closely related to the generalized Rényi-divergences [1], [23], [24]:

$$D_\alpha^{GR}(p||\rho) = \frac{1}{\alpha - 1} \log \left( \int [p^\alpha \rho^{1-\alpha} - \alpha \cdot p + (\alpha - 1)\rho + 1] d\mathbf{x} \right) \tag{9}$$

The very outlier-robust  $\gamma$ -divergences

$$D_\gamma(p||\rho) = \frac{1}{\gamma + 1} \log \left[ \left( \int p^{\gamma+1} d\mathbf{x} \right)^{\frac{1}{\gamma}} \cdot \left( \int \rho^{\gamma+1} d\mathbf{x} \right) \right] - \log \left[ \left( \int p \cdot \rho^\gamma d\mathbf{x} \right)^{\frac{1}{\gamma}} \right] \tag{10}$$

were been proposed by FUJISAWA&EGUCHI [10]. In the limit  $\gamma \rightarrow 0$   $D_\gamma(p||\rho)$  becomes the usual Kullback-Leibler-divergence for normalized densities. For  $\gamma = 1$  the *Cauchy-Schwarz-divergence*

$$D_{CS}(p||\rho) = \frac{1}{2} \log \left( \int \rho^2(\mathbf{x}) d\mathbf{x} \cdot \int p^2(\mathbf{x}) d\mathbf{x} \right) - \log \left( \int p(\mathbf{x}) \cdot \rho(\mathbf{x}) d\mathbf{x} \right) \tag{11}$$

is obtained, which was suggested for information theoretic learning by J. PRINCIPE investigating the Cauchy-Schwarz-inequality for norms [21].

The derivatives of divergences  $D$  with respect to  $\rho$  are *functional derivatives*. Hence, the mathematical framework is the concept of *Fréchet-derivatives* or *functional derivatives*  $\frac{\delta D(p||\rho)}{\delta \rho}$  [9], [14]. Applying this methodology we obtain for the  $\beta$ -divergences (5):

$$\frac{\delta D_\beta(p||\rho)}{\delta \rho} = -p \cdot \rho^{\beta-2} + \rho^{\beta-1} \tag{12}$$

with the special case

$$\frac{\delta D_{GKL}(p||\rho)}{\delta \rho} = -\frac{p}{\rho} + 1 \tag{13}$$

for the generalized Kullback-Leibler-divergence. For the  $\alpha$ -divergences we get

$$\frac{\delta D_\alpha(p||\rho)}{\delta \rho} = -\frac{1}{\alpha} (p^\alpha \rho^{-\alpha} - 1) \tag{14}$$

and the related generalized Rényi-divergences (9) can be treated according to

$$\frac{\delta D_\alpha^{GR}(p||\rho)}{\delta \rho} = -\frac{p^\alpha \rho^{-\alpha} - 1}{\int [p^\alpha \rho^{1-\alpha} - \alpha \cdot p + (\alpha - 1) \rho + 1] d\mathbf{x}}. \tag{15}$$

The  $\gamma$ -divergences can be handled by

$$\frac{\delta D_\gamma(p||\rho)}{\delta \rho} = \frac{\rho^\gamma}{(\int \rho^{\gamma+1} d\mathbf{x})} - \frac{p\rho^{\gamma-1}}{(\int p \cdot \rho^\gamma d\mathbf{x})}. \tag{16}$$

with the important special case of the Cauchy-Schwarz-divergence for  $\gamma = 1$ .

Due to the lack of space, the derivation of these results can be found in [27].

## 4 Exemplary Application in Remote Sensing Data Analysis Using SOMs

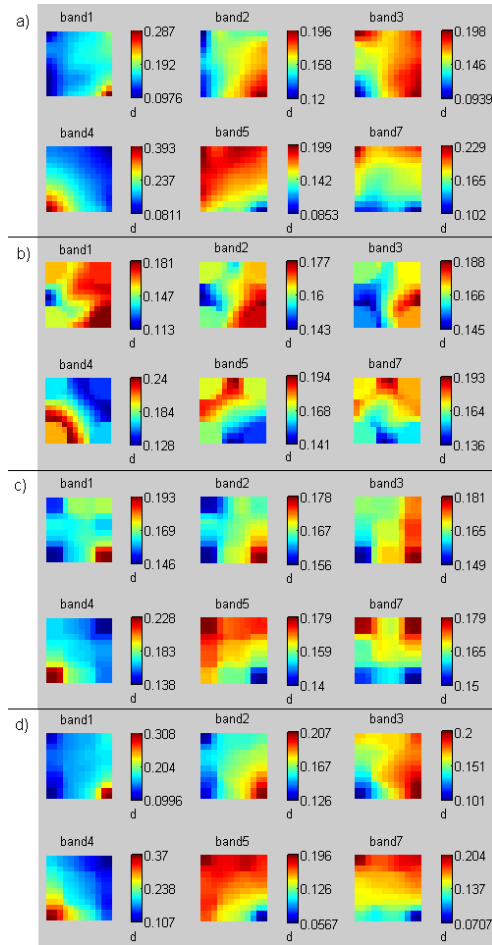
We demonstrate the online learning vector quantization by means of divergences for the widely used SOMs. The application area is remote sensing data analysis. Spectral remote sensing data  $\mathbf{v}$  reflect the responses of a spectral sensor at a suite of wavelengths [4], [6]. The spectrum is a characteristic fingerprint pattern that identifies the surface material within the area defined by an image pixel. LANDSAT TM satellite-based sensors produce images of the Earth in 7 different spectral bands. The ground resolution in meters is  $30 \times 30$  for bands 1 – 5 and band 7. Therefore, band 6 (thermal band) is often dropped resulting in six-dimensional data vectors. In the present contribution we use an image from the Colorado area, U.S.A [2], which was also used in earlier investigations [28]. All pixels of the Colorado image have associated ground truth labels, categorized into 14 classes covering several types of vegetation and soil [28]. From this study it is known that a two-dimensional ( $12 \times 12$ )-SOM-lattice is approximately sufficient for topology preserving data representation. Therefor we used a two-dimensional lattice also for the investigations here.

We used the Kullback-Leibler-, the Rényi, and the Cauchy–Schwarz-divergence to demonstrate the online vector quantization learning. To illustrate the effect of the different divergence approaches, we picture the component planes of the resulted SOMs, which show the distribution of the prototype values  $\mathbf{w}_r$  within each band according to their node position  $\mathbf{r}$  in the two-dimensional SOM-lattice A, see Fig 1.

For comparison we added the result for the SOM using the Euclidean metric. As expected, the results differ significantly. In fact, the degree of topology preservation is slightly decreased, which is judged by the topographic product  $TP$  [3] which should yield zero values for perfect topographic mapping. For the

---

<sup>2</sup> Thanks to M. Augusteijn (Univerity of Colorado) for providing this image.



**Fig. 1.** Component planes of the several SOMs using different dissimilarity measures: a) Euclidean distance, b) Kullback-Leibler divergence, c) Rényi divergence, and d) Cauchy-Schwarz divergence

Euclidean map we obtain  $TP_E = 0.0049$ , for the generalized Kullback-Leibler-divergence based SOM we get  $TP_{GKL} = 0.117$  suggesting a noticeable topographic violation. Yet, for the Cauchy-Schwarz- and the Renyi divergence we have  $TP_{CS} = 0.031$  and  $TP_{GR} = 0.039$ , respectively, which both still indicate a good topology preservation. As we can visually observe, the distribution becomes more sharpened compared to the Euclidean SOM which may lead to a better differentiation of small deviation. This hypothesis is supported by the inspection of the class distribution. Inspecting exemplarily the class distribution for Cauchy-Schwarz-SOM with the Euclidean SOM we see a more compact class distribution for the divergence based SOM, Fig 2.

|   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |   |   |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|---|
| 3 | 3  | 2  | 13 | 13 | 13 | 1  | 13 | 1  | 1  | 2  | 2  | 3  | 3  | 4  | 6  | 1  | 13 | 13 | 13 | 13 | 1  | 1  | 2  |   |   |
| 3 | 4  | 13 | 13 | 13 | 13 | 13 | 13 | 1  | 1  | 1  | 2  | 4  | 4  | 6  | 12 | 13 | 13 | 13 | 13 | 13 | 13 | 1  | 1  |   |   |
| 4 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 1  | 1  | 1  | 4  | 6  | 12 | 12 | 12 | 13 | 13 | 13 | 13 | 13 | 13 | 1  |   |   |
| 4 | 6  | 12 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 1  | 1  | 8  | 8  | 10 | 12 | 12 | 12 | 12 | 13 | 1  | 13 | 13 | 13 |   |   |
| 6 | 12 | 12 | 12 | 12 | 13 | 13 | 13 | 13 | 13 | 13 | 1  | 8  | 10 | 10 | 12 | 12 | 12 | 7  | 7  | 1  | 1  | 1  | 1  |   |   |
| 4 | 10 | 12 | 12 | 12 | 13 | 1  | 1  | 1  | 1  | 1  | 1  | 8  | 8  | 10 | 10 | 7  | 7  | 7  | 7  | 7  | 1  | 1  | 1  |   |   |
| 4 | 8  | 10 | 10 | 12 | 7  | 1  | 1  | 1  | 1  | 1  | 13 | 2  | 11 | 11 | 11 | 10 | 8  | 7  | 7  | 7  | 7  | 1  | 1  | 5 |   |
| 4 | 8  | 8  | 10 | 10 | 8  | 8  | 1  | 13 | 1  | 1  | 2  | 11 | 8  | 8  | 10 | 8  | 7  | 7  | 7  | 7  | 1  | 1  | 1  | 5 |   |
| 4 | 8  | 11 | 11 | 8  | 8  | 7  | 7  | 7  | 1  | 5  | 5  | 11 | 4  | 4  | 8  | 8  | 8  | 7  | 7  | 2  | 2  | 2  | 2  | 2 |   |
| 4 | 8  | 11 | 11 | 8  | 8  | 7  | 7  | 1  | 1  | 2  | 5  | 4  | 8  | 11 | 11 | 11 | 8  | 8  | 6  | 2  | 2  | 14 | 3  | 3 |   |
| 4 | 11 | 11 | 11 | 11 | 8  | 7  | 7  | 1  | 2  | 2  | 3  | 4  | 4  | 11 | 11 | 4  | 4  | 4  | 4  | 4  | 4  | 14 | 14 | 3 | 3 |
| 4 | 4  | 4  | 4  | 4  | 4  | 6  | 6  | 2  | 3  | 14 | 9  | 4  | 4  | 11 | 4  | 4  | 4  | 4  | 4  | 4  | 4  | 14 | 14 | 9 | 9 |

Fig. 2. Class distribution in the SOM lattice for Euclidean SOM (left) and Cauchy-Schwarz-divergence based SOM (right)

### 5 Conclusion

In this contribution we presented gradient descent learning in prototype-based vector quantization based on divergences. We explained, how divergences and their (Fréchet-) derivatives can be plugged into respective learning schemes like SOM, NG or other gradient based supervised and unsupervised vector quantizer. For unsupervised vector quantization this means that the expectation value of the relative information defined by the divergence is tried to minimize instead of the expectation value of the usually mean squared error as it is known from usual (Euclidean) vector quantization. We exemplarily show the effects of divergence utilization for data example stemming from remote sensing data analysis.

### References

1. Amari, S.-I.: Differential-Geometrical Methods in Statistics. Springer, Heidelberg (1985)
2. Banerjee, A., Merugu, S., Dhillon, I., Ghosh, J.: Clustering with bregman divergences. *Journal of Machine Learning Research* 6, 1705–1749 (2005)
3. Bauer, H.-U., Pawelzik, K.R.: Quantifying the neighborhood preservation of Self-Organizing Feature Maps. *IEEE Trans. on Neural Networks* 3(4), 570–579 (1992)
4. Campbell, J.: Introduction to Remote Sensing. The Guilford Press, U.S.A. (1996)
5. Cichocki, A., Zdunek, R., Phan, A., Amari, S.-I.: Nonnegative Matrix and Tensor Factorizations. Wiley, Chichester (2009)
6. Clark, R.N.: Spectroscopy of rocks and minerals, and principles of spectroscopy. In: Rencz, A. (ed.) *Manual of Remote Sensing*. John Wiley and Sons, Inc., New York (1999)
7. Cottrell, M., Hammer, B., Hasenfu, A., Villmann, T.: Batch and median neural gas. *Neural Networks* 19, 762–771 (2006)
8. Csiszr, I.: Information-type measures of differences of probability distributions and indirect observations. *Studia Sci. Math. Hungaria* 2, 299–318 (1967)
9. Frigvik, B.A., Srivastava, S., Gupta, M.: An introduction to functional derivatives. Technical Report UWEETR-2008-0001, Dept of Electrical Engineering, University of Washington (2008)

10. Fujisawa, H., Eguchi, S.: Robust parameter estimation with a small bias against heavy contamination. *Journal of Multivariate Analysis* 99, 2053–2081 (2008)
11. Heskes, T.: Energy functions for self-organizing maps. In: Oja, E., Kaski, S. (eds.) *Kohonen Maps*, pp. 303–316. Elsevier, Amsterdam (1999)
12. Hulle, M.M.V.: Kernel-based topographic map formation achieved with an information theoretic approach. *Neural Networks* 15, 1029–1039 (2002)
13. Jang, E., Fyfe, C., Ko, H.: Bregman divergences and the self organising map. In: Fyfe, C., Kim, D., Lee, S.-Y., Yin, H. (eds.) *IDEAL 2008*. LNCS, vol. 5326, pp. 452–458. Springer, Heidelberg (2008)
14. Kantorowitsch, I., Akilow, G.: *Funktionalanalysis in normierten Rumen*, 2nd edn. Akademie-Verlag, Berlin (1978) (revised edition)
15. Kohonen, T.: *Self-Organizing Maps*. Springer Series in Information Sciences, vol. 30. Springer, Heidelberg (1995) (Second Extended Edition 1997)
16. Kullback, S., Leibler, R.: On information and sufficiency. *Annals of Mathematical Statistics* 22, 79–86 (1951)
17. Lee, J., Verleysen, M.: Generalization of the  $l_p$  norm for time series and its application to self-organizing maps. In: Cottrell, M. (ed.) *Proc. of Workshop on Self-Organizing Maps, WSOM 2005*, Paris, Sorbonne, pp. 733–740 (2005)
18. Lehn-Schiler, T., Hegde, A., Erdogmus, D., Principe, J.: Vector quantization using information theoretic concepts. *Natural Computing* 4(1), 39–51 (2005)
19. Linde, Y., Buzo, A., Gray, R.: An algorithm for vector quantizer design. *IEEE Transactions on Communications* 28, 84–95 (1980)
20. Martinetz, T.M., Berkovich, S.G., Schulten, K.J.: ‘Neural-gas’ network for vector quantization and its application to time-series prediction. *IEEE Trans. on Neural Networks* 4(4), 558–569 (1993)
21. Principe, J.C., Fisher III, J.W., Xu, D.: Information theoretic learning. In: Haykin, S. (ed.) *Unsupervised Adaptive Filtering*, Wiley, New York (2000)
22. Qin, A., Suganthan, P.: A novel kernel prototype-based learning algorithm. In: *Proc. of the 17th Internat. Conf. on Pattern Recognition, ICPR 2004*, vol. 4, pp. 621–624 (2004)
23. Renyi, A.: On measures of entropy and information. In: *Proc. of the 4th Berkeley Symp. on Mathematical Statistics and Probability*. Univ. of California Press, Berkeley (1961)
24. Renyi, A.: *Probability Theory*. North-Holland Publish. Company, Amsterdam (1970)
25. Sato, A., Yamada, K.: Generalized learning vector quantization. In: Touretzky, D.S., Mozer, M.C., Hasselmo, M.E. (eds.) *Proc. of the 1995 Conf. on Advances in Neural Information Processing Systems*, vol. 8, pp. 423–429. MIT Press, Cambridge (1996)
26. Villmann, T., Claussen, J.-C.: Magnification control in self-organizing maps and neural gas. *Neural Computation* 18(2), 446–469 (2006)
27. Villmann, T., Haase, S.: Mathematical aspects of divergence based vector quantization using frchet-derivatives - extended and revised version. *Machine Learning Reports* 4(MLR-01-2010), 1–35 (2010), ISSN:1865-3960, [http://www.uni-leipzig.de/~compint/mlr/mlr\\_01\\_2010.pdf](http://www.uni-leipzig.de/~compint/mlr/mlr_01_2010.pdf)
28. Villmann, T., Merényi, E., Hammer, B.: Neural maps in remote sensing image analysis. *Neural Networks* 16(3-4), 389–403 (2003)
29. Villmann, T., Schleif, F.-M.: Functional vector quantization by neural maps. In: Chanussot, J. (ed.) *Proceedings of First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS 2009)*, pp. 1–4. IEEE Press, Los Alamitos (2009)
30. Zador, P.L.: Asymptotic quantization error of continuous signals and the quantization dimension. *IEEE Transaction on Information Theory* (28), 149–159 (1982)

# Using Feature Selection Approaches to Find the Dependent Features

Qin Yang, Elham Salehi, and Robin Gras

School of Computer Science, University of Windsor 401 Sunset Avenue Windsor,  
ON N9B 3P4 Canada

yang11t@uwindsor.ca, salehie@uwindsor.ca, rgras@uwindsor.ca

**Abstract.** Dependencies among the features can decrease the performance and efficiency in many algorithms. Traditional methods can only find the linear dependencies or the dependencies among few features. In our research, we try to use feature selection approaches for finding dependencies. We use and compare Relief, CFS, NB-GA and NB-BOA as feature selection approaches to find the dependent features among our artificial data. Unexpectedly, Relief has the best performance in our experiments, even better than NB-BOA, which is a population-based evolutionary algorithm that used the population distribution information to find the dependent features. It may be because some weak "link strengths" between features or due to the fact that Naïve Bayes classifier which is used in these wrapper approaches cannot represent the dependencies between features. However, the exact reason for these results still is an open problem for our future work.

**Keywords:** dependent features, feature selection, BOA.

## 1 Introduction

Finding high order dependent features has become increasingly important. Relevant features (the features which are relevant with the target concept) could be mutually dependent. This may greatly affect the classification performance and efficiency. Finding these features can help reduce program complexity, solve large problems and extract knowledge from datasets. Some traditional approaches such as measuring correlation coefficient have been used to solve this problem, but there are still difficulties to find non-linear dependencies or dependencies among many features. In our research, we try to use feature selection approaches to overcome these difficulties.

Feature subset selection can be reformulated as finding the minimal features which are necessary and sufficient to describe the target concept. Most typical feature subset selection approaches have two major steps: 1. a generation procedure to generate the next candidate subset; 2. an evaluation function to evaluate the feature subset. Based on the evaluation criterion, feature selection methods can be divided into filter model [1] and wrapper model [2]. Filter Model: select good features based on the certain data intrinsic properties. Wrapper Model is



defined as using classification accuracy as an evaluation criterion to evaluate and select the candidate feature subsets.

The generation procedure of feature selection can be considered as a search problem, thus each state in the search space represents a subset of the possible features of the task. For a large non-monotonic feature set, exhaustive evaluation of possible feature subsets is usually unfeasible because of the exponential computational time requirements. Genetic Algorithms (GAs), as randomized, evolutionary and population-based search algorithms, have been employed to solve this problem. They are inspired by biological evolution: reproduction, mutation, recombination, and selection. Evolution then takes place after the repeated application of the above operators. One drawback of GAs is it does not use the information about the dependencies among the related features. Estimate of Distribution algorithms (EDAs) [3], in which the new population is sampled from a probabilistic distribution from the selected individuals, can provide additional information about the problem being solved. The probabilistic model of the population that represents the dependencies among relevant features is an important source of information that can be exploited and used to enhance the performance of EDAs. It can also assist the user in a better interpretation and understanding of the underlying structure of the problem.

In our research, we try to use feature selection techniques to find the dependent features among datasets. There are several approaches that have been used and compared. For the filter model, we tested Relief and CFS. For the wrapper model, the classical genetic algorithm and Bayesian Optimization Algorithms (BOA) [4] as search algorithm are used for feature subset generation procedures (BOA is one of EDAs). Naïve Bayes [5] is the classifier in our experiments.

To clearly evaluate the performances of the chosen approaches, we simplified the experimental conditions in order that all the features relevant to the target concept in the artificial dataset are mutually dependent upon some other features. These relevant features are the dependent features that we try to find. We use capital letters like X, Y, Z... for the instances. Each instance composed of n features e.g.( $x_1, x_2, \dots, x_n$ ). Lower case letters a, b, c, d, e, f, g, h, s represent training datasets.

## 2 The Filter Model Feature Selection

The filter model uses some intrinsic properties such as distance, consistency, and correlation of data to select the optimal feature set.

### 2.1 Relief

Relief [6] is a feature weight based algorithm. Given training dataset  $s$ . Relief detects those features which are statistically relevant to the target concept. Differences of feature values between two instances X and Y are defined by the following function *diff*. When  $x_k$  and  $y_k$  are nominal,

$$diff(x_k, y_k) = \begin{cases} 0 & \text{if } x_k \text{ and } y_k \text{ are the same} \\ 1 & \text{if } x_k \text{ and } y_k \text{ are different} \end{cases} \quad (1)$$

$$diff(x_k, y_k) = (x_k - y_k) / nu_k, \text{ when } x_k \text{ and } y_k \text{ are numerical} \quad (2)$$

$nu_k$  is a normalization unit used to normalize the values of diff into the interval  $[0, 1]$ . Relief randomly picks  $m$  instances from the dataset and calculates each individual's Near-hit instance (the closest instance according to Euclidean Distance in same class) and Near-miss instance (the closest instance according to Euclidean Distance in opposite class). It updates the feature weight vector  $W$  for all  $m$  samples to determine the average feature relevance weight vector (of all the features to the target concept). Finally, Relief selects those features whose average weights ('relevance level') are above the given threshold  $\tau$ .

## 2.2 Correlation-Based Feature Selection (CFS)

The key point of the CFS algorithm [7] is a heuristic for evaluation of the worth or merit of subset features. The equation formalizing the heuristic is:

$$Merit_s = \frac{k\bar{r}_{cf}}{\sqrt{k + k(k-1)\bar{r}_{ff}}} \quad (3)$$

Where  $Merit_s$  is the heuristic "merit" of a feature subset  $s$  containing  $k$  features,  $\bar{r}_{cf}$  is the average feature-class correlation, and,  $\bar{r}_{ff}$  is the average feature-feature inter-correlation.

In order to apply this equation to estimate the merit of feature subsets, CFS uses symmetrical uncertainty ( $SU$ , a modified information gaining measure) [8] to estimate the degrees of correlation between discrete features.

$$SU = 2.0 \times \left[ \frac{H(x_i) + H(x_j) - H(x_i, x_j)}{H(x_i) + H(x_j)} \right] \quad (4)$$

## 3 The Wrapper Model Feature Selection

The wrapper Model evaluates the candidate feature subsets by its classification accuracy. Classifier and the feature subset generation procedure are two important components of it.

### 3.1 Classifier

Compared with the other classifiers (Decision Tree) [9] which we tried in our experiments, Naïve Bayer has higher classification accuracy and much less running time. In future experiments, it would be interesting to test other classifiers such as Neural Network [10], or SVM [11], which can represent complex networks of dependencies.

### 3.2 Feature Subset Generation Procedure

The feature subset generation procedure is used to generate the new candidate feature sets. How to find possible feature sets is a combinatorial optimization problem. In our case, each dataset has 10,000 instances, and each instance has

100 features. The exact search algorithms, because the size of the search space is exponential with the number of features, are not useable in our experiments. Local search algorithms such as Tabu search, simulated annealing and hill climbing use only a single solution at a time, which is modified to explore the search space guided by an objective function. On the contrary, GAs or EDAs are population based search algorithms, which mean that they use a sample of the search space instead of a single solution. This sample can be used to extract some properties about the research space, which can be useful to improve the exploration performance. These properties are directly linked to the dependent features. We expect that the properties which will be extracted during the exploration will be useful to localize the dependent feature. This is the reason why we chose to use evolutionary algorithms as our search method.

EDAs and BOA are evolutionary and population-based search algorithms, which do not use mutation and crossover operators. Contrary to GAs, these algorithms explicitly search for the dependencies among features, and use the population sample as a training set for building a probabilistic model. The new population is sampled from the probabilistic distribution by the probabilistic model. BOA uses a Bayesian Network as a probabilistic model to estimate the joint distribution of promising solutions, and then samples the new individuals (the new population) from the joint probability distribution encode by the Bayesian Network.

## 4 Experiments

Our goal is to discover the dependent features among the data. We want to evaluate the feature selection approaches we have mentioned on several artificial datasets which are specifically designed for our evaluation. Each dataset is composed of two set: D1 and D2, one for class 1 and one for class 2. Each set has 5,000 instances. Every instance is composed of 100 integer value features. Among these 100 features, there are 25 that are dependent, which means that each of them has at least one dependency relation with at least one of the other 24. The other 75 features are independent, and they do not have any dependency relation with others. In order to build the dependency relation among the 25 dependent features, a Bayesian Network based on the features, is associated to each set. Therefore, D1 and D2 use different Bayesian Networks, and in each dataset of our experiment we have two sets for which the  $D=25$  dependent features are generated from two different Bayesian networks. The other  $I=75$  independent features are generated by using the same distribution for the 2 class. This means that the properties that specifically correlated with each class only depend on the 25 dependent features. Therefore, the classifiers associated with feature selection approaches should be able to detect those 25 dependent features that are useful for classification. There are 8 datasets {a, b, c, d, e, f, g, h} in our experiments. The differences between these datasets are the degrees of the dependency and the complexity of the Bayesian network (Number of edges in the network).

**Table 1.** The characteristic of datasets including: the distribution of independent features(I), the maximum degree of dependency(K) and number of dependencies(A)

| Dataset | Distribution of I | K  | A   |
|---------|-------------------|----|-----|
| a       | random            | 5  | 40  |
| b       | (80,10,10)        | 5  | 40  |
| c       | (30,35,35)        | 5  | 40  |
| d       | random            | 2  | 40  |
| e       | (80,10,10)        | 2  | 40  |
| f       | random            | 5  | 70  |
| g       | random            | 10 | 70  |
| h       | random            | 10 | 120 |

#### 4.1 Experiments

We test the capacity of different feature subset selection approaches on different artificial datasets, which have different degrees of difficulty dependency on the complexity of the dependency network model on the different kinds of independent distribution. For the filter model, we use Relief, and CFS. We set the threshold  $\tau$  of Relief to 0. The programs we used are implemented by Weka [12]. The results are presented in Table 4. For the Wrapper model, we use the Naïve Bayes as the classifier with GAs and BOA search algorithm respectively (NB-GA, NB-BOA). The fitness function we used is:

$$fitness = Acc./n^p \quad (5)$$

$Acc.$  is the accuracy of the Naïve Bayes classifier,  $n$  is the number of features and the  $p$  is an adjustment parameter. We concentrate on finding the high order dependent features among the data. The classification accuracy is not the only issue we should consider. Different  $p$  value can bring very different results (compare Table 2, 3, 4 and 5). Increase the  $p$  value will reduce the selected features and slightly decrease the classification accuracy in some domains. How to set  $p$  is a difficult problem and it will be solved by using multiple-objective approach in future work.

Datasets are generated according to the method we mentioned before. The results of the classification accuracy for selected features by using different feature selection approaches are shown in Table 5. The classifier is Naïve Bayes with 3-fold cross validation. NB-BOA and NB-GA can get better classification performance than CFS and Relief when they have similar number of selected features. But we just focus on finding the dependent features among the data. The main measure we used to evaluate each approach is the total number of real dependent features and the selected features.

The value in the Table 2 and 3 is an average value over 12 runs. The BOA program is implemented by Pelikan [13]. According to Table 2 and Table 3, the NB-BOA and NB-GA got very similar results: similar fitness value, similar number of founded dependent features, and similar number of selected features. NB-BOA needs less generation for convergence. But considering that it needs more time to construct the Bayesian networks, the overall running time is not much less than NB-GA.

**Table 2.** The experiment results of NB-BOA and NB-GA. The fitness function adjustment parameter  $p=0.0025$ . “Gens” is the number of generations, and “time(s)” is the total running time in seconds needed for the NB-BOA and NB-GA to converge. “Dependent features” in the is the number of real dependent features discovered by our approaches. “Select features” is the total number of selected features by the feature selection approaches (dependent plus independent).

| NB-BOA(P=0.0025) |       |         |               |                    |                   | NB-GA(P=0.0025) |         |               |                    |                   |
|------------------|-------|---------|---------------|--------------------|-------------------|-----------------|---------|---------------|--------------------|-------------------|
| Dataset          | Gens. | Time(s) | Fitness Value | dependent features | selected features | Gens.           | Time(s) | Fitness Value | dependent features | selected features |
| a                | 31.17 | 272.4   | 0.9676        | 19                 | 25.3              | 41.67           | 320.58  | 0.9677        | 18.8               | 24.8              |
| b                | 32.92 | 286.4   | 0.9678        | 19                 | 25.7              | 40.5            | 321.33  | 0.9679        | 19                 | 24.9              |
| c                | 31.58 | 279.1   | 0.9678        | 19.3               | 27.3              | 37.83           | 298.42  | 0.9679        | 19                 | 27.7              |
| d                | 41.75 | 382.4   | 0.8841        | 19.1               | 45.1              | 52.58           | 443.83  | 0.8843        | 19.9               | 47.4              |
| e                | 37.33 | 336.7   | 0.8830        | 21.5               | 36                | 53.5            | 438.83  | 0.8832        | 20.9               | 37.1              |
| f                | 34.5  | 289.1   | 0.9495        | 16.2               | 25.4              | 44.08           | 333.33  | 0.9495        | 16.3               | 23.8              |
| g                | 39    | 348.8   | 0.9152        | 16.6               | 39                | 49.83           | 409.92  | 0.9153        | 16.8               | 40.3              |
| h                | 41.83 | 388.3   | 0.8062        | 18.1               | 49.3              | 47.75           | 404.58  | 0.8065        | 17.8               | 47.1              |

**Table 3.** The experiment results of NB-BOA and NB-GA where the fitness function adjustment parameter  $p=0.005$

| NB-BOA(P=0.005) |       |               |                    |                   | NB-GA(P=0.005) |               |                    |                   |  |
|-----------------|-------|---------------|--------------------|-------------------|----------------|---------------|--------------------|-------------------|--|
| Dataset         | Gens. | Fitness Value | dependent features | selected features | Gens.          | Fitness Value | dependent features | selected features |  |
| a               | 25.33 | 0.9605        | 18                 | 18.5              | 28             | 0.9605        | 18.8               | 18.8              |  |
| b               | 24.75 | 0.9605        | 18.5               | 18.9              | 28.75          | 0.9605        | 18.7               | 18.7              |  |
| c               | 25.58 | 0.9604        | 19                 | 20.3              | 28.17          | 0.9605        | 19                 | 20                |  |
| d               | 35    | 0.8761        | 19.1               | 32.8              | 45.67          | 0.8762        | 19.8               | 33.3              |  |
| e               | 32.17 | 0.8756        | 20.5               | 27.9              | 38.5           | 0.8758        | 21.1               | 27.2              |  |
| f               | 25.83 | 0.9428        | 14.2               | 14.7              | 31.67          | 0.9428        | 13.8               | 14.4              |  |
| g               | 31.58 | 0.9076        | 16                 | 22.6              | 38.25          | 0.9078        | 16                 | 21.2              |  |
| h               | 39.58 | 0.7984        | 17.3               | 41.3              | 48.42          | 0.7986        | 17.2               | 41.5              |  |

**Table 4.** The results of the comparison of using different feature selection approaches to find the dependent features. Where 14/15 mean (number of the selected dependent features)/ (total number of the selected feature). The threshold  $\tau$  of Relief is 0.

| Dataset | CFS   | Relief | BOA<br>$p=0.0025$ | BOA<br>$p=0.005$ | BOA<br>$p=0.01$ | GA<br>$p=0.0025$ | GA<br>$p=0.005$ | GA<br>$p=0.01$ |
|---------|-------|--------|-------------------|------------------|-----------------|------------------|-----------------|----------------|
| a       | 14/15 | 25/28  | 19/25             | 18/19            | 15/15           | 19/25            | 19/19           | 13/13          |
| b       | 14/14 | 25/28  | 19/25             | 19/19            | 15/15           | 19/25            | 19/19           | 14/14          |
| c       | 14/14 | 25/26  | 19/28             | 19/20            | 15/15           | 20/28            | 19/20           | 14/14          |
| d       | 11/14 | 25/26  | 20/48             | 19/32            | 19/21           | 20/47            | 20/33           | 19/21          |
| e       | 11/13 | 25/28  | 22/36             | 21/28            | 19/21           | 22/38            | 21/27           | 18/19          |
| f       | 10/11 | 25/26  | 16/25             | 14/15            | 10/11           | 16/24            | 13/14           | 9/9            |
| g       | 9/11  | 25/26  | 17/41             | 16/23            | 16/17           | 17/40            | 16/21           | 16/17          |
| h       | 8/10  | 19/22  | 18/49             | 17/41            | 13/17           | 17/49            | 17/42           | 13/15          |

In our experiment, we found that even though the dataset has become more complicated, Relief always has a similar good performance, whereas the performances of CFS, NB-GA and NB-BOA become worse (Table 4). But the performance of relief also decreases significantly for a very complicated network.

**Table 5.** The results of the comparison of classification accuracy of using different feature selection approaches. Classifier is Naive Bayes with 3-fold cross validation.

| Dataset | CFS    | Relief | BOA<br>p=0.0025 | BOA<br>p=0.005 | BOA<br>p=0.01 | GA<br>p=0.0025 | GA<br>p=0.005 | GA<br>p=0.01 |
|---------|--------|--------|-----------------|----------------|---------------|----------------|---------------|--------------|
| a       | 0.9714 | 0.9729 | 0.9754          | 0.9746         | 0.9739        | 0.9755         | 0.9747        | 0.9731       |
| b       | 0.9710 | 0.9727 | 0.9756          | 0.9747         | 0.9738        | 0.9757         | 0.9746        | 0.9733       |
| c       | 0.9710 | 0.9728 | 0.9758          | 0.9750         | 0.9739        | 0.9759         | 0.9750        | 0.9734       |
| d       | 0.8735 | 0.8840 | 0.8926          | 0.8915         | 0.8890        | 0.8928         | 0.8917        | 0.8893       |
| e       | 0.8731 | 0.8837 | 0.8909          | 0.8903         | 0.8888        | 0.8912         | 0.8904        | 0.8886       |
| f       | 0.9517 | 0.9483 | 0.9572          | 0.9555         | 0.9537        | 0.9571         | 0.9554        | 0.9524       |
| g       | 0.9053 | 0.9144 | 0.9236          | 0.9219         | 0.9204        | 0.9238         | 0.9218        | 0.9203       |
| h       | 0.7296 | 0.8040 | 0.8141          | 0.8134         | 0.8080        | 0.8143         | 0.8136        | 0.8074       |

We could expect that, because Relief uses a statistical method, only corresponding to the feature individually, it should obtain poor results for these problems as the selected features are all mutually dependent. Classic genetic algorithm uses problem independent recombination operators, which do not use the information about the dependencies among the decision features, this is contrary to BOA (or other EDAs). CFS computes the correlation (dependency) between the feature subsets and class by measuring the information gained. In theory, we expect that the feature selection approach by using BOA should be good at finding the dependent relevant features comparing with those approaches using the genetic algorithms, Relief and CFS. But eventually, according to the table 3, we found that BOA only has similar performances with GAs and the Relief is the best method we used for finding the dependent features in our experiments.

## 5 Conclusion and Future Work

We generate different dataset with different degrees of complexity by using Bayesian Networks. We use Relief, CFS, NB-GA and NB-BOA feature selection approaches to find the dependent features among the data. Eventually, Relief has the best performance.

Several questions arise here: Why do approaches like BOA, which can handle dependency between features, have a worse performance? Why can Relief get the best result? We think that it can be due to the fact that the classifier we used, Naïve Bayes, can not represent dependencies between features. Although we have tested with Decision Tree classifier which can represent some kind of dependencies, but the results we obtained are worse than the results we obtained using Naïve Bayes classifier. We think that a more efficient classifier, which is able to represent complex dependencies, like NN or SVM, may lead to better results. We also think that it may be the "strength" of dependencies (or link strengths [14], which measures the level of dependency) between these dependent features are too weak. BOA considers some weakly dependent features as independent and ignores them. On the other hand, the fact that our datasets do not include redundant features is an advantage for Relief. As one of the known problems of Relief, it is difficult to filter redundant features. Our dataset do not present

these difficulties for the Relief approaches. However, the exact reason is still an open problem and more work need to be done in the future. We plan to measure the "link strengths" between the features and design the dataset according to the linkage strength we want and test it again. We also want to use some multi-objective methods to improve the NB-GA and NB-BOA.

**Acknowledgments.** This work is supported by the NSERC grant ORGPIN 341854, the CRC grant 950-2-3617 and the CFI grant 203617. It is made possible by the facilities of the Shared Hierarchical Academic Research Computing Network (SHARCNET:www.sharcnet.ca).

## References

1. Dash, M., Liu, H.: Consistency-based search in feature selection. *Artificial Intelligence* 151(1), 155–176 (2003)
2. Kohavi, R., John, G.: Wrappers for feature subset selection. *Artificial intelligence* 97(1-2), 273–324 (1997)
3. Inza, I., Larranaga, P., Etxeberria, R., Sierra, B.: Feature subset selection by bayesian network-based optimization. *Artificial Intelligence* 123(1-2), 157–184 (2000)
4. Pelikan, M., Goldberg, D., Cantu-Paz, E.: Boa: The bayesian optimization algorithm. In: *Proceedings of the Genetic and Evolutionary Computation Conference GECCO 1999*, vol. 1, pp. 525–532 (1999) (Citeseer)
5. Lewis, D.: Naive (Bayes) at forty: The independence assumption in information retrieval. In: Nédellec, C., Rouveirol, C. (eds.) *ECML 1998*. LNCS, vol. 1398, pp. 4–18. Springer, Heidelberg (1998)
6. Kira, K., Rendell, L.: The feature selection problem: Traditional methods and a new algorithm. In: *Proceedings of the National Conference on Artificial Intelligence*, p. 129. John Wiley & Sons Ltd., Chichester (1992)
7. Hall, M.A.: Correlation-based feature selection for discrete and numeric class machine learning. In: *ICML 2000: Proceedings of the Seventeenth International Conference on Machine Learning*, pp. 359–366. Morgan Kaufmann Publishers Inc., San Francisco (2000)
8. Press, W., Teukolsky, S., Vetterling, W., Flannery, B.: *Numerical recipes in C*. Cambridge Univ. Press, Cambridge (1992)
9. Quinlan, J.: Induction of decision trees. *Machine learning* 1, 81–106 (1986)
10. Hagan, M., Demuth, H., Beale, M., et al.: *Neural network design*. PWS, Boston (1996)
11. Keerthi, S., Shevade, S., Bhattacharyya, C., Murthy, K.: Improvements to Platt's SMO algorithm for SVM classifier design. *Neural Computation* 13(3), 637–649 (2001)
12. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., H., I.: The weka data mining software: An update. *SIGKDD Explorations* 11 (2009)
13. Pelikan, M.: A simple implementation of the Bayesian optimization algorithm (BOA) in C++(version 1.0). IlliGAL Report 99011
14. Ebert-Uphoff, I.: *Measuring Connection Strengths and Link Strengths in Discrete Bayesian Networks*. Woodruff School of Mechanical Engineering, Atlanta, GA, Tech. Rep. (2006)

# Performance Assessment of Data Mining Methods for Loan Granting Decisions: A Preliminary Study

Jozef Zurada and Niki Kunene

Department of Computer Information Systems,  
College of Business,  
University of Louisville,  
Louisville, KY, USA  
jmzura01@louisville.edu

**Abstract.** After the greatest financial debacle since the great depression, the need for accurate and systematic assessment of loan granting decisions has never been more important than now. The paper compares the classification accuracy rates of six models: logistic regression (LR), neural network (NN), radial basis function neural network (RBFNN), support vector machine (SVM), k-Nearest Neighbor (kNN), and decision tree (DT) for loan granting decisions. We build models and test their classification accuracy rates on five very versatile data sets drawn from different loan granting decision contexts. The results from computer simulation constitute a fertile ground for interpretation.

**Keywords:** loan granting decisions, data mining methods, performance assessment, ROC chart.

## 1 Introduction

Lending companies have been using some financial ratios and/or statistical techniques to discriminate between creditworthy and non-creditworthy borrowers since 1930s. For example, FICO scores, back-end ratio (total monthly debt expense / gross monthly income) or loan to value ratio (mortgage loan amount requested / assessed property value) and traditional multiple regression analysis were commonly applied. To increase the accuracy of credit granting decisions, more sophisticated nonparametric artificial intelligent (AI) techniques, which could better capture an inherent nonlinearity between all the factors involved, have been proposed (Table 1). Even slight improvements in accuracy of predicting creditworthiness of customers can generate substantial future savings for lending companies.

In most recent years, however, the money lending industry in the U.S. has lacked any rigor and control in granting new loans for residential and commercial real estate properties, home refinancing, home equity, and credit cards. For example, a common phenomenon, which was tacitly promoted and agreed upon by money lending institutions, was the fact that real estate properties were often overvalued by appraisers. This way banks could justify lending more money to borrowers on home equity without adequate checking of their financial status and credit efficiency with



regard to paying a loan off and/or making timely payments on loans. Many of these loans turned out to be bad loans and they contributed to serious financial distress and/or collapse of many financial institutions in the last two years. To avoid bankruptcy a large number of financial firms had to be bailed out by the government. Thus, the adherence to accurate and systematic assessment of loan granting decisions and the need for building reliable models which could score potential borrowers according to their probability of default upon a loan has never been more important than now.

This study examines the classification accuracy rate of six models: logistic regression (LR), neural network (NN), radial basis function neural network (RBFNN), support vector machine (SVM), k-nearest neighbor ( $k$ NN), and decision tree (DT) on five data sets drawn from different financial contexts. To obtain reliable and unbiased error estimates for each of the six models and five data sets, we apply 10-fold cross-validation and repeat each experiment 10 times. We average the classification accuracy rates across 10 folds and 10 runs and use a 2-tailed paired  $t$ -test at  $\alpha=0.05$  recommended by Witten and Frank [1] to find out if the classification accuracy rates at 0.5 probability cut-off across the models and data sets are significantly different from LR which is used as the baseline. We also employ the ROC charts and examine the performance of the models at the probability cut-offs  $\neq 0.5$  which are more likely to be used by financial institutions.

Each of the methods proposed in this paper is already well-known and have successfully been used in many applications, including loan granting decisions or closely related fields (Table 1). Due to this reason and space constraints we do not include the description of the methods. To our best knowledge, the novel aspect of this contribution seems to be the fact that we build and test the models on five versatile real-world data sets, each having different characteristics in terms of the number of samples, the number and type of the variables, the presence of missing values, and the ratio of samples containing bad loans and good loans (Table 2). The results obtained provide a fertile ground for interpretation (Tables 3-6 and Figure 1). One can look at the efficiency of the models or the predictive power of the attributes contained in each of the five data sets. One can examine the ROC curves to determine the efficiency of the models at various operating points as well.

The paper is organized as follows. Section 2 briefly summarizes several previous studies regarding credit scoring and loan granting decisions. Section 3 presents the basic characteristics of the five data sets used, whereas section 4 describes computer simulation and the results. Finally, section 5 concludes the paper and outlines possible future research in this area.

## 2 Prior Research

There are a number of studies in which machine learning methods such as LR, NNs, RBFNNs, DTs, fuzzy systems (FS), neuro-fuzzy systems (NFS), genetic algorithms (GA), and many other techniques have been applied to credit scoring problems. Most of them report classification accuracy rates obtained for different models and computer simulation scenarios. A few studies concentrate on the models' interpretability issues and the feature reduction methods. Table 1 summarizes a representative sample of somewhat older and the most current research in the field.

**Table 1.** Summary of the Previous Studies

| Methods Used                                                                                                                                                                                                    | Data Sets Properties                                                                           | Results and Study                                                                                                                                                                                                                                                     |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NN, and NFS.                                                                                                                                                                                                    | Three data sets drawn from credit approval, loan default, and bank failure prediction context. | NN performed better than NFS. Unlike NN, however, NFS generated partially explainable if-then rules to explain the rationale behind the credit granting/denial decision [2], [3].                                                                                     |
| Five NN architectures multilayer perceptron, mixture-of-experts, RBFNN, learning vector quantization, and fuzzy adaptive resonance, discriminant analysis, LR, <i>k</i> NN, kernel density estimation, and DTs. | Two real world data sets.                                                                      | 10-fold cross-validation used. Among neural architectures the mixture-of-experts and RBFNN did best, whereas among the traditional methods LR analysis was the most accurate [4].                                                                                     |
| LR, NN, DT, MBR, and Ensemble Model.                                                                                                                                                                            | Data sets 3 and 4 from Table 2.                                                                | Both are comparative studies. DTs did best classification-wise. Also DTs are attractive tools as they can generate easy to interpret if-then rules [5], [6].                                                                                                          |
| Almost a dozen of the techniques: NN, RBFNN, CBR, rule extraction techniques, Naïve Bayesian classifier, and ensemble classifiers.                                                                              | One data set describing a bank's customers checking account information.                       | The overall classification accuracy rates were between 70% and 81%. The dependent variable took 3 distinct values: "declined", "risky", and "good" [7].                                                                                                               |
| SVMs and kernel attribute selection techniques and stepwise LR.                                                                                                                                                 | Two real world data sets containing a large number of attributes and samples.                  | Kernel methods correctly classified about 70% of "good" and "bad" loans, while the stepwise LR correctly classified about 90% and 24% of "good" and "bad" classes, respectively. The 24% classification accuracy rate for "bad" loans was certainly unacceptable [8]. |
| Multiple individual and ensemble methods: MLR, LR, NN, RBFNN, SVM. Ensemble models' decisions were based on fuzzy voting and averaging.                                                                         | Three data sets, including modified data set 1 (without missing values), and data set 3.       | Fuzzy group decision making (GDM) model outperformed other models on all 3 data sets [9].                                                                                                                                                                             |

### 3 Data Sets Used in the Study

The five data sets used in this study are drawn from different financial contexts and they describe financial, family, social, and personal characteristics of loan applicants. In two of the five data sets the names of the attributes have not been revealed because of the confidentiality issues. Two of the data sets are publicly available at the UCI

**Table 2.** The General Characteristics of the Five Data Sets Used in Computer Simulation

| Data set | Characteristics |                |                                                                                                            | Comments                                                                                                                                                                                                                                                                                                                                   |
|----------|-----------------|----------------|------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|          | # of cases      | # of variables | Class values:<br>target variable takes<br>B: bad loans or credit denied<br>G: good loans or credit granted |                                                                                                                                                                                                                                                                                                                                            |
| 1        | 690             | 16             | B: 383<br>G: 307                                                                                           | The Quinlan data set used in the number of studies. Describes financial attributes of Japanese customers. Available at the UCI Machine Learning Repository. The names of the attributes are not revealed. Well balanced. Bad loans are slightly overrepresented. Contains numeric and nominal variables. Some missing values are replaced. |
| 2        | 252             | 14             | B: 71<br>G: 181                                                                                            | The names of the attributes are not available. Unbalanced data set: bad loans are underrepresented. Includes only financial data of loan applicants. No missing values.                                                                                                                                                                    |
| 3        | 1000            | 21             | B: 300<br>G: 700                                                                                           | A data set from a German financial institution. Unbalanced data set: bad loans are underrepresented. The names of the attributes available. Contains numeric and nominal variables. No missing values.                                                                                                                                     |
| 4        | 5960            | 13             | B: 1189<br>G: 4771                                                                                         | The attributes describe financial, family, social, and personal characteristics of loan applicants. Unbalanced data set: bad loans are underrepresented by the ratio of about 1:4. Contains a large number of missing values which were replaced. Available from the SAS company.                                                          |
| 5        | 3364            | 13             | B: 300<br>G: 3064                                                                                          | Obtained from data set 4 by removing all missing values. Very unbalanced data set: bad loans are significantly underrepresented by the ratio of about 1:10.                                                                                                                                                                                |

Machine Learning Repository at <http://www.ics.uci.edu/~mllearn/databases/>. Table 2 presents the general features of each of the five data sets.

## 4 Experiments and Results

The computer simulation for this study was performed using widely popular and open-source software, Weka 3.7, written in Java ([www.cs.waikato.ac.nz/ml/weka/](http://www.cs.waikato.ac.nz/ml/weka/)). It contains a great deal of features for constructing and testing various models, transforming and reduction variables as well as interpreting and assessing the results.

In computer simulation we used the standard Weka settings for the 6 models. It is worth mentioning, however, that for at least the four methods, i.e., RBFNN, SVM, NN, and  $k$ NN a substantial experimentation with the models' parameters and multiple runs are required to choose the optimal set of parameters for each data set.

We only report the results for the 0.5 probability cutoff which means that the costs of making a Type I and Type II errors are the same. In the creditworthiness context, other cut-offs may also be appropriate. For example, a 0.3 cut-off means that Type II error (classifying a bad loan as a good loan) is 3.3 times more costly than the Type I error (classifying a good loan as a bad loan). This cutoff may be applicable to situations in which banks do not secure smaller loans, i.e., do not hold any collateral, whereas the 0.7 cutoff implies that the cost of making a Type I error is smaller than the cost of Type II error. This cut-off may typically be used when a financial institution secures larger loans by holding collateral such as customer's home.

The results obtained constitute a fertile ground for interpretation which could go in several directions. There are at least four dimensions to consider: (1) the methods, (2) the data sets, (3) the classification accuracy rates for overall, bad loans, and good loans at 0.5 probability cut-off, and (4) the area under the ROC curve which allows one to examine the models' performance for different than 0.5 operating points. For example, one may want to compare the performance of the six methods on five versatile data sets in an attempt to find the best two or the best method which work the best across all data sets. One may also look at the five data sets to find out one or two data sets which contain the best selection of features for reliable loan granting decisions. If detecting bad loans is of paramount interest, one could concentrate on finding the best model which does exactly that, etc. We leave most of these considerations to the reader and give only a brief interpretation of the results.

The classification accuracy rates are reported in Tables 3 through 5, and the area under the ROC curve which testifies to a general detection power of the models is presented in Table 6. Generally, one can see that the overall performance of the five models (out of six) is the highest and most stable on data set 1. This data set seems to contain the right balance of bad and good loans, with bad loans slightly overrepresented (Table 2). The results presented in Table 6 confirm these observations. Looking at each Table 3 through 6, however, enables one to draw more subtle conclusions.

**Table 3.** Overall Correct Classification Accuracy Rates [%] for the 6 Models

|          | LR   | NN                | RBFNN             | SVM               | $k$ NN            | DT                |
|----------|------|-------------------|-------------------|-------------------|-------------------|-------------------|
| Data Set |      |                   |                   |                   |                   |                   |
| 1        | 85.3 | 85.5              | 79.5 <sup>w</sup> | 84.9              | 86.1              | 85.6              |
| 2        | 78.0 | 78.6              | 74.5 <sup>w</sup> | 75.6 <sup>w</sup> | 79.0              | 75.5 <sup>w</sup> |
| 3        | 75.6 | 75.2              | 73.5 <sup>w</sup> | 75.5              | 72.9 <sup>w</sup> | 71.6 <sup>w</sup> |
| 4        | 83.6 | 86.9 <sup>b</sup> | 82.5 <sup>w</sup> | 82.9 <sup>w</sup> | 78.4 <sup>w</sup> | 88.9 <sup>b</sup> |
| 5        | 92.5 | 92.1 <sup>w</sup> | 91.1 <sup>w</sup> | 91.2 <sup>w</sup> | 91.5 <sup>w</sup> | 94.4 <sup>b</sup> |

<sup>w,b</sup> Significantly worse/better than LR at the 0.05 significance level.

Table 3 presents the overall percentage classification accuracy rates at the 0.5 probability cut-off point. RBFNNs, SVMs, *k*NN, and DTs in this order appear to perform significantly worse than LR and NNs methods. However, DTs seem to outperform LR and the remaining methods on data sets 4 and 5 of which the latter is highly unbalanced (Table 2). The best classification accuracy across all six models is for data set 5 in which the bad loan class is underrepresented by a ratio of 1:10. It mainly occurred due to the fact that good loans have been classified almost perfectly well.

Table 4 shows that all models classify bad loans consistently poorly on data sets 2 through 5. For these four data sets, the best and the worst classification accuracy rates are 59.0% (NNs) and 1.4% (SVMs). The latter do not appear to tolerate well, the data sets which are highly unbalanced. However, for data set 1, in which bad loans are slightly overrepresented, SVMs exhibit an extraordinary performance. DTs seem to be most efficient classifiers of bad loans for data sets 4 and 5 in which bad loans are underrepresented by a ratio of 1:4 and 1:10, respectively.

**Table 4.** “Bad Loan” Correct Classification Accuracy Rates [%] for the 6 Models

|          | LR   | NN                | RBFNN             | SVM               | <i>k</i> NN       | DT                |
|----------|------|-------------------|-------------------|-------------------|-------------------|-------------------|
| Data Set |      |                   |                   |                   |                   |                   |
| 1        | 86.4 | 84.2 <sup>w</sup> | 65.0 <sup>w</sup> | 92.1 <sup>b</sup> | 86.1              | 84.1 <sup>w</sup> |
| 2        | 45.5 | 35.5 <sup>w</sup> | 29.8 <sup>w</sup> | 48.7              | 40.3 <sup>w</sup> | 37.0 <sup>w</sup> |
| 3        | 49.0 | 50.3 <sup>b</sup> | 42.2 <sup>w</sup> | 47.8 <sup>w</sup> | 27.1 <sup>w</sup> | 41.0 <sup>w</sup> |
| 4        | 30.4 | 59.0 <sup>b</sup> | 35.5              | 18.5 <sup>w</sup> | 31.6              | 54.8 <sup>b</sup> |
| 5        | 22.7 | 14.2 <sup>w</sup> | 5.1 <sup>w</sup>  | 1.4 <sup>w</sup>  | 6.1 <sup>w</sup>  | 47.3 <sup>b</sup> |

<sup>w,b</sup> Significantly worse/better than LR at the 0.05 significance level.

For four out of five data sets, the *k*NN method, in general, outperforms the remaining methods in detecting good loans (Table 5). It is also evident that for data sets 2, 4, and 5, in which good loans substantially overrepresented bad loans, the classification models’ classification performance for good loans is well above 90%.

The area under the ROC curve is an important measure as it illustrates the overall performance of the models at various operating points. For example, if a target event

**Table 5.** “Good Loan” Correct Classification Accuracy Rates [%] for the 6 Models

|          | LR   | NN                | RBFNN             | SVM                | <i>k</i> NN       | DT                |
|----------|------|-------------------|-------------------|--------------------|-------------------|-------------------|
| Data Set |      |                   |                   |                    |                   |                   |
| 1        | 84.5 | 86.6              | 91.2 <sup>b</sup> | 79.1 <sup>w</sup>  | 86.1 <sup>b</sup> | 86.7 <sup>b</sup> |
| 2        | 90.6 | 95.2 <sup>b</sup> | 91.8              | 85.9 <sup>w</sup>  | 93.9 <sup>b</sup> | 90.3              |
| 3        | 87.0 | 85.8 <sup>w</sup> | 86.9              | 87.4               | 92.5 <sup>b</sup> | 84.8 <sup>w</sup> |
| 4        | 96.9 | 93.8 <sup>w</sup> | 94.3 <sup>w</sup> | 98.9 <sup>b</sup>  | 90.0 <sup>w</sup> | 97.4 <sup>b</sup> |
| 5        | 99.4 | 99.7 <sup>b</sup> | 99.5 <sup>b</sup> | 100.0 <sub>b</sub> | 99.9 <sup>b</sup> | 99.0 <sup>w</sup> |

<sup>w,b</sup> Significantly worse/better than LR at the 0.05 significance level.

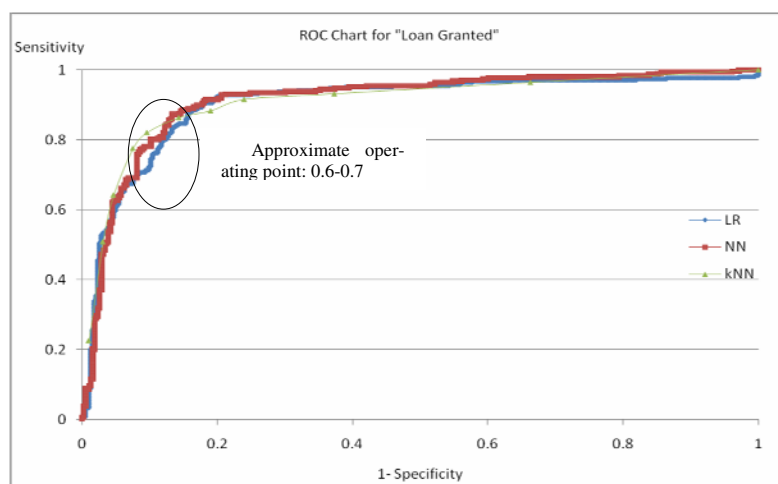
is detecting a bad loan and misclassifying the bad loan as a good loan is 3 times more costly than misclassifying a good loan as a bad loan, a lending institution may choose to use a 0.3 cut-off as the decision threshold. This means that a customer whose probability of default upon a loan is  $\geq 0.3$  will be denied the loan. Table 6 shows that data set 1 appears to contain the best attributes for distinguishing between good and bad loans across all six models. The RBFNN, SVM, and DT models significantly underperform the remaining three models. It is also apparent that more experiments are needed to find the best settings for the parameters of RBFNN and SVM.

**Table 6.** The Area Under the ROC Curve [%] for the 6 Models

|          | LR   | NN                | RBFNN             | SVM               | kNN               | DT                |
|----------|------|-------------------|-------------------|-------------------|-------------------|-------------------|
| Data Set |      |                   |                   |                   |                   |                   |
| 1        | 91.4 | 91.5              | 87.5 <sup>w</sup> | 85.6 <sup>w</sup> | 91.1              | 88.6 <sup>w</sup> |
| 2        | 73.9 | 74.4              | 69.1 <sup>w</sup> | 67.3 <sup>w</sup> | 72.2              | 57.7 <sup>w</sup> |
| 3        | 78.5 | 78.0 <sup>w</sup> | 75.2 <sup>w</sup> | 67.6 <sup>w</sup> | 74.4 <sup>w</sup> | 64.7 <sup>w</sup> |
| 4        | 79.4 | 86.3 <sup>b</sup> | 75.9 <sup>w</sup> | 58.7 <sup>w</sup> | 76.9 <sup>w</sup> | 84.4 <sup>b</sup> |
| 5        | 78.7 | 78.0              | 76.3 <sup>w</sup> | 50.7 <sup>w</sup> | 78.7              | 75.7 <sup>w</sup> |

<sup>w,b</sup> Significantly worse/better than LR at the 0.05 significance level.

To avoid clutter on the ROC charts and make them more transparent we chose to illustrate the performance of the best three models only for data set 1 (Fig. 1). These models are: LR, NN, and kNN. The three curves overlap to a large extent for most operating points exhibiting very good classification ability. However, kNN appears to outperform LR and NN at the operating points within the range of [0.6, 0.7]. A similar ROC chart could be developed for bad loans as well.



**Fig. 1.** The ROC charts for the LR, NN, and kNN Model for Data Set 1 (Quinlan)

## 5 Conclusion and Suggestions for Future Research

The paper evaluates the predictive performance of the six models on five versatile data sets drawn from different financial contexts. The Quinlan data set 1 appears to contain the best attributes for building effective models to classify consumer loans into the bad and good categories. More experimentation is needed with settings of the parameters for the RBFNN and SVM models, which proved to be very efficient classifiers in many other applications. It is also recommended to implement feature reduction techniques for possible improvement in the results as well as explore the rule induction methods if the models' interpretability is of paramount importance.

## References

1. Witten, I.H., Frank, E.: *Data Mining: Practical Learning Tools and Techniques*. Morgan Kaufmann Publishers, San Francisco (2005)
2. Piramuthu, S.: Financial Credit-Risk Evaluation with Neural and Neurofuzzy Systems. *European Journal of Operational Research* 112, 310–321 (1999)
3. Piramuthu, S.: Feature Selection for Financial Credit-Risk Evaluation Decisions. *INFORMS Journal on Computing* 11(3), 258–266 (1999)
4. West, D.: Neural Network Credit Scoring Models. *Computers & Operations Research* 27, 1131–1152 (2000)
5. Zurada, J.: Rule Induction Methods for Credit Scoring. *Review of Business Information Systems* 11(2), 11–22 (2007)
6. Zurada, J.: Could Decision Trees Improve the Classification Accuracy and Interpretability of Loan Granting Decisions? In: *Proceedings of the 43rd Hawaii International Conference on System Sciences (HICSS 2010)*, January 5-8. IEEE Computer Society Press, Hawaii (2010)
7. Huang, Y.-H., Hung, C.-M., Jiau, H.C.: Evaluation of Neural Networks and Data mining Methods on a Credit Assessment Task for Class Imbalance Problem. *Nonlinear Analysis: Real Worlds Applications* 7, 720–747 (2006)
8. Yang, Y.: Adaptive Credit Scoring with Kernel Learning Methods. *European Journal of Operational Research* 183(3), 1521–1536 (2007)
9. Yu, L., Wang, S., Lai, K.K.: An Intelligent-agent-based Fuzzy Group Decision Making Model for Financial Multicriteria Decision Support: The Case of Credit Scoring. *European Journal of Operational Research* 195, 942–959 (2009)

**Part III**  
**Image and Speech Analysis**



# A Three-Dimensional Neural Network Based Approach to the Image Reconstruction from Projections Problem

Robert Cierniak

Technical University of Czestochowa, Department of Computer Engineering,  
Armii Krajowej 36, 42-200 Czestochowa, Poland

**Abstract.** This paper presents a novel neural network approach to the problem of image reconstruction from projections obtained by spiral tomography scanner. The reconstruction process is performed during the minimizing of the energy function in recurrent neural network. Our method is of a great practical use in reconstruction from discrete cone-beam projections. Experimental results show that the appropriately designed neural network is able to reconstruct an image with better quality than obtained from used commercial conventional algorithms.

## 1 Introduction

Since the beginning of the 1990s, the use of spiral tomography scanners has been increasingly widespread (eg. [7]). The name comes from the shape of the path that the X-ray tube and its associated detector array follow with respect to the patient. With the development of *cone-beam computed tomography* (CBCT) there was a break with previous thinking, which led to a substantial increase in the width of the detector array. This in turn led to such an increase in scanning rate that it now became possible to scan organs physiologically in motion, such as the heart. Furthermore, because of the small distance between the rows of detectors, it was also possible to increase the scan resolution along the patient axis significantly. The possibility to acquire three-dimensional images of the investigated objects is realized by applying an appropriate method of projections acquisition and an appropriate image reconstruction algorithm. The key problem arising in computed tomography is image reconstruction from projections obtained from the x-ray scanner of a given geometry (in this case the cone-beam geometry with helical path of the x-ray tube. There are several reconstruction methods to solve this problem, for example the most popular reconstruction algorithms: the Feldcamp algorithm [8] and the reconstruction procedure described in the literature by its abbreviation ASSR (*advanced single slice rebinning*) [12].

Considering the increasing amount of soft computing algorithms used in different science disciplines, it is possible that in the foreseeable future they will occupy an important place in computerized tomography as well. The applications of neural networks in computerized tomography were presented in the past for example in [14], [15], [18]. The so-called neural algebraic approaches to reconstruction problem comparable to our algorithm are presented in papers [19], [20]. In this paper a new approach to the reconstruction problem will be developed based on transformation methodology. It resembles the traditional  $\rho$ -filtered layergram reconstruction method where the two-dimensional filtering is the crucial point of that approach [16]. Unfortunately,

two-dimensional filtering is computationally complex. Therefore, in our approach a recurrent neural network [9] is proposed to design the reconstruction algorithm. Some authors [1], [10] applied similar neural network structures to solve another problem, namely unidimensional signal reconstruction. Our approach significantly decreases the complexity of the tomographic reconstruction problem. The reconstruction method presented herein, originally formulated by the author, is applied to the cone-beam scanner geometry of the helical tomography device.

## 2 Image Reconstruction Algorithm

Shown in this paper an original reconstruction algorithm for cone-beam helical scanner is based on designed earlier neural network reconstruction method presented in papers

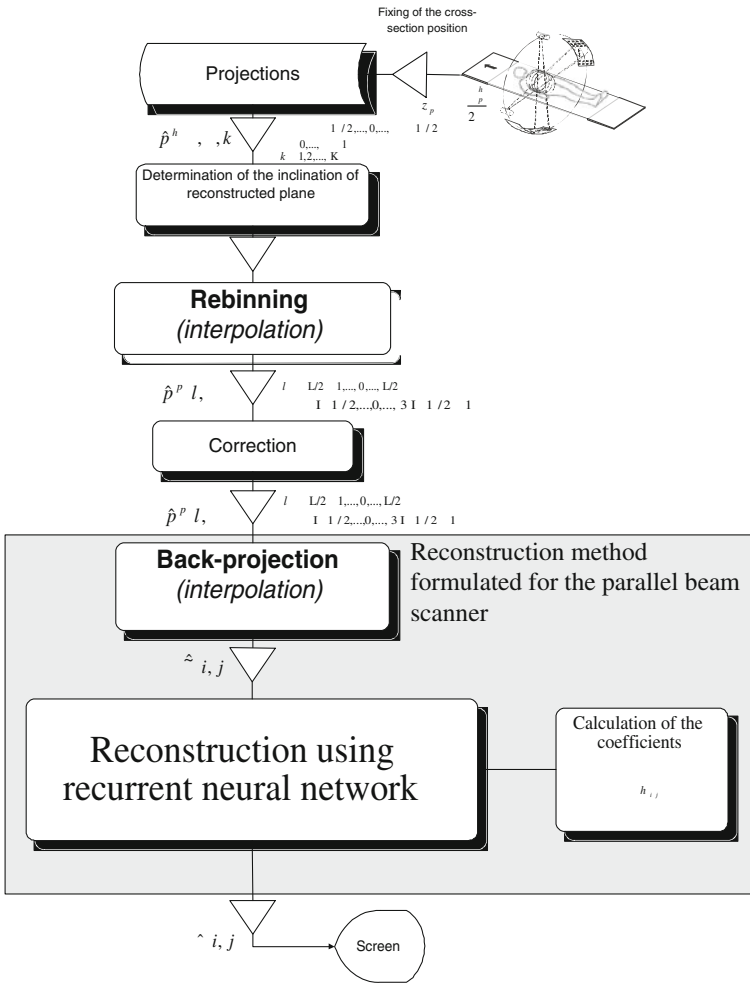
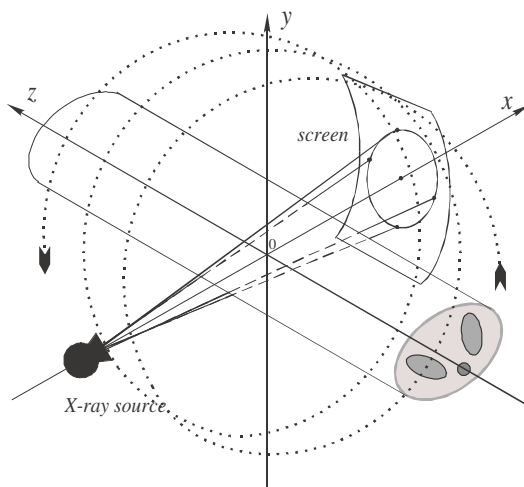


Fig. 1. A neural network image reconstruction algorithm with cone-beam geometry of the scanner



**Fig. 2.** The projection system of a cone-beam scanner – a three-dimensional perspective view

[3], [4], [5], [6]. Those solutions were dedicated for previous scanner geometries, that means for parallel and fan-beam geometries [11], [16]. The scheme of the proposed reconstruction method using the Hopfield-type neural network is shown in Fig 1, where the cone-beam geometry of collected projections is taken into consideration.

Presented in this paper reconstruction neural network algorithm is based on the advanced single slice rebinning (ASSR) algorithm [12]. The main difference between these two methods is a realization of the filtering. In our case the neural network is applied to two-dimensional filtering of the blurred image obtained after the back-projection operation instead of the one-dimensional filtering of the projections in ASSR method.

## 2.1 The Acquisition of the Projections

In the first step of the reconstruction algorithm a set of all fan-beams projections is collected using a scanner whose geometry is schematically depicted in Fig.2. Each ray emitted by the tube at a particular angle of rotation and reaching any of the radiation detectors can be identified by  $(\beta, \alpha^h, z)$  as follows:  $\beta$  is the angle between a particular ray in the beam and the axis of symmetry of the moving projection system,  $\alpha^h$  the angle at which the projection is made, i.e. the angle between the axis of symmetry of the rotated projection system and the y-axis,  $z$  the z-coordinate relative to the current position of the moving projection system.

Therefore, the projection function measured at the screen in a cone-beam system can be represented by  $p^h(\beta, \alpha^h, z)$ .

Before we start our reconstruction procedure we fix a point  $z_p$  on z-axis, where the cross-section will be reconstructed. The angle  $\alpha_p^h$  at which the projection system is at this moment positioned is described by following relation:

$$\alpha_p^h = z_p \frac{2\pi}{\lambda}, \quad (1)$$

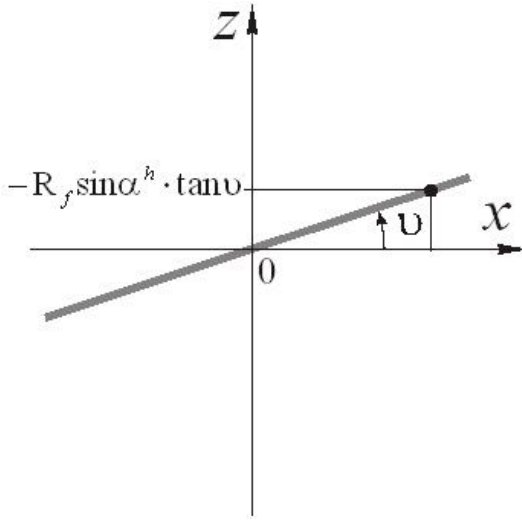


Fig. 3. The projection system of a cone-beam scanner – a three-dimensional perspective view

where:  $\lambda$  is the relative travel of the spiral described by the tube around the test object, measured in  $\left[\frac{\text{m}}{\text{rad}}\right]$ .

**2.2 Adjusting the Reconstruction Plane**

After selecting the angle of rotation  $\alpha_p^h$  of the spiral projection system in such way that the central ray of the beam intersects the z-axis at the midpoint of reconstructed slice, we then have to determine the inclination of the plane of the slice. The angle of inclination is represented by the symbol  $\nu$ , what is depicted in Fig. 3.

By quite complicated geometrical considerations, we obtain the following value for the optimum angle of inclination of the reconstruction plane:

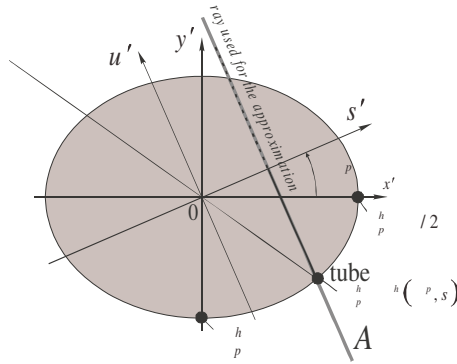
$$\nu = \arctan \left( \frac{\lambda \cdot \arccos \left( \frac{1}{2} (1 + \cos (\pi)) \right)}{2\pi R_f \sin \left( \arccos \left( \frac{1}{2} (1 + \cos (\pi)) \right) \right)} \right), \tag{2}$$

where:  $R_f$  is the radius of the circle described by the focus of the tube.

**2.3 Longitudinal Approximation**

Longitudinal approximation uses the fact that both the real ray obtained physically from the projection and the ray from the approximated projection are in the same plane parallel to the z-axis. If that plane is represented by the symbol  $A$ , then this property can be written as follows:

$$A \parallel \mathbf{z} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{3}$$



**Fig. 4.** The geometry of the longitudinal approximation

The position of plane A with respect to the reconstruction plane is shown in Figure 4.

The projection value measured when parallel X-rays pass through the reconstruction plane of a test object with attenuation function  $\mu(x, y, z)$  can be calculated from the formula:

$$p^{p'}(s', \alpha'^p) = \int \mu(x, y, z) \cdot \delta(x' \cos \alpha'^p + y' \sin \alpha'^p - s') dx' dy', \quad (4)$$

where:  $p^{p'}(s', \alpha'^p)$  is a virtual parallel projection in coordination system determined by reconstruction plane ( $(s', u')$  is coordination system rotated by angle  $\alpha'^p$ ;  $(x', y')$  is fixed coordination system).

Longitudinal approximation consists in determination of all needed for reconstruction procedure virtual parallel projections in reconstruction plane based on projections obtained in cone-beam helical scanner. This process is showed in details in [12] and [2]. Now since we have  $p^{p'}(s', \alpha'^p)$ , we can proceed to the final signal processing stages of our neural network reconstruction method, that is, the use of reconstruction method originally devised for a parallel projection system.

### 2.4 The Back-Projection Operation

The next step in the proceeding sequence is the back-projection operation [11], [16]. In practical realization of the proposed reconstruction algorithm it is highly possible that for any given projection no ray passes through a certain point  $(x', y')$  of the image. To take this into account we can apply interpolation expressed by the equation

$$\dot{p}'(s', \alpha'^p) = \int_{-\infty}^{+\infty} p^{p'}(s', \alpha'^p) \cdot I(s' - s') ds', \quad (5)$$

where  $I(\Delta s')$  is an interpolation function. Now the back-projection operation can be expressed as

$$\tilde{\mu}(x', y') = \int_0^{\pi} \dot{p}'(s', \alpha'^p) d\alpha'^p. \quad (6)$$

Function  $\tilde{\mu}(x', y')$  denotes a blurred image obtained after operations of projection and back-projection.

Owing to relation (4), (5) and (6) it is possible to define the obtained, after back-projection operation, image in the following way

$$\tilde{\mu}(x', y') = \int_0^\pi \int_{-\infty}^{+\infty} \left( \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \mu(\ddot{x}, \ddot{y}) \cdot \delta(\ddot{x} \cos \alpha'^p + \ddot{y} \sin \alpha'^p - s') d\ddot{x} d\ddot{y} \right) \cdot I(s' - s') ds d\alpha'^p. \tag{7}$$

According to the properties of the convolution we can transform formula (7) to the form

$$\tilde{\mu}(x', y') = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \mu(\ddot{x}, \ddot{y}) \left( \int_0^\pi I(\ddot{x} \cos \alpha'^p + \ddot{y} \sin \alpha'^p - x' \cos \alpha'^p - y' \sin \alpha'^p) d\alpha'^p \right) d\ddot{x} d\ddot{y}. \tag{8}$$

### 2.5 Discrete Reconstruction Problem

In presented method we take into consideration the discrete form of images  $\mu(x, y)$  and  $\tilde{\mu}(x, y)$ . That means we will substitute continuous functions of images in equation (6) for their discrete equivalents  $\hat{\mu}(i, j)$  and  $\hat{\tilde{\mu}}(i, j)$ ;  $i = 0, 1, \dots, I$ ;  $j = 0, 1, \dots, J$ , where  $I$  and  $J$  are numbers of pixels in horizontal and vertical directions, respectively. Additionally, we approximate the 2-D convolution function by two finite sums. In this way we express relation (8) in the following form

$$\hat{\tilde{\mu}}(i, j) \approx \sum_i \sum_j \hat{\mu}(i - \ddot{i}, j - \ddot{j}) \cdot h_{i\ddot{j}}, \tag{9}$$

where

$$h_{i\ddot{j}} = \Delta_\alpha^p (\Delta_s)^2 \cdot \sum_{\psi=0}^{\psi-1} I(\ddot{i} \Delta_s \cos \psi \Delta_\alpha^p + \ddot{j} \Delta_s \sin \psi \Delta_\alpha^p). \tag{10}$$

As one can see from equation (9), the original image in a given cross-section of the object, obtained in the way described above, is equal to the amalgamation of this image and the geometrical distortion element given by (10). The number of coefficients  $h_{i\ddot{j}}$  is equal to  $I \cdot J$  and owing to expression (10) values of these coefficients can be easily calculated.

The discrete reconstruction from projections problem can be formulated as the following optimisation problem [17]

$$\min_{\Omega} \left( p \cdot \sum_{i=1}^I \sum_{j=1}^J f(e_{i\ddot{j}}(\Omega)) \right), \tag{11}$$

where:  $\Omega = [\hat{\mu}(i, j)]$ —a matrix of pixels from original image;  $p$ —suitable large positive coefficient;  $f(\bullet)$ —penalty function and

$$e_{i\ddot{j}}(\Omega) = \sum_i \sum_j \hat{\mu}(i, j) \cdot h_{i-i, \ddot{j}-j} - \hat{\tilde{\mu}}(i, \ddot{j}). \tag{12}$$

If a value of coefficient  $p$  tends to infinity or in other words is suitably large, then the solution of (11) tends to the optimal result. Our research has shown that the following penalty function yields the best result

$$f(e_{ij}(\Omega)) = \beta \cdot \ln \cosh\left(\frac{e_{ij}(\Omega)}{\lambda}\right), \tag{13}$$

and derivation of (13) has the convenient form

$$f'(e_{ij}(\Omega)) = \frac{df(e_{ij}(\Omega))}{de_{ij}(\Omega)} = \frac{1 - \exp(e_{ij}(\Omega) / \beta)}{1 + \exp(e_{ij}(\Omega) / \beta)}, \tag{14}$$

where:  $\beta$ —slope coefficient.

### 2.6 Reconstruction Process Using Recurrent Neural Network

Now we can start to formulate the energy expression

$$E^t = w \cdot \sum_{i=1}^I \sum_{j=1}^J f(e_{ij}(\Omega^t)). \tag{15}$$

which will be minimized by the constructed neural network to realize the deconvolution task expressed by equation (9). In order to find a minimum of function (15) we calculate the derivation

$$\frac{dE^t}{dt} = w \cdot \sum_{i=1}^I \sum_{j=1}^J \sum_{i=1}^I \sum_{j=1}^J \frac{\partial f(e_{ij}(\Omega^t))}{\partial(e_{ij}(\Omega^t))} \frac{\partial(e_{ij}(\Omega^t))}{\partial \hat{\mu}^t(i, j)} \frac{d\hat{\mu}^t(i, j)}{dt}. \tag{16}$$

If we let

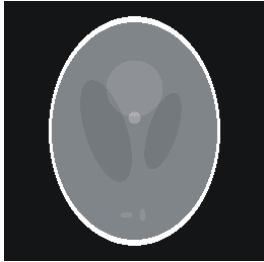


$$\frac{d\hat{\mu}^t(i, j)}{dt} = -w \sum_{i=1}^I \sum_{j=1}^J \frac{\partial f(e_{ij}(\Omega^t))}{\partial(e_{ij}(\Omega^t))} \frac{\partial(e_{ij}(\Omega^t))}{\partial \hat{\mu}^t(i, j)} = -w \sum_{i=1}^I \sum_{j=1}^J f'(e_{ij}(\Omega)) h_{ij}, \tag{17}$$

equation (16) takes the form

$$\frac{dE^t}{dt} = - \sum_{i=1}^I \sum_{j=1}^J \left(\frac{d\hat{\mu}^t(i, j)}{dt}\right)^2. \tag{18}$$

## 3 Experimental Results

A mathematical model of the projected object, a so-called phantom, is used to obtain projections during simulations. The most common mathematical phantom of head was proposed by Kak (see eg. (11)). In our experiment the size of the image was fixed at  $I \times J = 129 \times 129$  pixels.

|       | a                                                                                 | b                                                                                 | c                                                                                 |
|-------|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|
| Image |  |  |  |
| MSE   | –                                                                                 | 0.01204                                                                           | 0.0928                                                                            |
| Error | –                                                                                 | 0.2428                                                                            | 0.2401                                                                            |

**Fig. 5.** View of the images: a) original image; b) reconstructed image using standard reconstruction method (convolution/back-projection with rebinning method); c) neural network reconstruction algorithm described in this paper

The discret approximation of the interpolation operation expressed by equation (5) takes the form

$$\hat{p}^p(s, \psi) = \sum_{l=-L/2}^{L/2-1} \hat{p}^p(l, \psi) \cdot I(s - l\Delta_s^p). \tag{19}$$

The interpolation function  $I(\Delta s)$  can be defined for example as linear interpolation function

$$I_L(\Delta s) = \begin{cases} \frac{1}{\Delta_s} \left(1 - \frac{|\Delta s|}{\Delta_s}\right) & \text{if } |\Delta s| \leq \Delta_s \\ 0, & \text{if } |\Delta s| > \Delta_s \end{cases}, \tag{20}$$

where  $\Delta s = (i \cos \psi \Delta_\alpha^p + j \sin \psi \Delta_\alpha^p)$ .

The image was next subjected to a process of reconstruction using the recurrent neural network presented in section 2. The Euler’s method (see eg. [20]) was used to approximate (17) in following way

$$\mu(i, j)^{t+1} := \mu(i, j)^t + \Delta t \left( -w \sum_{i=1}^I \sum_{j=1}^J f' \left( e_{ij}(\Omega) \right) h_{ij} \right), \tag{21}$$

where  $e_{ij}$  is expressed by (12) and  $\Delta t$  is a sufficient small time step.

The difference between reconstructed images using the recurrent neural network algorithm described in this paper (Fig. 5a) and the standard convolution/back-projection method (Fig. 5b) is depicted below. The quality of the reconstructed image has been evaluated in this case by error measures defined as follows

$$MSE = \frac{1}{I \cdot J} \sum_{i=1}^I \sum_{j=1}^J [\mu(i, j) - \hat{\mu}(i, j)]^2 \tag{22}$$



and

$$ERROR = \left[ \frac{\sum_{i=1}^I \sum_{j=1}^J (y(i, j) - \hat{y}(i, j))^2}{\sum_{i=1}^I \sum_{j=1}^J (y(i, j) - \bar{y}(i, j))^2} \right]^{1/2}, \quad (23)$$

where  $y(i, j)$ ,  $\hat{y}(i, j)$  and  $\bar{y}(i, j)$  are the image of the Shepp-Logan phantom, the reconstructed image and the mean of the Shepp-Logan image, respectively. These images are obtained using window described in [13]. In this case were used following parameters of this window  $C = 1.02$ ,  $W = 0.2$ .

## 4 Conclusions

The performed simulations demonstrated a stability of the image reconstruction from projections algorithm based on the recurrent neural network described in this work. The reconstructed images obtained after thirty thousand iterations showed in Fig 5 have better level of quality in comparison to the result of the standard ASSR method with the same parameters of the obtained 3D projections. Although our procedure is time consuming, the hardware implementation of the described neural network structure could give incomparable better results than other reconstruction methods.

## Acknowledgments

This work was partly supported by Polish Ministry of Science and Higher Education (Research Project N N516 185235).

## References

1. Cichocki, A., Unbehauen, R., Lendl, M., Weinzierl, K.: Neural networks for linear inverse problems with incomplete data especially in application to signal and image reconstruction. *Neurocomputing* 8, 7–41 (1995)
2. Cierniak, R.: Computed tomography. Academic Publishing House EXIT, Warsaw (2005)
3. Cierniak, R.: A new approach to image reconstruction from projections problem using a recurrent neural network. *Applied Mathematics and Computer Science* 18, 147–157 (2008)
4. Cierniak, R.: A new approach to tomographic image reconstruction using a Hopfield-type neural network. *International Journal Artificial Intelligence in Medicine* 43, 113–125 (2008)
5. Cierniak, R.: A novel approach to image reconstruction from fan-beam projections using recurrent neural network. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008. LNCS (LNAI)*, vol. 5097, pp. 752–761. Springer, Heidelberg (2008)
6. Cierniak, R.: New neural network algorithm for image reconstruction from fan-beam projections. *Elsevier Science: Neurocomputing* 72, 3238–3244 (2009)
7. Crawford, C.R., King, K.F.: Computer tomography scanning with simultaneous patient translation. *Medical Physics* 17, 967–981 (1990)
8. Feldkamp, L.A., Davis, L.C., Kress, J.W.: Practical cone-beam algorithm. *J. Optical Society of America* 1(A), 612–619 (1984)
9. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. *Proc. National Academy of Science USA* 79, 2554–2558 (1982)

10. Ingman, D., Merlis, Y.: Maximum entropy signal reconstruction with neural networks. *IEEE Trans. on Neural Networks* 3, 195–201 (1992)
11. Jain, A.K.: *Fundamentals of Digital Image Processing*. Prentice Hall, New Jersey (1989)
12. Kachelrieß, M., Schaller, S., Kalender, W.A.: Advanced single-slice rebinning in cone-beam spiral CT. *Medical Physics* 27, 754–773 (2000)
13. Kak, A.C., Slanley, M.: *Principles of Computerized Tomographic Imaging*. IEEE Press, New York (1988)
14. Kerr, J.P., Bartlett, E.B.: A statistically tailored neural network approach to tomographic image reconstruction. *Medical Physics* 22, 601–610 (1995)
15. Knoll, P., Mirzaei, S., Mueller, A., Leitha, T., Koriska, K., Koehn, H., Neumann, M.: An artificial neural net and error backpropagation to reconstruct single photon emission computerized tomography data. *Medical Physics* 26, 244–248 (1999)
16. Lewitt, R.M.: Reconstruction algorithms: transform methods. *Proceeding of the IEEE* 71, 390–408 (1983)
17. Luo, F.-L., Unbehauen, R.: *Applied Neural Networks for Signal Processing*. Cambridge University Press, Cambridge (1998)
18. Munlay, M.T., Floyd, C.E., Bowsher, J.E., Coleman, R.E.: An artificial neural network approach to quantitative single photon emission computed tomographic reconstruction with collimator, attenuation, and scatter compensation. *Medical Physics* 21, 1889–1899 (1994)
19. Srinivasan, V., Han, Y.K., Ong, S.H.: Image reconstruction by a Hopfield neural network. *Image and Vision Computing* 11, 278–282 (1993)
20. Wang, Y., Wahl, F.M.: Vector-entropy optimization-based neural-network approach to image reconstruction from projections. *IEEE Transaction on Neural Networks* 8, 1008–1014 (1997)

# Spatial Emerging Patterns for Scene Classification

Łukasz Kobyliński and Krzysztof Walczak

Institute of Computer Science, Warsaw University of Technology  
ul. Nowowiejska 15/19, 00-665 Warszawa, Poland  
{L.Kobyliniski,K.Walczak}@ii.pw.edu.pl

**Abstract.** In this paper we propose a novel method of scene classification, based on the idea of mining emerging patterns between classes of images, represented in a symbolic manner. We use the 9DLT (Direction Lower Triangular) representation of images, which allows to describe scenes with a limited number of symbols, while still capturing spatial relationships between objects visible on the images. We show an efficient method of mining the proposed Spatial Emerging Patterns and present results of synthetic image classification experiments.

## 1 Introduction

The approaches to mining data in image databases vary greatly in the way spatial data is represented and used for reasoning about its content. Recently, the attention is aimed at extracting local features from images, predominantly using the SIFT descriptor [1] and its descendants. For example in [2] the authors represent the image as a collection of local regions, extracted using a feature descriptor and clustered into a chosen number of unique codewords. Another recent advancement in image mining consists of analyzing the spatial relationships between local features and taking into consideration the context of identified objects. In [3] the images are partitioned into several layers of sub-regions and histograms of local features inside the sub-regions are calculated. This extension of a bag-of-words model makes an assumption that similar parts of a scene always occur in the same parts of the image grid. In the approach chosen in [4] the object context is taken into consideration by using a conditional random-field framework and this way the categorization accuracy is greatly improved.

Independently from image understanding methods, a great number of data mining approaches for transactional databases have been developed in recent years. The association rules and, more recently, emerging patterns, are just two examples of fruitful research in the area of data analysis. As these methods proved to perform well in the area of market basket analysis, text analysis and mining of other symbolic data, a question arises whether they can be used in image understanding. Such an application requires that the images are represented symbolically, usually by first extracting their features and employing an unsupervised learning method to get a number of codewords used to describe the scene. The representation may be created at various concept levels, however, beginning with individual pixels, through low-level features to real-world

objects. Some of the first applications of data mining methods to discovery of knowledge in image databases has been proposed in [5], where association rules are mined between objects appearing in an image. The spatial relationships between features and objects have also been incorporated into proposed mining methods. One approach is to include the information about spatial context into the symbolic representation itself. In the 9DLT representation [6] the relationships between objects are denoted by associating directional codes with pairs of items, which provide information about the angle between two image features.

In our work we employ the very promising idea of mining emerging patterns in an image database consisting of scenes with identified objects, described symbolically with the 9DLT representation. In such a framework we may reason about spatial arrangement of objects visible on an image accurately and efficiently and use this knowledge in classification of scenes. This is done by mining a new type of emerging patterns – jumping spatial emerging patterns – in a database of symbolically represented training images with known category labels and then using this knowledge to classify previously unknown scenes.

## 2 Previous Work

The idea of using jumping emerging patterns for classification and their mining algorithm has been first proposed in [7]. Efficient discovery of emerging patterns has been studied in [8], while in [9] an efficient algorithm of mining jumping emerging patterns with recurrent items has been proposed. Such patterns proved to be useful in classification of multimedia data.

A method of discovering spatial association rules in image data has been proposed in [10]. The authors have used the 9DLT symbolic representation of images to allow mining interesting association rules between objects appearing in a visual database.

## 3 Emerging Patterns

Emerging patterns may be briefly described as patterns, which occur frequently in one set of data and seldomly in another. We now give a formal definition of emerging patterns in transaction systems.

Let a transaction system be a pair  $(\mathcal{D}, \mathcal{I})$ , where  $\mathcal{D}$  is a finite sequence of transactions  $(T_1, \dots, T_n)$  (database), such that  $T_i \subseteq \mathcal{I}$  for  $i = 1, \dots, n$  and  $\mathcal{I}$  is a non-empty set of items (itemspace). A support of an itemset  $X \subset \mathcal{I}$  in a sequence  $D = (T_i)_{i \in K \subseteq \{1, \dots, n\}} \subseteq \mathcal{D}$  is defined as:  $\text{supp}_D(X) = \frac{|\{i \in K : X \subseteq T_i\}|}{|K|}$ .

Let a decision transaction system be a tuple  $(\mathcal{D}, \mathcal{I}, \mathcal{I}_d)$ , where  $(\mathcal{D}, \mathcal{I} \cup \mathcal{I}_d)$  is a transaction system and  $\forall T \in \mathcal{D} |T \cap \mathcal{I}_d| = 1$ . Elements of  $\mathcal{I}$  and  $\mathcal{I}_d$  are called condition and decision items, respectively. A support for a decision transaction system  $(\mathcal{D}, \mathcal{I}, \mathcal{I}_d)$  is understood as a support in the transaction system  $(\mathcal{D}, \mathcal{I} \cup \mathcal{I}_d)$ .

For each decision item  $c \in \mathcal{I}_d$  we define a decision class sequence  $C_c = (T_i)_{i \in K}$ , where  $K = \{k \in \{1, \dots, n\} : c \in T_k\}$ . Notice that each of the transactions

from  $\mathcal{D}$  belongs to exactly one class sequence. In addition, for a database  $D = (T_i)_{i \in K \subseteq \{1, \dots, n\}} \subseteq \mathcal{D}$ , we define a complement database  $D' = (T_i)_{i \in \{1, \dots, n\} - K}$ .

Given two databases  $D_1, D_2 \subseteq \mathcal{D}$  the growth rate of an itemset  $X \subset \mathcal{I}$  from  $D_1$  to  $D_2$  is defined as:

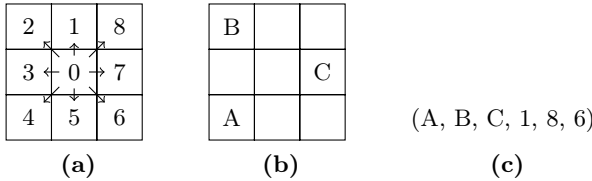
$$GR_{D_1 \rightarrow D_2}(X) = \begin{cases} 0 & \text{if } \text{supp}_{D_1}(X) = 0 \text{ and } \text{supp}_{D_2}(X) = 0, \\ \infty & \text{if } \text{supp}_{D_1}(X) = 0 \text{ and } \text{supp}_{D_2}(X) \neq 0, \\ \frac{\text{supp}_{D_2}(X)}{\text{supp}_{D_1}(X)} & \text{otherwise.} \end{cases} \quad (1)$$

Given a minimum growth rate  $\rho$ , we define an itemset  $X \subset \mathcal{I}$  to be a  $\rho$ -emerging pattern ( $\rho$ -EP) from  $D_1$  to  $D_2$  if  $GR_{D_1 \rightarrow D_2}(X) > \rho$ . Furthermore, we say that an itemset  $X$  is a jumping emerging pattern (JEP), when its growth rate is infinite, that is  $GR_{D_1 \rightarrow D_2}(X) = \infty$ . Having a minimum support threshold  $\xi$ , we define a  $\xi$ -strong jumping emerging pattern to be a JEP from  $D_1$  to  $D_2$  for which  $\text{supp}_{D_1}(X) = 0$  and  $\text{supp}_{D_2}(X) > \xi$ . A set of all JEPs from  $D_1$  to  $D_2$  is called a JEP space and denoted by  $JEP(D_1, D_2)$ .

### 4 Image Representation

We use the 9DLT string representation of images to capture the spatial arrangement of objects visible in a scene. The symbolic representation, which consists of object labels and directional codes indicating spatial relationships between them, allows us to use data mining methods to reason about large image databases.

The 9DLT representation defines nine directional codes,  $\mathcal{R} = \{0, 1, \dots, 8\}$ , which are an equivalent of a range of angles between two objects in a scene. Figure 1(a) depicts the use of codes: "0" means "the same spatial location as", "1" means "the north of", "2" means "the north-west of", and so on.



**Fig. 1.** The 9DLT representation: (a) directional codes, (b) example scene, (c) its symbolic representation

We now use the definition of a spatial pattern, presented in [10], to extend the definition of our transactional system. A spatial pattern  $X^s$  is defined as a pattern of a form  $X^s = (i_1, i_2, \dots, i_n, r_1, r_2, \dots, r_m)$ , where  $i_j \in \mathcal{I}$  are items and  $r_k \in \mathcal{R}$  are directional codes. Here,  $m = C_2^n = n(n-1)/2$ ,  $1 \leq j \leq n$ ,  $1 \leq k \leq m$  and  $n \geq 2$ . Each of the directional codes denotes a spatial relationship between two corresponding items, taken from left to right, e.g. the relationship between  $i_1$  and  $i_2$  is  $r_1$ , while between  $i_1$  and  $i_3$  is  $r_2$ .

*Example 1.* Consider the image presented on Fig. [1b](#). Its symbolic representation as a spatial pattern takes the form shown on Fig. [1c](#)

We say that spatial pattern  $Y^s = (i'_1, i'_2, \dots, i'_n, r'_1, r'_2, \dots, r'_m)$  is a sub-pattern of a pattern  $X^s = (i_1, i_2, \dots, i_n, r_1, r_2, \dots, r_m)$ , denoted as  $Y^s \sqsubseteq X^s$ , when  $\{i'_1, i'_2, \dots, i'_n\} \subseteq \{i_1, i_2, \dots, i_n\}$  and each spatial relationship between every two items is exactly the same in both patterns. Furthermore, we say that two spatial relationships  $r_i, r_j \neq 0$  are complementary, when  $r_i = (r_j + 4) \bmod 8$ .

*Example 2.* Consider the following 4-element spatial pattern:  $X^s = (A, B, C, D, 8, 7, 8, 6, 1, 2)$ . There are four 3-element sub-patterns of pattern  $X^s$ :  $Y_1^s = (A, B, C, 8, 7, 6)$ ,  $Y_2^s = (A, B, D, 8, 8, 1)$ ,  $Y_3^s = (A, C, D, 7, 8, 2)$  and  $Y_4^s = (B, C, D, 6, 1, 2)$ .

A spatial transactional system is a pair  $(\mathcal{D}^s, \mathcal{I})$ , where  $\mathcal{D}^s$  is a finite sequence of transactions  $(T_1^s, \dots, T_n^s)$  for  $i = 1, \dots, n$ . Each transaction is a pattern  $(i_1, i_2, \dots, i_n, r_1, r_2, \dots, r_m)$ , where  $i_j \in \mathcal{I}$  are items and  $r_k \in \mathcal{R}$  are directional codes. A support of a spatial pattern  $X^s$  in a sequence  $\mathcal{D}^s = (T_i^s)_{i \in K \subseteq \{1, \dots, n\}} \subseteq \mathcal{D}^s$  is defined as:

$$\text{supp}_{\mathcal{D}^s}(X) = \frac{|\{i \in K : X^s \sqsubseteq T_i^s\}|}{|K|}. \tag{2}$$

## 5 Spatial Emerging Patterns

### 5.1 Formal Definition

Based on the earlier definitions, we now define a new kind of patterns, namely Spatial Emerging Patterns (SEPs), which are able to capture interesting differences between sets of spatial data. Given two spatial databases  $D_1^s$  and  $D_2^s$ , we define the growth rate of a pattern  $X^s$  in the same way as stated by Eq. [1](#), in which we use the definition of support presented by Eq. [2](#). Having a minimum growth rate  $\rho$ , we define a pattern  $X^s$  to be a  $\rho$ -spatial emerging pattern ( $\rho$ -SEP) from  $D_1^s$  to  $D_2^s$  if  $GR_{D_1^s \rightarrow D_2^s}(X) > \rho$ . The definition of a jumping spatial emerging pattern (JSEP) and a  $\xi$ -strong jumping spatial emerging pattern is analogous to the one proposed for regular EPs.

We may introduce another way of representing spatial emerging patterns, which shows the connection between SEPs and regular emerging patterns. By enumerating all encoded relationships and creating unique item for each of them, we get a new space of items, which is defined as  $\mathcal{I}' = \mathcal{I} \times \mathcal{R} \times \mathcal{I}$ . Formally, we can say that each pattern of a form  $X^s = (i_1, i_2, \dots, i_n, r_1, r_2, \dots, r_m)$  may be represented as:  $X^s = (i_1 i_2 r_1, i_1 i_3 r_2, \dots, i_1 i_n r_k, \dots, i_{n-1} i_n r_m)$ .

*Example 3.* A pattern  $X^s = (A, B, C, 1, 8, 6)$  may also be represented as  $X^s = (AB1, AC8, BC6)$ , written for convenience as  $X^s = (A_1 B, A_8 C, B_6 C)$ .

It is important to note, that while all patterns may be represented in the second manner, not all patterns may be described in the original, shortened form. It is the case when not all relationships between particular items are known.

*Example 4.* Consider two sets of spatial data, represented by 9DLT patterns:  $D_1 = ((A, B, C, 1, 8, 6)) = ((A_1B, A_8C, B_6C))$  and  $D_2 = ((A, B, 1), (A, C, 8), (B, C, 7)) = ((A_1B), (A_8C), (B_7C))$ . We mine strong JSEPs between these sets by looking for minimal patterns, which occur in one set and never in the other. In the case of JSEPs from  $D_1$  to  $D_2$  we have  $JSEP_1 = (B, C, 6) = (B_6C)$  and  $JSEP_2 = (A, B, C, 1, 8, ?) = (A_1B, A_8C)$ . Similarly, in the direction of  $D_2$  to  $D_1$  we have  $JSEP_3 = (B, C, 7) = (B_7C)$ .

### 5.2 Mining Algorithm

In our approach, we are only interested in mining patterns for the use in building classifiers. For that reason we may limit ourselves to mining only strong jumping spatial emerging patterns, that is JSEPs, which are minimal and have a specified minimum support in one of the databases.

An efficient algorithm for mining emerging patterns has been presented in [8], which introduces the notion of borders to represent a large number of patterns. A border is an ordered pair  $\langle \mathcal{L}, \mathcal{R} \rangle$  such that  $\mathcal{L}$  and  $\mathcal{R}$  are antichains,  $\forall X^s \in \mathcal{L} \exists Y^s \in \mathcal{R} X^s \subseteq Y^s$  and  $\forall X^s \in \mathcal{R} \exists Y^s \in \mathcal{L} Y^s \subseteq X^s$ . The collection of sets represented by a border  $\langle \mathcal{L}, \mathcal{R} \rangle$  is equal to:

$$[\mathcal{L}, \mathcal{R}] = \{Y^s : \exists X^s \in \mathcal{L}, \exists Z^s \in \mathcal{R} \text{ such that } X^s \subseteq Y^s \subseteq Z^s\}. \tag{3}$$

The left border in this representation corresponds to minimal patterns found in a particular dataset. As such, we may follow the methodology of finding only the left border of the set of jumping spatial emerging patterns between the two databases. Having databases  $D_1$  and  $D_2$  this may be performed by subtracting all patterns in  $D_2$  from each of the patterns in  $D_1$  and vice versa. By aggregating all the resulting patterns, we get the set of minimal JSEPs from  $D_2$  to  $D_1$  and from  $D_1$  to  $D_2$  respectively.

The straightforward way of calculating this differential would be to find all sub-patterns of each of the database transactions and eliminate these patterns in  $R_1$ , which also occur in  $R_2$ . To avoid the cost of checking all possible relationships in both databases, an iterative procedure may be used, which comes from the idea presented in [8]. It has been shown there that the collection of minimal itemsets  $\text{Min}(\mathcal{S})$  in a border differential  $\mathcal{S} = [\{\emptyset\}, \{U^s\}] - [\{\emptyset\}, \{S_1^s, \dots, S_k^s\}]$  is equivalent to:  $\text{Min}(\{\bigcup\{s_1, \dots, s_k\} : s_i \in U^s - S_i^s, 1 \leq i \leq k\})$ .

We may iteratively expand candidate patterns and check if they are minimal, avoiding in this way generating many unnecessary patterns. The complete procedure is presented as Algorithm 1 below. We need to iteratively call the Border-differential function and create a union of the results to find the set of all minimal jumping spatial emerging patterns from  $C'_c$  to  $C_c$ .

### 5.3 Scene Classification Using Spatial Emerging Patterns

Having discovered spatial patterns on the basis of the learning set, we may use the built classifier to categorize previously unseen images from the testing set. To classify a particular scene, we first transform the image into its symbolic form,

**Algorithm 1.** Border differential

---

```

Input : $\langle \{\emptyset\}, \{U^s\} \rangle, \langle \{\emptyset\}, \{S_1^s, \dots, S_k^s\} \rangle$
Output: \mathcal{L}
1 $T_i^s \leftarrow U^s - S_i^s$ for $1 \leq i \leq k$
2 if $\exists T_i^s = \{\emptyset\}$ then
3 | return $\langle \{\}, \{\} \rangle$
4 end
5 $\mathcal{L} \leftarrow \{\{x\} : x \in T_1^s\}$
6 for $i = 2$ to k do
7 | $NewL \leftarrow \{X^s \in \mathcal{L} : X^s \cap T_i^s \neq \emptyset\}$
8 | $\mathcal{L} \leftarrow \mathcal{L} - NewL$
9 | $T_i^s \leftarrow T_i^s - \{x : \{x\} \in NewL\}$
10 | foreach $X^s \in \mathcal{L}$ sorted according to increasing cardinality do
11 | | foreach $x \in T_i$ do
12 | | | if $\forall Z^s \in NewL$ $\text{supp}_{Z^s}(X^s \cup \{x\}) = 0$ then
13 | | | | $NewL \leftarrow NewL \cup (X^s \cup \{x\})$
14 | | | | end
15 | | | end
16 | | end
17 | $\mathcal{L} \leftarrow NewL$
18 end

```

---

using the 9DLT representation. Next, we aggregate all minimal JSEPs, which are supported by the representation. A scoring function is calculated and a category label is chosen by finding the class with the maximum score:  $\text{score}(T^s, c) = \sum_{X^s} \text{supp}_{C_c}(X^s)$ , where  $C_c \subseteq \mathcal{D}_T^s$  and  $X^s \in \text{JSEP}_m(C'_c, C_c)$ , such that  $X^s \subseteq T^s$ .

## 6 Experimental Results

To assess the effectiveness of the proposed method, we have performed experiments using synthetic data to build and then test the JSEP-based classifiers. The data is prepared as follows: we generate two classes of transactions, consisting of uniformly distributed objects in a  $n \times n$  image. For each of the classes a characteristic pattern of size  $m \times m$ ,  $m < n$  is randomly constructed and overlaid on each of the random images. Finally, the images are transformed to 9DLT strings. This way we can assess the performance of the described classification method in recognizing the differentiating pattern in a set of otherwise random data. Apart from the image and pattern sizes the following parameters of the data generator may be changed freely: number of available objects ( $K$ ), number of objects on a single image ( $L$ ) and number of transactions in each of the classes ( $D$ ). Having generated the synthetic dataset we perform a ten-fold cross-validation experiment, by first discovering the jumping spatial emerging patterns in 90% of available data and testing the accuracy of classification in the other 10%. This procedure is repeated 10 times and an average accuracy is presented in results below.



**Table 1.** Classification accuracy of the synthetic dataset with relation to image and pattern sizes

| Image size<br>( $n$ ) | Accuracy<br>(%) | Time<br>(ms) |
|-----------------------|-----------------|--------------|
| 4                     | 95,50           | 1738         |
| 5                     | 92,20           | 2790         |
| 6                     | 94,30           | 3218         |
| 7                     | 92,70           | 3607         |
| 8                     | 95,00           | 3752         |
| 9                     | 93,10           | 3934         |
| 10                    | 92,90           | 3653         |

| Pattern size<br>( $m$ ) | Accuracy<br>(%) | Time<br>(ms) |
|-------------------------|-----------------|--------------|
| 2                       | 82,00           | 2397         |
| 3                       | 95,00           | 3545         |
| 4                       | 98,00           | 5840         |
| 5                       | 98,50           | 8109         |

$K = 10, L = 5, D = 100, m = 3$

**Table 2.** Classification accuracy of the synthetic dataset with relation to the number of available objects and objects on a single image

| Object space<br>( $K$ ) | Accuracy<br>(%) | Time<br>(ms) |
|-------------------------|-----------------|--------------|
| 10                      | 93,00           | 3663         |
| 15                      | 97,80           | 2604         |
| 20                      | 98,30           | 1637         |
| 25                      | 98,50           | 1264         |
| 30                      | 99,50           | 935          |
| 40                      | 99,83           | 923          |
| 50                      | 100,00          | 911          |

| Number of objects<br>( $L$ ) | Accuracy<br>(%) | Time<br>(ms) |
|------------------------------|-----------------|--------------|
| 3                            | 96,70           | 376          |
| 4                            | 96,00           | 1549         |
| 5                            | 92,40           | 3663         |
| 6                            | 86,00           | 9116         |
| 7                            | 81,20           | 18968        |
| 8                            | 78,50           | 42730        |

$L = 5, D = 100, n = 8, m = 3$

Firstly, we have experimented with the influence of the relation between pattern and image sizes on classification accuracy and the time needed to mine spatial patterns. The results are presented in Table 1 and show an increase of accuracy when pattern size approaches the size of the image. This is because there is relatively less random noise in the generated data in comparison to the differentiating pattern. The image size alone however, does not directly influence the classification accuracy or pattern mining time, as it has no relation to the size of 9DLT representation and number of discovered JSEPs.

The influence of the size of object space and the number of objects appearing on a particular image on classification results may be assessed from the data in Table 2. We can see that increasing the object space size while maintaining a constant number of objects results in better classification accuracy and less time needed to mine JSEPs, while increasing the number of objects having a constant objects space size has an opposite effect. The number of objects on individual images has a direct impact on the average length of the 9DLT representation and thus the transaction size. As the average transaction length increases, the

number of patterns becomes larger and so does the ratio between noise and the patterns which allow to differentiate between classes.

## 7 Conclusions and Future Work

In this paper we have introduced Spatial Emerging Patterns, a new data mining method of discovering knowledge in image databases and its use in classification of scenes. The presented results of the experiments look promising and show that the method may be used to classify image data, in which a preliminary object recognition step has been performed. Such images may be transformed into 9DLT representation and used as a basis for JSEP mining and classification. Other symbolic representations may be proposed in the place of 9DLT and it remains a further work to assess the influence of the representation methods used on overall accuracy.

## References

1. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60(2), 91–110 (2004)
2. Li, F.F., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: *Proc. 2005 IEEE Conf. Computer Vision and Pattern Recognition (CVPR 2005)*, Washington, DC, USA, pp. 524–531. IEEE Computer Society, Los Alamitos (2005)
3. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: *Proc. 2006 IEEE Conf. Computer Vision and Pattern Recognition (CVPR 2006)*, Washington, DC, USA, pp. 2169–2178. IEEE Computer Society, Los Alamitos (2006)
4. Rabinovich, A., Vedaldi, A., Galleguillos, C., Wiewiora, E., Belongie, S.: Objects in context. In: *11th Int. Conf. Comp. Vision (ICCV 2007)*, Rio de Janeiro, pp. 1–8 (2007)
5. Ordóñez, C., Omiecinski, E.: Discovering association rules based on image content. In: *Proc. IEEE Forum on Research and Technology Advances in Digital Libraries (ADL 1999)*, Washington, DC, USA. IEEE Computer Society, Los Alamitos (1999)
6. Chan, Y., Chang, C.: Spatial similarity retrieval in video databases. *Journal of Visual Communication and Image Representation* 12, 107–122 (2001)
7. Li, J., Dong, G., Ramamohanarao, K.: Making use of the most expressive jumping emerging patterns for classification. *Knowledge and Information Systems* 3(2), 1–29 (2001)
8. Dong, G., Li, J.: Mining border descriptions of emerging patterns from dataset pairs. *Knowledge and Information Systems* 8(2), 178–202 (2005)
9. Kobyliński, E., Walczak, K.: Efficient mining of jumping emerging patterns with occurrence counts for classification. In: Chan, C.-C., Grzymala-Busse, J.W., Ziarko, W.P. (eds.) *RSCTC 2008. LNCS (LNAI)*, vol. 5306, pp. 419–428. Springer, Heidelberg (2008)
10. Lee, A.J.T., Hong, R.W., Ko, W.M., Tsao, W.K., Lin, H.H.: Mining spatial association rules in image databases. *Information Sciences* 177(7), 1593–1608 (2007)

# Automatic Methods for Determining the Characteristic Points in Face Image

Mariusz Kubanek

Czestochowa University of Technology,  
Institute of Computer and Information Science,  
Center for Intelligent Multimedia Techniques,  
Dabrowskiego Street 73, 42-200 Czestochowa, Poland  
mariusz.kubanek@icis.pcz.pl .

**Abstract.** There are described in this paper algorithms of extract the characteristic points in a face image with complex background. Presented algorithms combine known tasks of image processing and developed procedures. Novel peculiarity of the method, in comparison to the methods described in the existing literature, is using its own algorithms for edge detection of iris and eyelid edges. The methods for automatic face location, eyes, eye iris, corners of eyes, edges of the eyelids, corners of lip and external edges of lip location - were described and developed. The part of the work dealing with face recognition was based on the technique of automatic authentication of a person with assumption of the use of automatically extracted, structural characteristics of the leads for the biometrical authentication systems' improvement. Achieved results are satisfactory for purpose of use in developed face recognition methods.

**Keywords:** face localisation, face recognition, iris detection, edges of the eyelids detection, lip corner detection, edges of lip detection.

## 1 Introduction

Nowadays face and characteristic points in face image detection methods have wide scope of use in many computer vision applications, such as systems of human-computer interaction [1,2]. Significant part of applications is used in systems for face recognition in access control and model-based video coding [2,3].

Automatically location of the face and facial characteristic points allows the building of biometric systems that may successfully carry out the identification, verification of people and for example lip reading.

The first step is to locate face area on image. This is done by algorithm combining skin-like color detection method [4], median filtering and simple method of determining region boundaries. Once the face has been detected the next step is to locate both eyes' area. For purpose of detecting eyes the 3-stage gray-level image projection algorithm was applied. The iris detection has been accomplished - through combining gray-level image thresholding (with automated threshold selection), median filtering, *Canny* edge detection and procedure of

finding circles in eye's edge-image. Similarly, the edge of the eyelid detection is implemented. Detection of outer edges and corners of mouth based on color image is realized.

## 2 Face Detection and Eyes Localisation

Frontal face image was assumed as an input of face recognition procedure to have an *RGB* color space. The algorithm of thresholding in *I2* color space was applied for enabling face detecting in an input image [4,6]. The *I2* color space enables to detect skin-like color regions in input *RGB* image with satisfactory result.

To transform image to *I2* color space, we have to subtract *R* and *B* components of *RGB* space. Both components are dependent on *G* component. Dividing the components *R* and *B* by the component *G* significantly reduces light changes' impact on the method correctness. Similarly the reduction of lighting effects to the method's quality, gives *HSV* color space application. That has been also implemented in our system. Let  $I_R$ ,  $I_G$  and  $I_B$  represent matrix containing *R*, *G* and *B* components respectively, for each pixel of input image. The output  $I_{RB}$  matrix we can obtain after subtraction:

$$I_{RB}[i][j] = \sum_{i=0}^w \sum_{j=0}^h \frac{I_R[i][j] - I_B[i][j]}{I_G[i][j]} \quad (1)$$

where:  $w, h$  - width and height of input image.

Once we have an *I2* color space image the next step is to apply thresholding procedure which may be described by following formula:

$$I'[x][y] = \begin{cases} 255 & \text{for } I_{RB}[x][y] > T \\ 0 & \text{for } I_{RB}[x][y] \leq T \end{cases} \quad (2)$$

where:  $x = 0 \dots w$ ,  $y = 0 \dots h$  - coordinates of the pixel,  $w, h$  - width and height of the image,  $I_{RB}$  - input *I2* color space matrix,  $I'$  - output, thresholding image and  $T$  - threshold.

The modified median filtering procedure was applied - in order to eliminate noise around the face area and small objects in background. In classic median filtering values of pixels neighboring analyzed pixel (including analyzed pixel) are sorted from lowest to highest (or reverse). The median value of surroundings pixels is being assigned and to analyzed pixel [6]. The result of modified median filtering is image with smoothed edges of face area. Also small background objects were eliminated.

The boundaries of face region are acquired in simple procedure. The upper boundary is the row of image where number of white pixels (occurring continuously) exceeds given value. The lower boundary - the line where number of white pixels decreases below given value. Vertical boundaries are white pixels found most on left and right side of the image (in limits of horizontal boundaries).

It has been possible to search the pupils by looking for two dark regions lying within a certain area of the face. There are applied Gradient Method and Integral Projection (GMIP) [4] to find horizontal (*h line*) and vertical (*v line 1*, *v line 2*) line of eyes. Before that, the image face is converted to the grayscale. Following equations (3,4,5) describe way of finding the pupils.

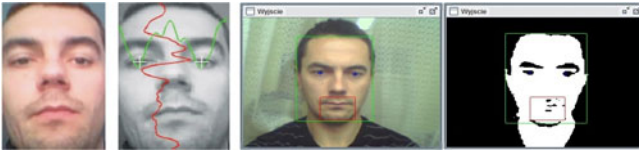
$$h \text{ line} = \max_i \left( \sum_{j=1}^{w-1} |im_{i,j} - im_{i,j+1}|, i = 1..h \right) \quad (3)$$

$$v \text{ line } 1 = \max_j \left( \sum_{i=h \text{ line}-t}^{2t} |im_{i,j} - im_{i+1,j}|, j = 1.. \frac{w}{2} \right) \quad (4)$$

$$v \text{ line } 2 = \max_j \left( \sum_{i=h \text{ line}-t}^{2t} |im_{i,j} - im_{i+1,j}|, j = \frac{w}{2} + 1..w \right) \quad (5)$$

where:  $w$  - width of image  $im$  in pixels,  $h$  - height of image  $im$  in pixels,  $t$  - half of the section containing the eye area.

To search the lips initially, the approximate positions of the corners of mouth are predicted, using the positions of the eyes, the model of face and the assumption, therefore one should to have a near frontal view. Fig. 1 shows example of eyes localisation and area of mouth detection on basic of eyes position



**Fig. 1.** Example of eyes localisation and area of mouth detection on basic of eyes position

### 3 The Mechanism of Iris Detection

We have to find circle circumscribed about the iris on area containing image of the eye (see Fig. 2). This task may be completed in following steps.

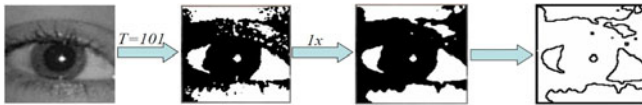


**Fig. 2.** Determined area containing image of the eye

**First step:** Thresholding the gray-level image with automatically selected threshold and applying Canny edge detection procedure. In our procedure value of threshold is set as average value of pixels in analyzed image incremented by constant (101, this constant was determined experimentally). The threshold value is calculated using the relationship:

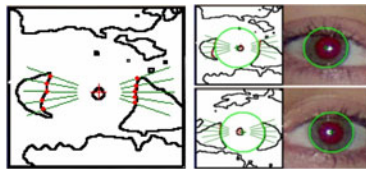
$$T = 1 + \frac{1}{w \cdot h} \sum_{x=1}^w \sum_{y=1}^h I[x][y] \tag{6}$$

After applying threshold, the median filtering procedure was used (see section 2). Next, there was applied the Canny edge detection procedure. Stage of Gaussian filtering was omitted due to reducing edge sharpness. Image of the eye after edge detection is shown on Fig. 3



**Fig. 3.** Processed eye image before and after Canny edge detection procedure

**Second step:** Finding points on edge of the iris and determining best-fitted circle. We use 10 rays with center in point of maximum projection in order to detect edge of the iris (see section 2). The rays are inclined at angles of: 340°, 350°, 0°, 10°, 20°, 160°, 170°, 180°, 190°, 200°. In range of 20 to 65 pixels from the center, along each ray we find first occurrence of black pixel and set it as edge of the iris. This gives us set of 10 points - candidates for circle circumscribed about the iris. Result of above procedure is presented in Fig. 4. Next step is to verify which points belong to iris circle.



**Fig. 4.** Result of finding point on edge of the iris

## 4 The Mechanism of Edges of the Eyelids Detection

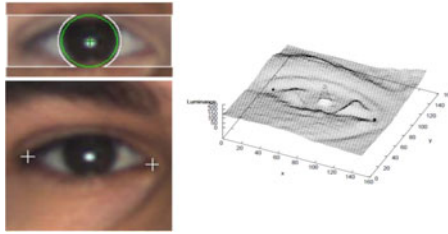
After determining the position and shape of the iris, the next step is to extract characteristics eyes - to determine the shape of the upper and lower eyelids. It is assumed that the shape edges of both eyes will be approximated third-degree polynomials.

In order to extract the edge of the eyelid, method based on processing information about the luminance image (LM - Luminance Minimum and ILP - Interval Luminance Projection) and based on an analysis of the space (R/G, B/G) was proposed. Information about the luminance was collected from the image represented in space RGB according to the rule:

$$Y = 0,299 \cdot R + 0,587 \cdot G + 0,114 \cdot B \quad (7)$$

With such representation of the eye area, we can easily distinguish the edge of the eyelid, searching the smallest value of luminance in the vicinity of the iris.

To seek internal and external corner of the eye separate areas were designated. When setting the initial eye corners, there were adopted points contained in the extreme positions to the left and the right in the search areas [7]. Fig. 5 shows separate areas, luminance levels for the eye and pre-set eye corners.



**Fig. 5.** Separate areas, luminance levels for the eye and pre-set eye corners

In order to determine points located on the edge of the upper eyelid, in the matrix luminance there were searched values of local minima starting from the point of the outer corner of the eye. Because of the unreliability of the method for images with large differences in values luminance at the edge of the eye, RGBG procedure based on image processing in space (R/G, B/G) was proposed. Analysis of the components of such a space allows accurate separation of areas with a specific color, depending on the adopted threshold. Using the method of least squares, based on the found items, set the shape of the polynomials approximated edge of eyelids. After the appointment of polynomials for the upper and lower eyelids, as the final corners of the eye, the nearest of the pupil intersections of polynomials was adopted [7]. Fig. 6 shows threshold in the space (R/G, B/G) and the designated edges of eyelids.

## 5 The Mechanism of Mouth Detection

To search the lips initially, the approximate positions of the lip's corners were predicted, using the position of eyes, the face's model and the assumption, therefore one should to have a near frontal view. Exact finding of corners of mouth was very important because the edge of mouth would be defined on basis of corners of mouth. In this work, there was assumed that one should has appointed only external edges of mouth.



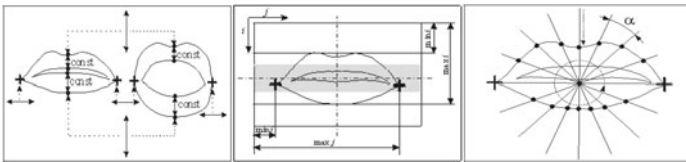
**Fig. 6.** Threshold in the space (R/G, B/G) and the designated edges of eyelids

The position of mouth corners may be found through the application of the own method of finding mouth corners. The mouth corners are defined on basic of specific mouth color and specific mouth shape (Color and Shape of Mouth, CSM) [5]. First, the mouth position is found using specific color, and then mouth corners are defined as extreme points of situated mouth. To qualification of specific mouth color, video frame is spreaded on component RGB. Then, for mouth color  $MA$  it's accepted area, in which:

$$MA = \begin{cases} \frac{R}{G} - \frac{B}{G} < T1 \\ \text{and} \\ \frac{R}{B} - \frac{G}{B} < T2 \end{cases} \quad (8)$$

One should  $T1$  and  $T2$  to accept experimental.

Having definite mouth area and mouth corners, it is possible to mark external mouth edges (Lip Contour Detection, LCD) [5]. On basis of mouth corners resource of circle is defined, for which what  $\alpha$  a ray is drawn, beginning from one from appointed corners of mouth. It's got  $2\pi/\alpha$  of rays. Moving oneself along every from rays in direction of resource of circle, points are marked where appointed earlier area of mouth  $MA$  begins [5]. Fig. 7 shows scheme of assumed visual motion of mouth, location of corners of mouth and definition of external edges of mouth.



**Fig. 7.** Scheme of assumed visual motion of mouth, location of corners of mouth and definition of external edges of mouth

In order to eliminate significant mistakes, each of appointed points is being compared with neighboring points, and suitably modified. In dependence from settle angle of jump  $\alpha$  there are received  $2\pi/\alpha$  points [5]. One should  $\alpha$  to accept experimental. It was accepted  $\alpha = 2/16$ . Fig. 8 shows examples of appointed corners and external edges of mouth.





**Fig. 8.** Examples of appointed corners and external edges of mouth

## 6 Experimental Results

In first experiment, effectiveness of the method of location of corners of the eyes and effectiveness of the method of location of corners of the eyes was tested. It was tested twenty different persons, recorded by the camera. Every recording contained about 100 frames/s, so it was tested near 2000 video frames. For every frame box, corners of eyes and corners of mouth were marked by hand, and next this corners were compared with automatically situated corners of eyes and corners of mouth respectively, received by using the automatic location of corners of the eyes and corners of mouth. Tab. 1 shows results of first experiment.

**Table 1.** Average vertical and horizontal incorrect location of corners of the eyes and corners of mouth in pixel

| method                       | amount of frames | error x [pixels] | error y [pixels] |
|------------------------------|------------------|------------------|------------------|
| corners of the eyes location | 2000             | 5,3              | 7,1              |
| corners of mouth location    | 2000             | 4,8              | 5,7              |

In second experiment the effectiveness of edge detection of the eyelids and the lips were tested. There were tested also twenty different persons, recorded by the camera. Each recorded frame box was checked, defining membership of recognized object to object correct object, incorrect object (+) and incorrect object (-). Incorrect object (+) marks, that corners became recognized correctly and incorrect object (-) marks, that corners became recognized irregularly. Tab. 2 shows results of eyelids and lip tracking in real-time.

**Table 2.** Result of eyelids and lip tracking in real-time

| method                    | amount of frames | correct object [%] | incorrect object (+) [%] | incorrect object (-) [%] |
|---------------------------|------------------|--------------------|--------------------------|--------------------------|
| edges of eyelids location | 2000             | 87,7               | 3,4                      | 8,9                      |
| edges of lips location    | 2000             | 92,2               | 2,2                      | 5,6                      |

## 7 Conclusion and Future Work

There were presented in this paper methods for automatic face location, eyes, eye iris, corners of eyes, edges of the eyelids, corners of lip and external edges of lip location. The main aim was to show how to locate specific points on the human face automatically. Presented methods give satisfactory results in aim of future processing in person verification system based on facial asymmetry and tracking of characteristic points on human face. A major defect of the methods is the manual selection of threshold values. The methods would be improved by realization of automated threshold selection.

Described methods have already been partially implemented in our system. In future, we plan to build the system for identification and verification identity of users, based on the automatic location of a human face and its characteristic features. Main aim would be to implement elements of the face location based on gray-scale images and immunization of the system to a variable value of lighting. Destiny of the created system would be working in close-circuit television systems.

## References

1. Fan, L., Sung, K.K.: Model-based Varying Pose Face Detection and Facial Feature Registration in Colour Image. In: PRL 2003, January 2003, vol. 24(1-3), pp. 237–249 (2003)
2. Rydzek, S.: Extract Iris Shape Determination in Face Image with Complex Background. Computing, Multimedia and Intelligent Techniques, Special Issue on Live Biometrics and Security 1(1), 191–200 (2005)
3. Eisert, P., Wiegand, T., Girod, B.: Model-Aided Coding: A New Approach to Incorporate Facial Animation into Motion-Compensated Video Coding. *CirSysVideo* 2000 10(3), 344–358 (2000)
4. Kukharev, G., Kuzminski, A.: Biometric Technology. Part. 1: Methods for Face Recognition, Szczecin University of Technology, Faculty of Computer Science (2003) (in Polish)
5. Kubanek, M.: Method of Speech Recognition and Speaker Identification with use Audio-Visual Polish Speech and Hidden Markov Models. In: Saeed, K., Pejas, J., Mosdorof, R. (eds.) *Biometrics, Computer Security Systems and Artificial Intelligence Applications*, pp. 45–55. Springer Science + Business Media, New York (2006)
6. Rydzek, S.: Iris Shape Evaluation in Face Image with Complex Background. *Biometrics*. In: Saeed, K., Pejas, J., Mosdorof, R. (eds.) *Computer Security Systems and Artificial Intelligence Applications*, pp. 79–87. Springer Science + Business Media, New York (2006)
7. Rydzek, S.: Automatic authentication method based on measurement of the characteristics of asymmetry of the eyes and/or mouth, Dissertation, Czestochowa (2007) (in Polish)

# Effectiveness Comparison of Three Types of Signatures on the Example of the Initial Selection of Aerial Images

Zbigniew Mikrut\*

AGH University of Science and Technology, Institute of Automatics  
al. Mickiewicza 30, 30-059 Krakow, Poland  
zibi@agh.edu.pl

**Abstract.** The paper describes, implements and compares three types of pulsed neural networks (ICM and two PCNNs). These networks generated more than 900 image signatures from aerial photos. The signatures have been divided into two classes: suitable and unsuitable for the next stages of photogrammetric analysis. *Backpropagation* neural networks with various sizes of the hidden layer have been used for the classification of signatures. The effectiveness of the three types of image signatures has been determined based on the recognition results.

**Keywords:** aerial images, photogrammetry, PCNN, ICM, image signatures.

## 1 Introduction

A comparison of effectiveness of three image representations have been executed, using the example of preliminary selection of aerial sub-images. This exemplary problem boils down to verification whether an extracted sub-image is "informative" enough and should be qualified as suitable for further processing. Further processing in this context is understood as marking some characteristic points, belonging to selected sub-images in the first image, and identifying their counterparts in the second image [8].

In the paper [10] an attempt was made to automate the initial selection stage: the authors chose a relatively new and not widely known *image signature* method of creating sub-image representations. The signatures are generated by a pulsed neural network. The operating principles of such networks have been presented in [4] [6] [9] [10] and will be shortly recalled in section 2. A *backpropagation* neural network has been used as the classifier of the created representations, as it is known for its effectiveness and widely used [12]. The combination of the neural network and representation generation in the form of signatures resulted in reasonably good results. The test set has been recognized in more than 73% of cases, and the application of the recognition reliability evaluation method has increased the recognition results to more than 80% (with 20% rejection rate) [10].

---

\* This work has been partially supported by the AGH UST grant No 10.10.120.783.

Good results have stimulated the author to inspect thoroughly different versions of pulsed networks used for signature generation. It has been decided to compare three types of signature generating networks with respect to recognition results obtained for the aforementioned procedure of aerial photos sub-image selection.

## 2 Types of Signature Generating Networks

The model of Pulse Coupled Neural Network (PCNN) has been introduced as a result of a study of the cat’s visual system, carried out under supervision of Eckhorn [3]. The network proposed by Eckhorn consists of a layer of neurons, interconnected by two types of feedback: *feeding* (F) and *linking* (L), The connection layout is schematically presented in Fig. 1. The size of the network is equal to the size of the processed digital image. The combination of the image’s pixel value S with the F and L signals produces the cumulative signal U (internal activity), which is compared to the neuron threshold level  $\Theta$ . Once the threshold is exceeded, a pulse is generated (the output of Y neuron takes the value of 1). At this point, the threshold level is increased. It then gradually decreases to the steady state level, allowing the production of subsequent pulses. The  $W^*Y$  and L signals reflect the activity of neighboring neurons by means of properly defined matrices of coefficients. Similarly to the threshold, the values of F and L signals also decrease with time.

The equations describing the action of a single pulsed neuron take the form:

$$F_{ij} [n + 1] = fF_{ij} [n] + S_{ij} + V_F \sum_{kl} W_{ijkl} Y_{kl} [n] \tag{1}$$

$$L_{ij} [n + 1] = lL_{ij} [n] + V_F \sum_{kl} M_{ijkl} Y_{kl} [n] \tag{2}$$

$$U_{ij} [n + 1] = F_{ij} [n + 1] (1 + \beta L_{ij} [n + 1]) \tag{3}$$

$$\Theta_{ij} [n + 1] = g\Theta_{ij} [n] + V_{\Theta} Y_{ij} [n] \tag{4}$$

$$Y_{ij} [n + 1] = \begin{cases} 1 & \text{when } U_{ij} [n + 1] > \Theta_{ij} [n] \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where:

$U_{ij}$  - is an internal state of a neuron at coordinates  $ij$

$S_{ij}$  - is the activation (input image pixel, rescaled to the [0,1] range)

$F_{ij}, L_{ij}$  - are the components accounting for the feedbacks (*feeding* and *linking*)

$\Theta_{ij}$  - is the neuron threshold

$Y_{ij}$  - is an external state of a neuron (1 - pulse, 0 - no pulse)

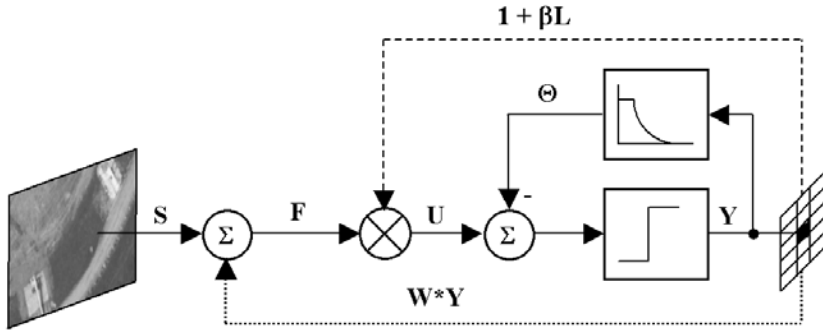
$W_{ijkl}$  - are the neighborhood coefficients of the neuron  $ij$  at coordinates  $kl$  for F

$M_{ijkl}$  - are the neighborhood coefficients of the neuron  $ij$  at coordinates  $kl$  for L

$V_F, V_L$  - are the amplification coefficients for direct F and L feedbacks

$V_{\Theta}$  - determines the threshold step at the pulse generation

$n$  - is the iteration number



**Fig. 1.** Three types of block schemes for the neuron model, forming the PCNN (ICM) network

The  $f$ ,  $g$  and  $l$  coefficients determine the time constants for signal changes of  $\mathbf{F}$ ,  $\Theta$  and  $\mathbf{L}$  ( $f, g < 1, f > g$ ). The  $\beta$  coefficient defines the influence of  $\mathbf{L}$  connection on the internal state of neuron  $\mathbf{U}$ . The network's output is a sequence of binary patterns (images)  $\mathbf{Y}[n]$ , of the size equal to the input image  $\mathbf{S}$ . The  $\mathbf{W}$  and  $\mathbf{M}$  matrices are usually the same.

The idea of image (object) signatures has been introduced by Johnson in 1994 [9]. The basis for the signature calculation is the time function  $G[n]$ , obtained as summation result of  $\mathbf{Y}$  neuron outputs ("white pixels") in each step of the calculation:

$$G[n] = \sum_{ij} Y_{ij}[n] \quad (6)$$

Johnson has found that for analyses of simple objects located on a uniform background, the  $G[n]$  function becomes periodic after a certain number of iterations. The periodically repeated sequence has been dubbed as "signature". It has been shown, that for simple images the signature shape is independent of the rotation, translation, zooming or even the change of object viewing angle [4] [6] [9]. A real-life image (an aerial photo in particular) usually does not contain simple objects against a black background. Therefore the signature has been redefined as the function course in the first several tens of simulation steps (in most cases 25 or 50 steps are included [1] [4]). In other papers [5] the authors point out the fact that the network generates some features, manifesting as local maxima of the  $G$  function. In their opinion, the classification effectiveness depends on the number of features taken into account and thus can be calculated.

Three types of pulsed networks have been selected, in order to expose the differences in their architecture. It proved problematic to collect complete network descriptions, including the full set of coefficients described by formulas (1) - (5), size and definition of the neighborhood (the  $\mathbf{W}$  matrix) and the initial conditions. The following network types have been selected for the study:

- ICM (Intersecting Cortical Model) network - which is a simplified version of the PCNN network, defined by Kinser [7]. The simplification consists in

elimination of the L (linking) feedback loop from the neuron model, marked with the dashed line in Fig. 1. The network is denoted as: Atm,

- the so called Basic PCNN Model network, described by Johnson in [6]: such network would correspond to the block scheme in Fig. 1, after eliminating the feedback connection (dotted line). Denoted as Joh,
- one of the varieties of the complete PCNN network, studied by Nazmy [11]. The neurons forming such network can be represented by the full block scheme in Fig. 1. Denoted as Naz2\*.

Table 1 lists the parameters defining the three network types, that have been extracted from [7][6][11].

**Table 1.** A list of coefficients for the three types of pulsed networks

| Coefficient description and label      | Atm   | Joh | Naz2*  | Comments            |
|----------------------------------------|-------|-----|--------|---------------------|
| <i>linking</i> coefficient $\beta$     | 0     | 0.2 | 0.2    | eq. (3)             |
| damping for <i>linking</i>             | 1     | 0   | 0.3679 | 0.3679 eq. (2)      |
| damping for <i>feeding</i>             | f     | 0.9 | 0      | 0.9048 eq. (1)      |
| damping for threshold                  | g     | 0.8 | 0.8187 | 0.6065 eq. (4)      |
| threshold step $V_{\Theta}$            | 20    | 20  | 10     | eq. (4)             |
| weight for <i>feeding</i>              | $V_F$ | 1   | 0      | 0.01 eq. (1)        |
| weight for <i>linking</i>              | $V_L$ | 0   | 0.2    | 0.3* eq. (2)        |
| neighborhood radius r                  | 1     | 3   | 3      | for <b>W</b> matrix |
| central value of <b>W</b> $w(r+1,r+1)$ | 0     | 1   | 0      |                     |
| initial threshold $\Theta(0)$          | 0     | 1   | 0      | eqs. (4),(5)        |
| <b>W</b> normalization                 | yes   | yes | no     |                     |

\*star denotes a parameter modification, the original value was 1.

The **W** matrix has been created after calculating the inverse of individual element distances from the central point (r+1). Another widely accepted method consists of assigning the values of a two dimensional Gauss function [11].

Figure 2 presents examples of signatures generated by the three aforementioned network types. The input image was a sub-image of an aerial photo, later used for creation of a learning set for the neural networks.

In the initial simulation step the Atm and Naz2\* networks generate maximal signal level, as each neuron pulses when the threshold exceeding condition is fulfilled - see (5) and the  $\Theta(0)$  values in Table 1. For these two networks "nothing happens" in the next step, because the  $\Theta$  threshold has reached values in excess of U. The threshold value gradually decreases in consecutive steps - see (4) and Fig. 1 - and the neurons with high pixel values and that receiving high cumulative feedback signal start to pulse. A sequence of binary images is created, which - after applying equation (6) - forms the signature. As the plot shapes shown in Fig. 2 indicate, signatures are essentially different for the three networks implemented.

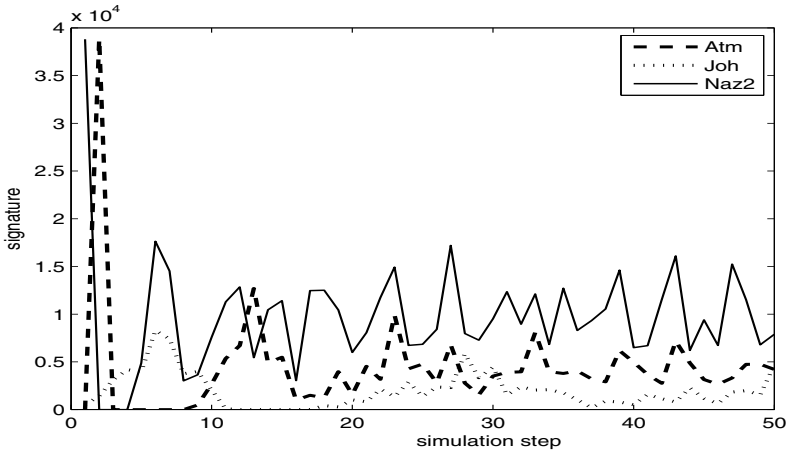


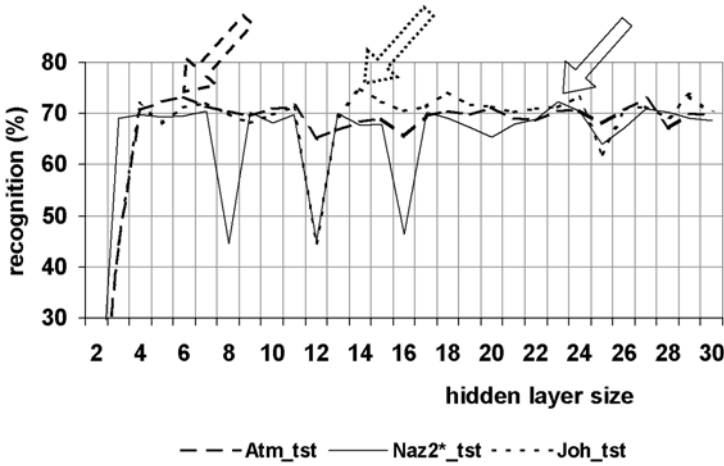
Fig. 2. Sample signatures generated by the three types of pulsed networks

### 3 Experiments with Neural Networks

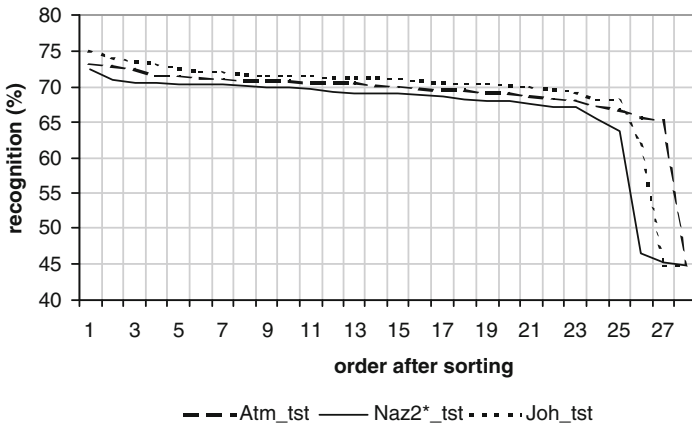
In order to compare the three types of generated signatures - for the same parts of aerial photo images - it has been decided to calculate three sets of signatures and then to compare the recognition results for these representations obtained by the neural networks. Two aerial photo images of the Krakow region have been used to construct the learning and test sets for the neural networks. Both feature city and rural settlement areas, as well as arable lands and forests. 452 areas have been selected from the first photo, 261x261 pixels each, which have been divided into two classes: areas that are suitable for future use in the photogrammetric analysis (231 images) and the unsuitable ones (221 images). Similarly, the second photo has been processed and used to create a test set: 452 images have been selected, from which 202 images have been described as promising and 250 as unsuitable from the photogrammetric point of view. Thus, both image classes included comparable numbers of elements. Five experts classified each of those sub-images independently as belonging to one of two above mentioned groups. The median value of classification determined that a given sub-image belonged to a specific class.

For all the images 50 element signatures have been generated, using the three types of networks. The signatures have been normalized to the  $[0,1]$  range. The study has been carried out as described in [10]. For all the experiments the Neural Network Toolbox [2] has been used, being a part of the Matlab environment. During consecutive experiments the size of the hidden layer has been changed from 3 to 30 elements. The networks have been trained using the *batch* method and the Levenberg-Marquardt algorithm. The training results proved to be similar - in most cases high recognition results, comparable to 100%, have been obtained. The testing results are presented in Fig. 3.

In such wide area comparative studies, the network architectures with the highest recognition results are usually selected. In this study the best results



**Fig. 3.** Testing results obtained from *backpropagation* networks with variable number of elements in the hidden layer, using the three types of signatures



**Fig. 4.** Sorted recognition results obtained by *backpropagation* networks for three types of signatures

were 73%, 72% and 75%, reached by Atm, Naz2\* and Joh signatures respectively. The selected architectures are used for further studies, after changing the weight coefficients initialization and the sequence of the learning set presentation. Studies are then repeated with various settings of the pseudo-random generator [12]. For the study presented here there was no need to apply such procedure. The results achieved by networks with various numbers of elements in the hidden layer have been sorted by the test set recognition results, as presented in Fig. 4. The relative location of the three plots indicates, that the three representations lead to different, though comparable, results. If one takes the best 25 (of 27) recognition rates into account, the three presented curves do not



overlap. This justifies a conclusion that - having the small differences in mind - the effectiveness of the tested networks and thus the quality of signatures can be determined. The best representation has been generated by the Joh network, followed by the Atm network and by the Naz2\*.

## 4 Summary

The paper investigated whether the structure and parameters of a pulsed network may affect the shape and quality of generated signatures. Three types of pulsed networks have been implemented. A sample application has been selected to test the signature quality, consisting in the division of aerial image fragments into two classes: "suitable" and "unsuitable", with respect to the subsequent procedure of marking distinctive points in one image and matching them to their equivalents on the other.

More than 900 aerial photo sub-images have been processed to obtain three forms of 50 element signatures. The data have been divided into a learning and a testing set. The former has been used for training the *backpropagation* neural networks with one hidden layer. The network's size varied between 3 and 30 neurons.

The results have been presented as plots of *sorted* recognitions results (see Fig. 4). The results turned out to be similar. Distinct, non-overlapping curves have been obtained for almost all of the ordered recognition rates, by means of sorting the signature recognition results obtained by *backpropagation* networks with different structures. This in turn allowed to create a ranking-list of the examined pulsed networks, with respect to the quality of generated signatures, in the context of photogrammetric sub-images selection.

## References

1. Atmer, J.: Image Signatures from PCNN using Computers. Diploma Work, Dept. of Physics, Royal Institute of Technology (KTH), Stockholm (2003)
2. Demuth, H., Beale, M., Hagan, M.: Neural Network Toolbox 5 Users Guide. The MathWorks, Inc., Natick (1992-2007)
3. Eckhorn, R., Reitboeck, H.J., Arndt, M., Dicke, P.: Feature Linking via Synchronisation among Distributed Assemblies: Simulations of Results from Cat Cortex. *Neural Computation* 2, 293-307 (1990)
4. Ekblad, U., Kinser, J.M., Atmer, J., Zetterlund, N.: The Intersecting Cortical Model in image processing. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 525(1-2), 392-396 (2004)
5. Forgáč, R., Mokriš, I.: Pulse Coupled Neural Network Models for Dimension Reduction of Classification Space. In: Proc. WIKT, Bratislava (2006)
6. Johnson, J.L., Padgett, M.L.: PCNN Models and Applications. *IEEE Trans. on Neural Networks* 10(3), 480-498 (1999)
7. Kinser, J.M.: A Simplified Pulse-Coupled Neural Network. In: Proc. SPIE, 2760(3) (1996)

8. Kurczyński, Z.: *Aerial and Satellite Imagery of the Earth*. Warsaw University of Technology Publishing House, Warsaw (2006) (in Polish)
9. Lindblad, T., Kinser, J.M.: *Image Processing Using Pulse-Coupled Neural Networks*. Springer, Heidelberg (2005)
10. Mikrut, S., Mikrut, Z.: *Neural Networks in the Automation of Photogrammetric Processes*. In: *Proc. XXI Congress ISPRS (International Society for Photogrammetry and Remote Sensing)*, Beijing, China, vol. XXXVII, pp. 331–336 (2008)
11. Nazmy, T.M.: *Evaluation of the PCNN standard model for image processing purposes*. *IJICIS* 4(2) (2004)
12. Tadeusiewicz, R.: *Neural Networks*. RM Academic Publishing House, Warsaw (1993) (in Polish)

# Combined Full-Reference Image Quality Metric Linearly Correlated with Subjective Assessment

Krzysztof Okarma

West Pomeranian University of Technology, Szczecin  
Faculty of Electrical Engineering,  
Chair of Signal Processing and Multimedia Engineering,  
26. Kwietnia 10, 71-126 Szczecin, Poland  
okarma@zut.edu.pl

**Abstract.** In the paper a new combined image quality metric is proposed, which is based on three methods previously described by various researchers. The main advantage of the presented approach is the strong linear correlation with the subjective scores without additional nonlinear mapping. The values and the obtained correlation coefficients of the proposed metric have been compared with some other state-of-art ones using two largest publicly available image databases including the subjective quality scores.

**Keywords:** Image quality assessment.

## 1 Review of Image Quality Assessment Techniques

Image quality assessment techniques play an important role in most computer vision systems, especially for the development of some new image processing (e.g. nonlinear filtering, compression or reconstruction) algorithms. Classical methods based on the Mean Squared Error and similar metrics [3,4] are poorly correlated with human perception but during recent years a rapid development of image quality assessment methods may be observed, started in 2002 by the proposal of the Universal Image Quality Index (UQI) by Wang and Bovik [24]. Further extensions of that metric into Structural Similarity (SSIM) [26] and Multi-Scale SSIM [27] can be also considered as milestones for the image quality assessment.

The most general classification of image quality assessment methods divides them into subjective and objective ones. The first group is based on the quality scores taken from a number of observers, leading to the Mean Opinion Scores (MOS) or Differential MOS (DMOS) as the differences between "reference" and "processed" Mean Opinion Scores. Such approach cannot be used directly by researchers for the optimisation of image processing algorithms because of necessary human interaction. Nevertheless, it can be helpful for the development of some new objective metrics, which can be computed as the vector values or single scalar metrics, which are the most desired for practical applications. Regardless of the simple interpretation, their computational complexity is usually much lower in comparison to the vector approach.

Further division of objective metrics is related to the required knowledge of the original image without any distortions. Most of the currently known methods, especially the universal ones, use the whole original image (full-reference methods) but there are also some reduced-reference [2,29] and "blind" (no-reference) ones [6]. Nevertheless, contemporary no-reference techniques are not universal but sensitive to usually only one or two types of distortions e.g. block effects on JPEG compression [1,10,25,28] or image blurring [8,15].

The correlation of results calculated by a quality metric with the scores obtained as subjective evaluations by human observers should be preferably linear. A serious increase of the correlation coefficient's value can also be obtained by the use of the nonlinear mapping based on the logistic function, as suggested by the Video Quality Experts Group (VQEG) [30]. One of the disadvantages of such approach is the fact that the proper choice of the function's coefficients require the usage of some additional optimisation procedures [18]. Besides, obtained coefficients strongly depend on the image set used in the experiments and the types of distortions present in the database.

The ideal image quality metric should be calculated using some nonlinear functions inside the computational procedure but the final results should be well correlated with the subjective assessment without additional mapping. In that sense the results presented in the literature with the correlation coefficients about 0.99 should be treated with caution, because they are usually presented as the effects of the additional nonlinear mapping. In that case, the obtained coefficients of the logistic function are usually different for each type of distortions present in a given database and the results obtained for the whole database are not necessarily so good.

For verification purposes numerous images contaminated by several types of distortions should be used. Currently, there are only two worth noticing, publicly available databases containing a relatively large number of images together with subjective scores. The first one is commonly used LIVE Image Quality Assessment Database delivered by Laboratory for Image and Video Engineering (LIVE) from the University of Texas at Austin [19]. It contains the DMOS values for 779 images with five types of distortions: JPEG2000 and JPEG compression, white Gaussian noise, Gaussian blur and JPEG2000 compressed images transmitted over simulated fast fading Rayleigh channel with bit errors typical for the wireless transmission. The database contains 24-bit colour images, but many researchers treat them as greyscale ones analysing the luminance channel only or converting them before the analysis. The analysis of the influence of the colour data on the image quality is a separate problem [13] and is not the topic of this paper.

Another, recently published, database is Tampere Image Database (TID2008) containing 25 reference images contaminated with 17 types of distortions of four levels each [16]. Each of 1700 distorted images has been assessed by 838 observers from three countries: Finland, Italy and Ukraine and the MOS values are included in the database. The types of distortions used in that database are: additive Gaussian noise, additive noise in colour components, spatially

correlated noise, masked noise, high frequency noise, impulse noise, quantization noise, Gaussian blur, image denoising, JPEG compression, JPEG2000 compression, JPEG transmission errors, JPEG2000 transmission errors, non eccentricity pattern noise, local block-wise distortions, mean intensity shift and contrast change.

Comparing to the LIVE database, over five times more observers participated in the experiments with slightly differing methodology of visual quality evaluation (pair-wise sorting instead of five level scale based evaluation). Due to much more contaminations included, TID covers more practical applications and features of human visual system (HVS) with much smaller number of abnormal experiments. The comparative analysis of many popular metrics' performance using the TID2008 database can be found in the Ponomarenko's paper [17].

## 2 Proposed Combined Metric and Its Verification

Taking into account the correlation coefficients between Mean Opinion Scores and the popular metrics, two of them are especially worth noticing. The first one is the Multi Scale Structural Similarity (MS-SSIM) and the second one is the Visual Information Fidelity (VIF) metric [20,21]. MS-SSIM is the extended version of the Structural Similarity which originates from the UQI metric. Single scale SSIM index for two images denoted as  $x$  and  $y$  is calculated as the mean value of the quality map obtained using the sliding window approach (similar to the convolution filtering) as in the following formula (in shortened form)

$$SSIM = \frac{(2\bar{x}\bar{y} + C_1) \cdot (2\sigma_{xy} + C_2)}{(\sigma_x^2 + \sigma_y^2 + C_1) \cdot [(\bar{x})^2 + (\bar{y})^2 + C_2]}, \tag{1}$$

where  $C_1$  and  $C_2$  are constants chosen in such a way that they do not introduce any significant changes in the results (e.g.  $C_1 = (0.01 \times 255)^2$  and  $C_2 = (0.03 \times 255)^2$ ) as suggested by the authors of the paper [26]. These coefficients prevent only from the possible division by zero in flat and dark areas in the image. In fact, the total result is the product of luminance distortions, contrast loss and the structural distortions (mean, variance and correlation comparison). Extension of that metric into the Multi Scale SSIM is based on the idea of operating over a dyadic pyramid. The luminance ( $l$ ), contrast ( $c$ ) and structure ( $s$ ) factors are weighted using the exponents for each scale so the final MS-SSIM definition is given as [27]

$$MS\text{-}SSIM(x, y) = [l_M(x, y)]^{\alpha_M} \cdot \prod_{j=1}^M [c_j(x, y)]^{\beta_j} [s_j(x, y)]^{\gamma_j}, \tag{2}$$

where  $M$  is the highest scale obtained after  $M - 1$  iterations of low-pass filtering and downsampling the filtered image by a factor of 2.

The VIF metric is based on the wavelet decomposition, although it also has the pixel domain version. It is defined in general as

$$VIF = \frac{\sum_{j=0}^S \sum_{i=1}^{M_j} I(c_{i,j}; f_{i,j})}{\sum_{j=0}^S \sum_{i=1}^{M_j} I(c_{i,j}; e_{i,j})}, \tag{3}$$

where  $S$  stands for the number of sub-bands (or scales in the pixel domain),  $M_j$  is the number of blocks at  $j$ -th sub-band or scale and  $I(x; y)$  denotes the mutual information between  $x$  and  $y$ . The denominator and numerator can be treated as the information that vision extracts from the reference image and from the distorted one assuming that  $c$  is a block vector at a specified location in the reference image,  $e$  is the perception of block  $c$  by a human viewer with additive noise  $n$ , and  $f$  is the perception of distorted block  $c$  [21].

Recently, the application of the Singular Value Decomposition (SVD) for image quality assessment purposes has also been investigated. Apart from earlier research related to the  $M_{SVD}$  metric [22][23] and the application of some other transform based metrics [5], a few novel methods based on the reflection factors [7] and R-SVD metric [9] have been proposed. These two metrics use similar assumptions and the R-SVD metric leads to better results, but unfortunately requires the additional nonlinear mapping. The calculation of R-SVD quality index can be performed as

$$R-SVD = \frac{\sqrt{\sum_{i=1}^m (d_i - 1)^2}}{\sqrt{\sum_{i=1}^m (d_i + 1)^2}}, \tag{4}$$

where  $d_i$  are the singular values of the reference matrix calculated as  $\hat{R} = \hat{U} \Lambda V^T$ . The matrices  $U$ ,  $S$  and  $V^T$  are calculated as the result of the SVD decomposition of the original image  $A$ , and  $\hat{U}$ ,  $\hat{S}$  and  $\hat{V}^T$  of the distorted one ( $\hat{A}$ ).  $\Lambda$  is the matrix with ones on the diagonal and zeros elsewhere. In fact, instead of the right singular matrix of the distorted image  $\hat{V}^T$ , the original matrix  $V^T$  is used and the left singular matrix  $\hat{U}$  is calculated as

$$\hat{U}_i = \begin{cases} 0 & \text{if } \hat{s}_i = 0 \\ \hat{A} \cdot V_i / \hat{s}_i & \text{otherwise} \end{cases}, \tag{5}$$

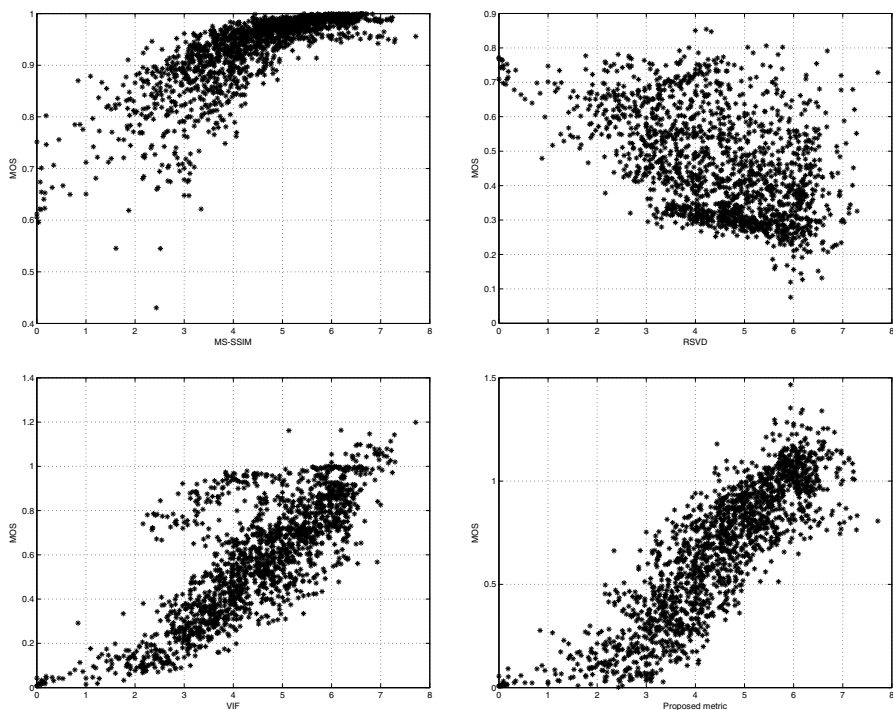
where  $\hat{s}_i$  denotes the singular values of the matrix representing the distorted image.

Taking into account the different properties of the three metrics presented above, the proposed combined quality metric can be defined as

$$CQM = (MS-SSIM)^a \cdot (VIF)^b \cdot (R-SVD)^c \tag{6}$$

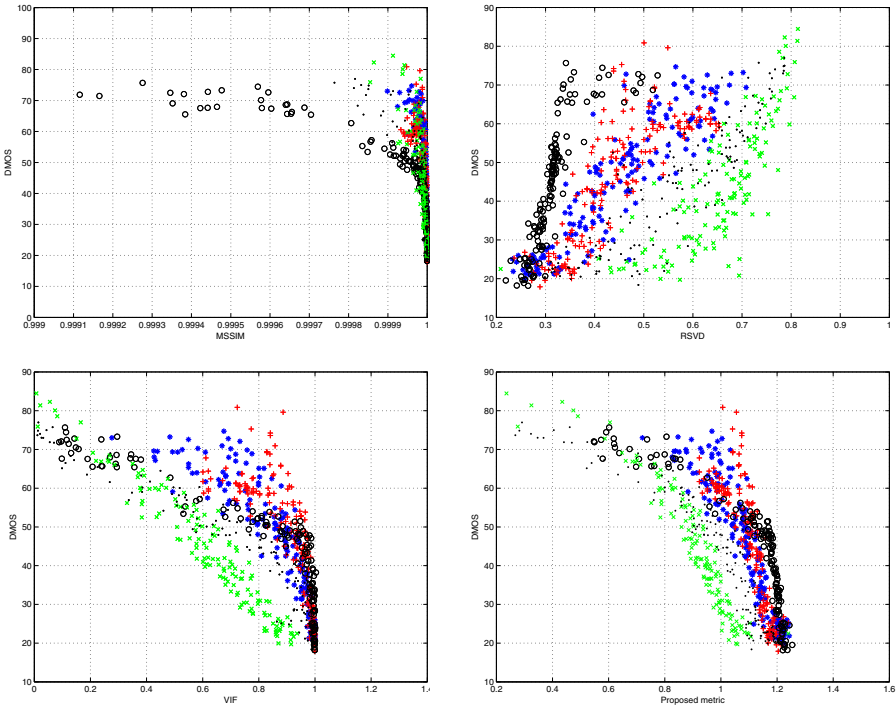
**Table 1.** Results of the linear correlation coefficients obtained during experiments for the TID2008 database

| Quality Metric | Correlation coefficient |          |         |
|----------------|-------------------------|----------|---------|
|                | Pearson                 | Spearman | Kendall |
| MS-SSIM        | 0.784                   | 0.853    | 0.654   |
| VIF            | 0.778                   | 0.750    | 0.586   |
| R-SVD          | 0.478                   | 0.470    | 0.325   |
| CQM (proposed) | 0.860                   | 0.872    | 0.677   |



**Fig. 1.** Scatter plots of the MOS values and three metrics being used for the construction of the proposed one, obtained for the TID2008 database (each point represents single distorted image)

Because of the fact that even the scores of 1700 images present in the Tampere Image Database do not fully correspond to the Human Visual System, there is a little sense in the "exact" optimisation of the exponents. Nevertheless, the simple combination of  $a = 7$ ,  $b = 0.3$  and  $c = -0.15$  is near optimal and leads to the Pearson correlation coefficient equal to 0.86 for the TID2008 database. The linear correlation results obtained using each of three metrics separately and the combined metric are presented in Figs. 1 and 2 as well as in the Table 1 (for the TID2008 database). As the result of the experiment for the LIVE database with realigned DMOS values for the same combination of the expo-



**Fig. 2.** Scatter plots of the DMOS values and three metrics being used for the construction of the proposed one, obtained for the LIVE database (each point represents a single distorted image corrupted by one of five types of distortions)

When comparing the proposed metric with DMOS values for the LIVE database, the Pearson linear correlation coefficient for the proposed metric is equal to 0.7214, while for the other metrics the following values have been obtained: 0.4762 for MS-SSIM, 0.7327 for VIF and 0.4999 for R-SVD. Such a result is slightly worse than for the VIF metric but it should be remembered that only five types of distortions are present in the LIVE database and the Visual Information Fidelity has been developed with the use of that database.

### 3 Conclusions

Analysing the results presented in the Table 1 and both figures, it can be easily observed that proposed combined metric has a great advantage of strong linear correlation with the subjective quality evaluation. It can be an interesting stimulus for further research related to the combined metric which could be more suitable for the colour image quality assessment [14]. Proposed metric has been tested using two largest public domain colour image databases but all the images have been converted to greyscale before the calculations, what is typical also in most publications by other researchers related to the image quality assessment.



Another possible direction of further experiments is the use of the saliency maps and the extension of the statistical approach [11][12] in order to increase the processing speed.

## References

1. Bovik, A., Liu, S.: DCT-domain Blind Measurement of Blocking Artifacts in DCT-Coded Images. In: Proc. Int. Conf. Acoustics, Speech and Signal Processing, Salt Lake City, USA, pp. 1725–1728 (2001)
2. Carnec, M., Le Callet, P., Barba, P.: An Image Quality Assessment Method Based on Perception of Structural Information. In: Proc. Int. Conf. Image Processing, Barcelona, Spain, vol. 2, pp. 185–188 (2003)
3. Eskicioglu, A., Fisher, P., Chen, S.: Image Quality Measures and Their Performance. IEEE Trans. Comm. 43(12), 2959–2965 (1995)
4. Eskicioglu, A.: Quality Measurement for Monochrome Compressed Images in the Past 25 Years. In: Proc. IEEE Int. Conf. Acoust. Speech Signal Process, Istanbul, Turkey, pp. 1907–1910 (2000)
5. Girshtel, E., Slobodyan, V., Weissman, J., Eskicioglu, A.: Comparison of Three Full-Reference Color Image Quality Measures. In: Proc. SPIE of 18th IS&T/SPIE Annual Symposium on Electronic Imaging, Image Quality and System Performance, San Jose, CA, vol. 6059 (2006), doi:10.1117/12.644226
6. Li, X.: Blind Image Quality Assessment. In: Proc. IEEE Int. Conf. Image Proc., pp. 449–452 (2002)
7. Mahmoudi-Aznavah, A., Mansouri, A., Torkamani-Azar, F., Eslami, M.: Image Quality Measurement Besides Distortion Type Classifying. Optical Review 16(1), 30–34 (2009)
8. Marziliano, P., Dufaux, F., Winkler, S., Ebrahimi, T.: A No-Reference Perceptual Blur Metric. In: Proc. IEEE Int. Conf. Image Processing, Rochester, USA, pp. 57–60 (2002)
9. Mansouri, A., Mahmoudi-Aznavah, A., Torkamani-Azar, F., Jahanshahi, J.A.: Image Quality Assessment Using the Singular Value Decomposition Theorem. Optical Review 16(2), 49–53 (2009)
10. Meesters, L., Martens, J.-B.: A Single-Ended Blockiness Measure for JPEG-Coded Images. Signal Processing 82(3), 369–387 (2002)
11. Okarma, K., Lech, P.: Monte Carlo Based Algorithm for Fast Preliminary Video Analysis. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2008, Part I. LNCS, vol. 5101, pp. 790–799. Springer, Heidelberg (2008)
12. Okarma, K., Lech, P.: A Statistical Reduced-Reference Approach to Digital Image Quality Assessment. In: Bolc, L., Kulikowski, J.L., Wojciechowski, K. (eds.) ICCVG 2008. LNCS, vol. 5337, pp. 43–54. Springer, Heidelberg (2009)
13. Okarma, K.: Colour Image Quality Assessment using Structural Similarity Index and Singular Value Decomposition. In: Bolc, L., Kulikowski, J.L., Wojciechowski, K. (eds.) ICCVG 2008. LNCS, vol. 5337, pp. 55–65. Springer, Heidelberg (2009)
14. Okarma, K.: Two-Dimensional Windowing in the Structural Similarity Index for the Colour Image Quality Assessment. In: Jiang, X., Petkov, N. (eds.) CAIP 2009. LNCS, vol. 5702, pp. 501–508. Springer, Heidelberg (2009)
15. Ong, E.-P., Lin, L.W., Yang, Z., Yao, S., Pan, F., Jiang, L., Moschetti, F.: A No-Reference Quality Metric for Measuring Image Blur. In: Proc. 7th Int. Symp. Signal Processing and Its Applications, Paris, France, pp. 469–472 (2003)

16. Ponomarenko, N., Carli, M., Lukin, V., Egiazarian, K., Astola, J., Battisti, F.: Color Image Database for Evaluation of Image Quality Metrics. In: Proc. Int. Workshop on Multimedia Signal Processing, Cairns, Queensland, Australia, pp. 403–408 (2008)
17. Ponomarenko, N., Battisti, F., Egiazarian, K., Astola, J., Lukin, V.: Metrics Performance Comparison for Color Image Database. In: Proc. 4th Int. Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, Arizona, USA (2009)
18. Sendashonga, M., Labeau, F.: Low Complexity Image Quality Assessment Using Frequency Domain Transforms. In: Proc. IEEE Int. Conf. Image Processing, pp. 385–388 (2006)
19. Sheikh, H.R., Wang, Z., Cormack, L., Bovik, A.C.: LIVE Image Quality Assessment Database Release 2, <http://live.ece.utexas.edu/research/quality>
20. Sheikh, H.R., Bovik, A.C., de Veciana, G.: An Information Fidelity Criterion for Image Quality Assessment Using Natural Scene Statistics. IEEE Trans. Image Processing 14(12), 2117–2128 (2005)
21. Sheikh, H.R., Bovik, A.C.: Image Information and Visual Quality. IEEE Trans. Image Processing 15(2), 430–444 (2006)
22. Shnayderman, A., Gusev, A., Eskicioglu, A.: A Multidimensional Image Quality Measure Using Singular Value Decomposition. In: Proc. SPIE Image Quality and Syst. Perf., vol. 5294(1), pp. 82–92 (2003)
23. Shnayderman, A., Gusev, A., Eskicioglu, A.: An SVD-Based Gray-Scale Image Quality Measure for Local and Global Assessment. IEEE Trans. Image Processing 15(2), 422–429 (2006)
24. Wang, Z., Bovik, A.: A Universal Image Quality Index. IEEE Signal Processing Letters 9(3), 81–84 (2002)
25. Wang, Z., Bovik, A., Evans, B.: Blind Measurement of Blocking Artifacts in Images. In: Proc. IEEE Int. Conf. Image Processing, pp. 981–984 (2000)
26. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image Quality Assessment: From Error Measurement to Structural Similarity. IEEE Trans. Image Processing 13(4), 600–612 (2004)
27. Wang, Z., Simoncelli, E., Bovik, A.: Multi-Scale Structural Similarity for Image Quality Assessment. In: Proc. 37th IEEE Asilomar Conf. on Signals, Systems and Computers, Pacific Grove, CA (2003)
28. Wang, Z., Sheikh, H., Bovik, A.: No-Reference Perceptual Quality Assessment of JPEG Compressed Images. In: Proc. IEEE Int. Conf. Image Processing, Rochester, USA, pp. 477–480 (2002)
29. Wang, Z., Simoncelli, E.: Reduced-Reference Image Quality Assessment using a Wavelet-Domain Natural Image Statistic Model. In: Proceedings of SPIE, Proc. Human Vision and Electronic Imaging Conference, San Jose, USA, vol. 5666, pp. 149–159 (2005)
30. VQEG, Final Report on the Validation of Objective Models of Video Quality Assessment (August 2003), <http://www.vqeg.org>

# Evaluation of Pose Hypotheses by Image Feature Extraction for Vehicle Localization

Kristin Schönherr<sup>1</sup>, Björn Giesler<sup>1</sup>, and Alois Knoll<sup>2</sup>

<sup>1</sup> Audi Electronics Venture GmbH, 85080 Gaimersheim, Germany

<sup>2</sup> Institute of Computer Science VI, University of Technology Munich, 85748 Garching b. München, Germany

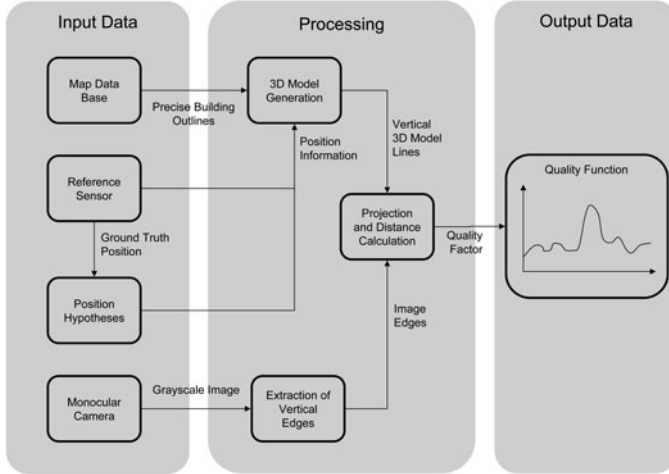
**Abstract.** For the realization of driving assistance and safety systems vehicles are being increasingly equipped with sensors. As these sensors contribute a lot to the cost of the whole package and at the same time consume some space, car manufacturers try to integrate applications that make use of already integrated sensors. This leads to the fact that each sensor has to fulfil several functions at once and to deliver information to different applications. When estimating very precise positioning information of a vehicle existing sensors have to be combined in an appropriate way to avoid the integration of additional sensors into the vehicle.

The GPS receiver, which is coupled with the navigation assistant of the vehicle, delivers a rough positioning information, which has to be improved using already available information from other built in sensors. The approach discussed in this paper uses a model-based method to compare building models obtained from maps with video image information. We will examine, if the explorative coupling of sensors can deliver an appropriate evaluation criteria for positioning hypotheses.

**Keywords:** Localization, probability density function, pose hypotheses, quality factor.

## 1 Introduction

Precise vehicle localization is turning into one of the most important challenges for driving assistance and security systems. Sensors are already available on the market, that provide exact vehicle localization, accurate enough to fulfil the demands. But the cost of these systems is still prohibitive for production vehicles. Additionally, integrating extra sensors into the vehicle is a major challenge to automotive design. The mentioned available high precise sensors do not fulfil these requirements at all. Therefore the preferred solution is to use information from already-integrated automotive sensors for multiple applications. The camera as sensor for image information delivery is already integrated in series-production vehicles for lane detection and parking assistance. It seems that the camera may be a suitable application-independent candidate for vehicle localization, see [7], [6]. In combination with additional sensor information like the rough position



**Fig. 1.** Overview of processing and data information for evaluation of position hypotheses

data of a GPS<sup>1</sup> receiver and precise maps (GIS<sup>2</sup>), we are doing research on how to obtain the exact vehicle position.

Creating appropriate evaluation criteria for the position hypotheses is a challenge, especially when utilizing combined sensor information. A probability density function to evaluate a position hypothesis upon the given sensor data, should reflect the precise vehicle position in form of a distinctive maximum. Model-based object pose estimation algorithms known from computer vision, such as RAPiD<sup>3</sup> [3], minimize distance information between projected model edges and detected object edges within a video image for estimating object positions. These algorithms work well in the laboratory; it is worthwhile though to evaluate, if they can work in automotive practice as well.

We build a 3D-building lattice model (compare [2], [4] and [5]) from precise outline information originating from highly accurate map material. This 3D lattice model is overlaid over the video image using a hypothesis for the car pose. It is assumed that every building has the same height and that building outlines are often occluded by spurious objects (objects that are not part of the model, and cannot contribute to the positioning process, such as parking or moving cars). Therefore the focus is set on vertical building edges, which are the most probable to be at least partially unoccluded. The model generation as well as the necessary image processing, has to be adapted to the generation of vertical model and object edges.

The sequence schema shown in Fig. 1 illustrates determination and evaluation the quality factor of different vehicle position hypotheses step by step.

<sup>1</sup> Global Positioning System.

<sup>2</sup> Geographic Information System.

<sup>3</sup> Real-time Attitude and Position Determination.

Different position hypotheses are generated based on precise ground truth position, determined by a highly accurate reference sensor system. The distance values between vertical model and object edges are used to generate a quality factor, though evaluating position hypotheses. The transferred quality function should show a maximum near the ground truth position.

## 2 Extraction of Vertical Lattice Lines

For evaluating a pose hypothesis, we use accurate map material to extract hypothetical building outlines. Thus it would be visible, where the car at the postulated hypothetical position is located.

At first the preparation of the precise map material, in which building outlines are stored as polygon lines, is explained. The point of view which is represented by the position information, decides whether a model line is visible or not, compare [11].

### 2.1 Backface Culling

In our map, building outline information is stored as polygons, whose edges are arranged in a clockwise orientation. For every line the normal vector is derived and compared with the viewing direction vector, to distinguish between visible and invisible edges. If the normal vector is directed along the viewing direction vector, the polygon line is backfaced and thus not visible. This line is not considered for further calculations. For this step there is no z-coordinate (the up vector in our coordinate system), so the backface culling [11] method is reduced to a 2-dimensional problem.

### 2.2 Ground Plane Based Angle Separation

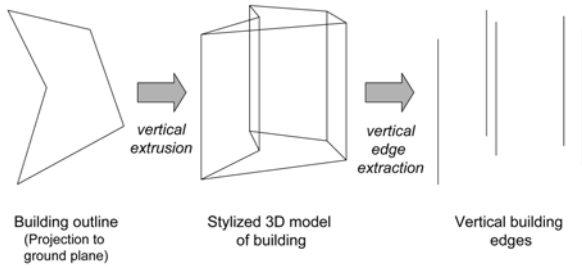
After the rough differentiation between visible and non visible borders, we take into account inter-object occlusions. From a certain viewing direction, some buildings can occlude others. The aim is to identify the outline edges, which are actually visible. Therefore, we sort the remaining edges of the backface culling operation by their distance to the viewer. We select the line closest to the viewer and determine the angular range it occupies. The angle range of the next lines is then compared to the already known occupied range.

### 2.3 Visible Vertical 3D Model Lines

Starting from the visible outlines of the buildings, we add an assumption of height, see Fig. 2.

This results in a 3D building model, that contains all lattice edges of these buildings which should be visible from the viewer's current position.

It has to be noticed, that in comparison with the video image, bottom edges of the generated model are often occluded by unmapped objects. Additionally,



**Fig. 2.** Generation of vertical building edges by using the complete outline

the top edge of the building can not be used for calculation, because the actual height is not known and our assumption of it is certainly incorrect in most cases.

Therefore we reduce the 3D building-environment-model to vertical lattice lines, which represent the vertical building edges within in the image.

### 3 Model and Image Combination

The vertical lattice lines, which are projected into the image, when compared to the real object edges, should present the difference between ground truth pose and the pose hypothesis.

Taking a closer look at the pose of a vehicle, direction and  $x / y$  position are particularly interesting. To test our algorithm, we take the ground truth position as a starting point and distort it randomly, what represents the different hypotheses. The variation of the pose information results from the error range of a GPS receiver, allowing us to simulate GPS inaccuracies while still knowing the ground truth position.

Based on tracking methods like RAPID algorithm [3], the deviation between object and model edges is determined by distance calculation. To do this, we divide the projected model lines into equal line segments. Based on these control points, orthogonal search vectors in the image domain are created, that are used to look for corresponding edges. One of the problems with this approach is the large number of misdetections (i.e. image edges caused by unmapped environment features or image noise) and non-detections (i.e. a model edge that does not find a match in the image domain because the image edge is too weak). We use a RANSAC<sup>4</sup>-type algorithm [10] to filter all extracted object edges to remove outliers, but this does not remove all outliers reliably and it still leaves the problem of non-detections.

Our goal is to obtain a function, that compares the hypothetical edge model with the edges found in the image and delivers a clear maximum, where the position hypothesis and ground truth are close. Therefore a good weight for the interpolated points with and without distance information has to be found.

<sup>4</sup> **Random Sample Consensus.**

## 4 Determination of a Quality Factor

Different mathematical methods to determine the quality factor are considered according to performance and functional characteristics. The generated quality function using the distance values is being examined for its maxima. It is our goal, that the maximum of the quality function is at close range to the ground truth position. The analysis was started using the weighting  $W_1$ , where the quality is being calculated for every hypotheses  $s_{i,t}$ :

$$W_1(s_{i,t}) = \frac{n_{\text{gef}}}{\sum_{j=0}^{n_{\text{gef}}} l_j} \quad (1)$$

Here the amount of control points  $n_{\text{gef}}$ , where a corresponding edge to an image model edge was found, is divided by the sum of the pixel lengths  $l_i$  of the normal vectors of all found control points. It is assumed, that the current projection of the model edge for the actual position and direction gets the better the more corresponding control points are found and the shorter the lengths of the normals are. Tests carried out, using the this weighting on an idealized model, at first confirmed this approach, because the weight of model edges and the extracted edges from the video image are lying nearly one upon the other.

However the method is very vulnerable to inhomogeneous areas or occlusions in the video image where the method determines values nearly as high as for the appropriate model. As the weighting  $W_1(s_{i,t})$  is rating most of the hypotheses with the same quality, the hypotheses are never converging to one specific point when performing tests with a high amount of iterations.

The weighting  $W_1(s_{i,t})$  delivers the same value to an independent amount of control points  $n_{\text{gef}}$  with constant normal lengths  $l$ . But an urban scenario is normally characterized by high amounts of adjacent buildings and many image edges.

Therefore, in addition to the found control points  $n_{\text{gef}}$ , we also consider the demanded control points  $n_{\text{ges}}$ , even if they do not yield a match. This approach appears in weighting  $W_2(s_{i,t})$ . It is assumed, that the result improves by the relation  $\frac{n_{\text{gef}}}{n_{\text{ges}}}$  between found control points and demanded control points. Therefore the outcome should be the more precise the more corresponding control points were found within the video image. As a result, the amount of found control points  $n_{\text{gef}}$  is squared to give more weight to those edges, that result from a high amount of found pixels. Additionally we square the relation  $\frac{n_{\text{gef}}}{n_{\text{ges}}}$  to give it more weight.

This results in an extended weighting  $W_2(s_{i,t})$ :

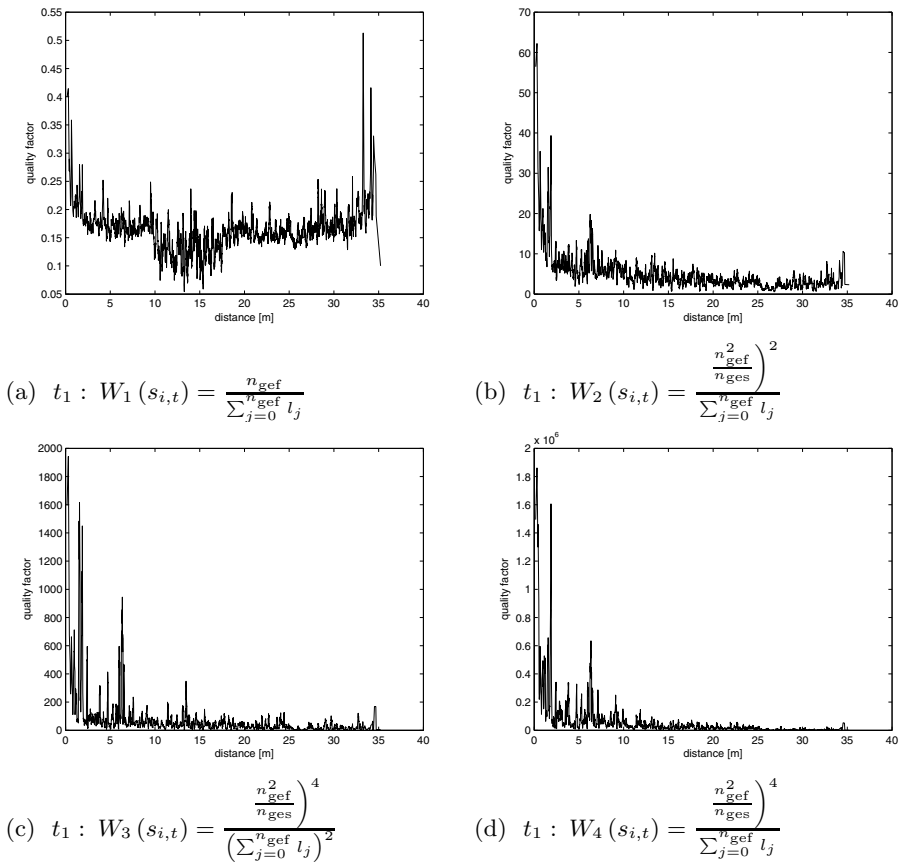
$$W_2(s_{i,t}) = \frac{\left(\frac{n_{\text{gef}}}{n_{\text{ges}}}\right)^2}{\sum_{j=0}^{n_{\text{gef}}} l_j} \quad (2)$$

We evaluate more than these two weightings described here in detail, to generate a convenient propability function. Especially the ratio between control points and distance values is additionally varied.

## 5 Practical Results for Quality Factor

The variation of the mathematical methods is analysed by the different quality functions, which are shown in Fig. 3. A smoothed result of quality factor for each position hypothesis is represented in these diagrams, that means the average of all hypotheses in an area of 10cm is calculated for eliminating single outliers. Single hypotheses with a high quality factor which have nearly identical distances to ground truth, but different position values are downgraded. The comparison of the different equations shows, that the used values are basically correct, only the ratio between control points and distance values has to be balanced. The diagram 3 a) shows different peaks, whereas the other mathematical equations b) - d) deliver the assumed maximum at ground truth position.

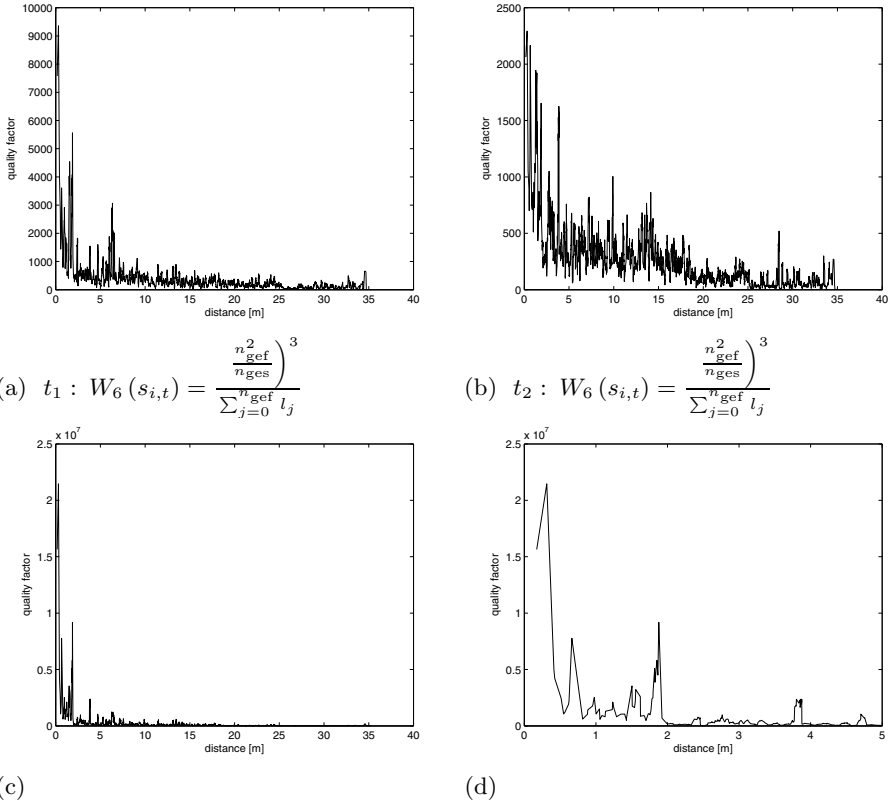
The equation relating to Fig. 4 is extracted for further consideration, because of the clear maximum at ground truth position, compared to the other lower



**Fig. 3.** The diagrams show the results of the different mathematical methods for calculating the quality factor. The x axis of the coordinate system represents the distance values relating to ground truth position. a) - d) Smoothed quality functions.



peaks. The diagrams 4 a) and b) show the result at time  $t_1$  and after a time step by  $t_2$ . The quality function of the second diagram is more distorted with different high potential peaks. The Fig. 4 c) illustrates the multiplication of the two functions above and a clear maximum at ground truth position is shown. So time analysis eliminates the disruptive peaks, what justifies the utilization of a Particle filter (compare 8) for pose estimation. Therefore the determined weighting for each position hypothesis is transformed into a probability density function. The zoomed in figure supports our expectancy to reach an accuracy of less than 1m by the combined sensor approach.



**Fig. 4.** The diagrams show the result of the time analysis. a) Smoothed quality function at time  $t_1$  b) Smoothed quality function at time  $t_2$  c) Multiplied quality functions of  $t_1$  and  $t_2$  d) Zoomed-in on diagram c.

## 6 Conclusion

With the combined use of map material and image data, we are able to generate a probability density function, which shows a clear maximum near ground truth position. It does not always find one single maximum only, but false / additional

maxima are shifting around, caused by movement of the car. This is not the case with the main maximum, what makes the probability density function a good candidate for further filtering. We hope to reduce the deviation of the GPS sensor (currently more than 20m) to less than 1m with this approach. So transferring the preferred weighting to a Particle filter for pose estimation turned out to be a promising method.

## References

1. Nischwitz, A., Fischer, M., Haberäcker, P.: Computer Graphics and Image Processing, pp. 67–68. Friedr. Vieweg u. Sohn Verlag, GWV Fachbuchverlag, Wiesbaden (2007) (in German)
2. Lowe, D.G.: Fitting Parameterized Three-Dimensional Models to Images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 441–450 (1991)
3. Harris, C., Stennet, C.: RAPID - A Video Rate Object Tracker. In: *Proceedings of the British Machine Vision Conference*, pp. 73–77 (1990)
4. Armstrong, M., Zissermann, A.: Robust object tracking. In: *Proceedings of the Asian Conference on Computer Vision*, pp. 58–62 (1995)
5. Lepetit, V., Fua, P.: Monocular Model-Based 3D Tracking of Rigid Objects: A Survey. In: *Foundations and Trends in Computer Graphics and Vision* (2005)
6. Davison, A.J.: Real-time simultaneous localisation and mapping with a single camera. In: *Proceedings of the International Conference on Computer Vision* (2003)
7. Davison, A.J., Murray, D.W.: Mobile Robot Localisation Using Active Vision. In: *Proceedings of Fifth European Conference on Computer Vision*, pp. 809–825 (1998)
8. Dellart, F., Fox, D., Burgard, W., Thrun, S.: Monte Carlo Localisation for Mobile Robots. In: *IEEE International Conference on Robotics and Automation* (1999)
9. Mathias, A., Kanther, U., Heidger, R.: Insideness and collision detection algorithm. In: *Proc. Tyrrhenian International Workshop on Digital Communications - Enhanced Surveillance of Aircraft and Vehicles* (2008)
10. Fischler, M.A., Bolles, R.C.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. In: *Readings in computer vision: issues, problems, principles and paradigms*, pp. 726–740 (1987)
11. Schönherr, K., Giesler, B., Knoll, A.: Vehicle Localization by Utilization of Map-based Outline Information and Grayscale Image Extraction. In: *Proceedings of the International Conference on Computer Graphics and Imaging* (2010)

# Beyond Keypoints: Novel Techniques for Content-Based Image Matching and Retrieval

Andrzej Śluzek<sup>1,2</sup>, Duanduan Yang<sup>1</sup>, and Mariusz Paradowski<sup>1,3</sup>

<sup>1</sup> Nanyang Technological University, SCE,  
Blk N4 Nanyang Av, Singapore 639798

<sup>2</sup> Nicolaus Copernicus University, Faculty of Physics, Astronomy and Informatics,  
ul.Grudziadzka 5, 87-100 Toruń, Poland

<sup>3</sup> Wrocław University of Technology, Faculty of Comp. Science and Management,  
ul.Lukasiewicza 5, 50-371 Wrocław, Poland  
{assluzek, ddyang, mparadowski}@ntu.edu.sg.com

**Abstract.** Keypoints are a well established tool for image matching and retrieval problems. The paper reports development of novel techniques that (by exploiting advantages of keypoints and trying to correct their certain inadequacies) provide higher accuracy and reliability of content-based image matching. The area of ultimately intended applications is *near-duplicate image fragment* retrieval, a difficult problem of detecting visually similar fragments embedded into images of unknown and unpredictable contents. Two supplementary approaches are proposed: (1) *image warping* for non-linearly distorted images to obtain the best match between related fragments and (2) detection of maximum regions that are related by affine transformations. Other relevant results are also briefly mentioned. The reported work is a part of an ongoing project so that further improvements and modifications of the proposed methods can be expected in the near future.

**Keywords:** keypoints, keypoint descriptors, image matching, TPS warping, affine transformation, sub-image retrieval.

## 1 Introduction

The original idea of keypoints has been proposed almost 30 years ago (e.g. [1], [2]). *Keypoints* (*interest points*) indicate image fragments with distinctive characteristics. It is generally assumed those characteristics are so prominent that whenever the objects shown in an image appear in another image, most of such fragments would be again detected as keypoints. Therefore, by identifying (matching) corresponding pairs of keypoints, the image contents can be locally compared and/or similar images can be eventually found.

First applications were relatively simple (stereovision, [1]) with correspondingly simple keypoints (corner points detected over small areas, e.g. [1], [2]). In the following years, a more advanced problem of image matching (for search, retrieval, etc.) emerged as the primary application of keypoints. Then, keypoints

also became more advanced, first in a form of circular areas of the appropriate scale (e.g. [3]) and later in a form of ellipses, [4], that (partially) addressed the problem of perspective distortions. For such keypoints, the term *keyregions* seems more appropriate.

Keypoints are typically characterized by  $n$ -dimensional descriptors (representing selected local properties of image intensities or colours) so that similarity between keypoints can be measured as a distance between  $n$ -dimensional points. Various keypoint descriptors have been proposed (e.g. [5], [6], [7]) and benchmarked (see [8]). Until recently, SIFT (see [5]) has been considered the most effective one, but we have found SURF (see [9]) a better performer.

Hundreds/thousands of keypoints typically detected in a single image provide a large amount of (usually) reliable and useful visual data. Therefore, keypoint-based image matching, search and retrieval have recently gained popularity in various applications (ranging from video browsing, e.g. [10], to urban navigation systems, e.g. [11]). However, such methods often fail in several scenarios. Typical examples of such scenarios are:

1. Images contain similar object(s) but the image backgrounds are different.
2. The image objects are flexible and can be significantly deformed.
3. The same objects appear in different geometric configurations.

In such cases, the majority of matched keypoint pairs are usually outliers, and the primary task is to identify keypoints that should be actually matched. If there is no prior knowledge about the image contents, and if inliers constitute only a small fraction of keypoints, the task can be challenging.

In this paper we overview two novel techniques that can be instrumental in enhancing keypoint-based image matching. The techniques have been developed in a project on visual information retrieval so that image retrieval is the application of primary interest (although other applications can also benefit from the techniques). In particular, our intention is to provide mechanisms for *near-duplicate image fragment* retrieval. The term "near-duplicate image fragment" retrieval is a generalization of near-duplicate retrieval and sub-image retrieval that recently attract interests of the research community (e.g. [12], [13], [14]). We can formulate the problem as follows:

*Given a query image  $Q$  and an image database  $S$ , retrieve from  $S$  images containing fragments which are near-duplicates of unspecified fragments of  $Q$ . "Near-duplicates" refer to almost exact duplicates of image fragments that differ, however, in scene and camera settings, photometric conditions, digitization parameters and possibly geometric distortions of the underlying objects.*

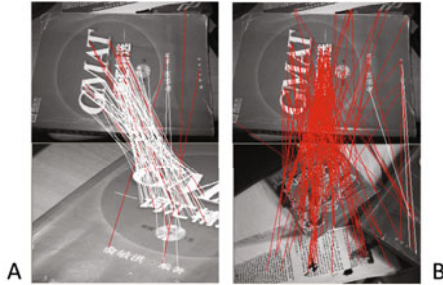
We believe the proposed methods are important steps towards the solution of such a problem. Thus, in Section 2 of the paper we define the scope of the developed techniques, while Sections 3 and 4 present the techniques. Image alignment using *thin plate spline* (TPS) warping is discussed in Section 3. The method attempts to align images based on the distribution of the most reliable keypoint matches. Section 4 explains how the histograms of affine transforms can be

used for determining image fragments related by such transformations. Section 5 concludes the paper.

## 2 Keypoint Matching for Image Matching

When keypoints are characterized by  $n$ -dimensional descriptors, keypoint matching becomes a standard problem of pattern analysis. Various options exist for keypoint correspondences (see [13]) including many-to-many (M2M), one-to-many (O2M), and one-to-one (O2O) variants. O2O assumes that two keypoints are matched if their descriptors are mutual nearest neighbours. Usually, O2O matching provides the highest precision of matching and, therefore, the O2O matching scheme is adopted in this paper. Pairs of keypoints matched by the O2O scheme will be referred to as *coherent pairs* of keypoints.

Even though O2O provides higher precisions than other matching schemes (with fewer matches, though) the image matching results based on the number of matched pairs (e.g. CMK algorithm used in [12]) can be disappointing. Fig. 1 shows exemplary images for which the related images (Fig.1A) have fewer coherent pairs than the unrelated ones (Fig.1B) where most coherent pairs are outliers. Therefore, more advanced algorithms are needed for image matching based on coherent pairs. Two such algorithms are discussed in Sections 3 and 4.



**Fig. 1.** Pairs of images with O2O-matched keypoints. 127 coherent pairs detected for the relevant images (B) and 171 pairs for the irrelevant images (A). White lines indicate correct matches while red lines indicate outliers.

## 3 TPS-Based Image Warping

*Thin plate spline* (TPS) warping, [15], is a method for surface interpolation over scattered data, obtained by minimizing the bending energy of the warping function  $f(x, y)$ . The solution for the warping function  $f(x, y)$ , which is the desired displacement at a point  $(x, y)$ , is expressed as

$$f(x, y) = a_1 + a_x x + a_y y + \sum w_i U(\|(x_i, y_i) - (x, y)\|) \tag{1}$$

where  $U(r) = r^2 \log r^2$  is the *kernel function*, and  $(x_1, y_1), \dots, (x_i, y_i), \dots, (x_n, y_n)$  are *control points* (i.e. points for which the warped coordinates are known).



**Fig. 2.** TPS warping of images using two selections of source and target images

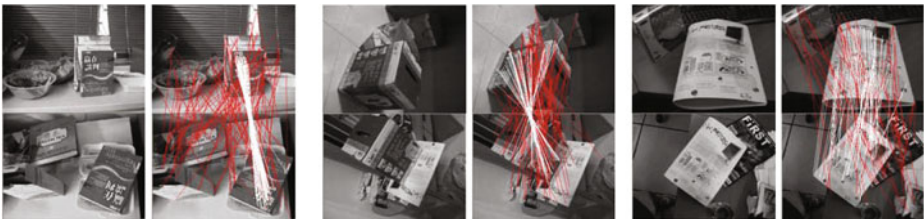
In image matching (where we need two warping functions  $f_x$  and  $f_y$  for  $X$  and  $Y$  displacements) the natural candidates for control points are the most reliable coherent pairs (details of the control points selection and warping function building can be found in [16]):

We arbitrarily select one of the matched images to be the *target image*, while the other one becomes the *source image*. The visual results of image alignment obviously depend on which image is selected as the source (see Fig. 2) but our experiments show that in the contexts of image retrieval the performance is not affected by the choice. If the images actually have similar contents, the warped source images would be brought very closely to the target images (or to their relevant fragments).

### 3.1 Outlier Removal

We can also conclude that if images contain different fragments, the source coherent keypoints would **not** be brought by TPS warping to their target counterparts. Therefore, we can use TPS warping as a mechanism for removal of outliers, i.e. coherent pairs of keypoints would be rejected if after the warping the coordinates are clearly incorrect.

The remaining coherent pairs of keypoints indicate similar fragments in the matched images. Our experiments show that even if the underlying objects are nonlinearly distorted, similar fragments are usually reliably detected. Several examples are shown in Fig. 3.



**Fig. 3.** TPS-based removal of outliers for exemplary images (the removed coherent pairs are connected by red lines)



**Fig. 4.** Exemplary results of sub-image retrieval (using CMK algorithm) after a prior TPS-based removal of outliers

In spite of certain limitations, we have found the TPS-based outlier removal a very effective tool for sub-image retrieval. Illustrative results are given in Fig. 4 where the leftmost images are queries for which four exemplary relevant images retrieved by CMK algorithm (with a prior outlier removal) are shown. Since these examples illustrate *sub-image retrieval* performances, the shapes matching the queries are outlined in the relevant images. Strong deformations (including nonlinear distortions) compared to the queries can be noticed.

## 4 Histograms of Affine Transformations

In this algorithm, we try to solve the problem of near-duplicate sub-image retrieval for planar objects, i.e. for regions related by affine transformations.

Using any two triplets of coherent pairs from two images, we can build an affine transformation. Therefore, a large number of such transformations can be built for a pair of images, and the corresponding histogram in the parameter space of affine transforms could be accumulated. If certain regions are affine-related, the coherent triangles (i.e. triplets of coherent pairs) from those regions should contribute only to a single bin of the histogram. Alternatively, peaks of the histogram identify affine transformations for affine-related regions. It should be noted that no prior knowledge about the image contents is assumed. A similar approach was proposed (regarding scaling and rotation only) in [17].

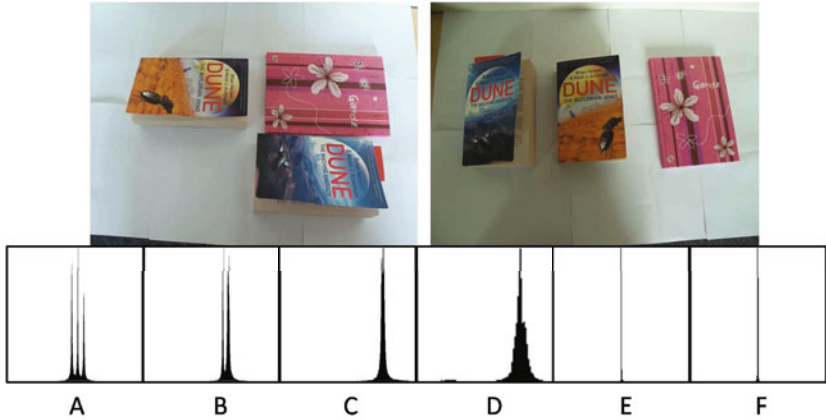
There are, nevertheless, certain practical limitations of the above idea, for which the following solutions have been proposed:

1. Affine transformations need six parameters and 6D histogram cannot be easily constructed and processed (e.g. memory allocation issues).

*Solution:* The histograms are built using hash-tables. Each histogram bin is represented as a single entry in the hash table containing data about contributing triangles and affine transforms belonging to that bin. This is an efficient solution since the data dimensionality is much lower than the total dimensionality of the histogram itself.

2. The algebraic description of affine transforms is not very informative for determining the underlying deformations/motions of image objects.

*Solution:* Affine transformations are decomposed into geometrically meaningful forms. We actually use both SVD decomposition (see [18]) which



**Fig. 5.** A pair of matched images and histograms of SVD-decomposed affine transforms (A, B translations; C, D 2D rotations; E, F scales)

incorporates only 2D transformations, and a decomposition emulating 3D motions of the depicted objects (including 3D rotations using the Euler angles formalism).

3. With a growing number of coherent pairs, the number of coherent triangles would grow correspondingly ( $O(n^3)$ ). Additionally, the importance of coherent triangles varies with their geometry. For example, it is unlikely the whole images are affine-related so that triangles spanning over large areas are not significant.

*Solution:* We discard triangles that are too large, too small and triangles with too small angles (they might be affected by precision and digitization errors). Secondly, a uniform distribution of triangles over the whole image is obtained by building triangles using only  $m$  neighbours of each coherent



**Fig. 6.** Pairs of affine-related regions detected in exemplary images. Note that some regions are only approximately planar.



keypoint (even if more triangles can be built). The value of  $m$  is an important parameter, as it decides how many triangles will be generated altogether.

Fig. 5 gives an exemplary pair of matched images and the corresponding profiles of the affine transformation histogram (projected onto the relevant subspaces for convenient visualization). Prominent spikes representing parameters of transformations between the shown objects can be clearly seen.

From such spikes (using the set of contributing triangles) we can estimate in both images the shapes of fragments related by affine transformations. A detailed description of the algorithm is included in [19], but exemplary results are presented in Fig. 6 (including the results for the Fig. 5 images).

## 5 Conclusions

Performances of the proposed two methods indicate they are significant steps towards a general solution of the *near-duplicate image fragment* retrieval problem. The first method can extract fragments similar to a given query even in the presence of strong non-linear deformations. The second method extracts in arbitrary images fragments related by affine transformations and characterizes their relative 3D (or 2D) motions. We can, obviously, envisage a collaboration of the methods. Then, affine-related regions would be clustered into larger, more complex objects. The validity of such clusterizations could be verified by the TPS-based outlier removal.

It should be, nevertheless, highlighted that the paper presents results of an ongoing project so that further changes and modifications of the algorithms are continuously introduced. In particular, we focus on the computational efficiency of the algorithms. Currently, some elements of the algorithms are computationally too expensive for real-time applications and for handling large databases of images.

**Acknowledgments.** The research presented in this paper is a part of A\*STAR Science & Engineering Research Council grant 072 134 0052. The financial support of SERC is gratefully acknowledged.

## References

1. Moravec, H.: Rover visual obstacle avoidance. In: Int. Joint Conf. on Artificial Intelligence, Vancouver, pp. 785–790 (1981)
2. Harris, C., Stephens, M.: A combined corner and edge detector. In: 4th Alvey Vision Conference, pp. 147–151 (1988)
3. Schmid, C., Mohr, R., Bauckhage, C.: Evaluation of interest point detectors. Int. J. Computer Vision 37(2), 151–172 (2000)
4. Mikolajczyk, K., Schmid, C.: Scale and affine invariant interest point detectors. Int. J. Computer Vision 60(2), 63–86 (2004)
5. Lowe, D.: Distinctive image features from scale-invariant keypoints. Int. J. Computer Vision 60(2), 91–110 (2004)

6. Mindru, F., Tuytelaars, T., van Gool, L., Moons, T.: Moment invariants for recognition under changing viewpoint and illumination. *Computer Vision & Image Understanding* 94(1-3), 3–27 (2004)
7. Islam, M.S., Sluzek, A.: Relative scale method to locate an object in cluttered environment. *Image and Vision Computing* 26(3), 259–274 (2008)
8. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. *IEEE Trans. PAMI* 27, 1615–1630 (2005)
9. Bay, H., Ess, A., Tuytelaars, T., van Gool, L.: Surf: Speeded up robust features. *Computer Vision and Image Understanding* 110(3), 346–359 (2008)
10. Zhao, W.-L., Jiang, Y.-G., Ngo, C.-W.: Keyframe retrieval by keypoints: Can point-to-point matching help? In: *Int. Conf. on Image and Video Retrieval*, pp. 72–81 (2006)
11. Zhang, W., Kosecka, J.: Image based localization in urban environments. In: *3rd Int. Symp. on 3D Data Processing, Visualization and Transmission*, pp. 33–40 (2006)
12. Ke, Y., Sukthankar, R., Huston, L.: Efficient near-duplicate detection and sub-image retrieval. In: *ACM Multimedia Conference*, pp. 869–876 (2004)
13. Zhao, W.-L., Ngo, C.-W., Tan, H.-K., Wu, X.: Near-duplicate keyframe identification with interest point matching and pattern learning. *IEEE Transactions on Multimedia* 9(5), 1037–1048 (2007)
14. Luo, J., Nascimento, M.A.: Content based sub-image retrieval via hierarchical tree matching. In: *1st ACM Int. Workshop on Multimedia Databases*, pp. 2–9 (2004)
15. Bookstein, F.L.: Principle warps: thin plate splines and the decomposition of deformations. *IEEE Trans. PAMI* 16, 460–468 (1989)
16. Yang, D., Sluzek, A.: Aligned matching: an efficient image matching technique. In: *IEEE Conf. Image Proc. ICIP*, pp. 165–168 (2009)
17. Zhao, W.-L., Ngo, C.-W.: Scale-rotation invariant pattern entropy for keypoint-based near-duplicate detection. *IEEE Trans. on Image Processing* 18(2), 412–423 (2009)
18. Xiao, J., Shah, M.: Two-frame wide baseline matching. In: *9th IEEE Int. Conf. on Computer Vision*, pp. 603–609 (2003)
19. Paradowski, M., Sluzek, A.: Matching planar fragments of images using histograms of decomposed affine transforms. NTU Singapore (2009) (unpublished)

# Sequential Coordinate-Wise DNMF for Face Recognition

Rafal Zdunek<sup>1</sup> and Andrzej Cichocki<sup>2,3,4</sup>

<sup>1</sup> Institute of Telecommunications, Teleinformatics and Acoustics,  
Wroclaw University of Technology, Wybrzeze Wyspianskiego 27,  
50-370 Wroclaw, Poland

<sup>2</sup> Laboratory for Advanced Brain Signal Processing  
RIKEN BSI, Wako-shi, Japan

<sup>3</sup> Warsaw University of Technology, Poland

<sup>4</sup> Systems Research Institute, Polish Academy of Science (PAN), Poland

**Abstract.** This paper proposes the Sequential Coordinate-Wise Algorithm (SCWA) to Discriminant Nonnegative Matrix Factorization (DNMF) for improving face recognition. DNMF incorporates Linear Discriminant Analysis (LDA) into NMF using the multiplicative updating rules that are simple in use but usually require many iterations to converge and they do not guarantee the convergence to a stationary point. The SCWA solves the Quadratic Programming (QP) problem by updating only a single variable at each iterative step, which considerably accelerates the convergence for sequentially projected Least Squares (LS) problems that take place in NMF. Moreover, the limit point of the SCWA is the stationary point. The proposed algorithm is tested for supervised face recognition problems where the facial images are taken from the ORL database.

## 1 Introduction

Facial images may be distinguished by various illuminations, facial expressions, viewing conditions, ageing, occlusions such as individuals wearing glasses or scarfs. Due to this diversity the inner-class similarities are very weak with reference to outer-class dissimilarities. For these reasons, face recognition is one of the most challenging applications of image classification.

NMF [1] belongs to the class of unsupervised learning methods, and it is able to decompose facial images into basis vectors that capture nonnegative parts-based representations such as eyes, hair, ears, mouths, etc. Hence, each image can be represented by a linear and additive (not subtractive) combination of basis vectors. Due to the additivity and parts-based representations, NMF outperforms PCA, especially with handling partial occlusions and some illumination problems [2,3,4].

However, the basic NMF algorithms proposed by Lee and Seung [1] do not usually provide perfect spatially localized representations, and many more complicated algorithms for NMF have been proposed to better tackle this problem. One of them is Local NMF (LNMF) [2] that incorporates additional constraints

into the original cost functions to improve the locality of the learned features. The constraints attempt to make the basis vectors as orthogonal as possible, while maximizing the variance of the components (coefficients of linear combinations). Fisher NMF (FNMF) [5] combines LDA with NMF by imposing the constraints on the coefficients of linear combinations in such a way that the inner-class scatter is minimized and the outer-class scatter is maximized. The inner-class scatter models the dispersion of vectors that belong to the same class around their mean, while the outer-class scatter refers to distances of the class mean vectors around the global mean. The minimization of the Fisher criterion is also used in Discriminant NMF (DNMF) [6].

All the above-mentioned NMF algorithms are based on the multiplicative updating rules that ensure a non-increasing behavior in minimizing the cost function. However, they may fail to converge to a stationary point [7] according to the Karush Kuhn Tucker (KKT) optimality conditions that are necessary for a solution of the constrained optimization problem to be optimal. One of strategies to tackle the convergence problem is to apply the Projected Gradient (PG) rules [8] directly to the penalized cost function. Kotsia *et al* [9] applied the PG algorithm [8] with the Armijo rule to the multi-criteria cost function taken from DNMF, and they obtained the PGDNMF algorithm that outperforms the standard DNMF algorithm.

In this paper, we take advantage from the concept of DNMF and formulate the penalized quadratic function which we minimize with the Sequential Coordinate-Wise Algorithm (SCWA) that was first developed by Franc *et al.* [10], and adopted to NMF problems in [11,8]. Our choice is justified by the fact that the limit point of the SCWA is a stationary point, and DNMF uses the discriminant information on the training vectors, which is particularly suitable for classification problems.

The remainder has the following organization. Section 2 introduces to NMF, LNMF and DNMF. The SCW-DNMF algorithm for face recognition is derived in Section 3. The experiments are given in Section 4. Finally, Section 5 contains the brief conclusions.

## 2 Nonnegative Matrix Factorization

The aim of NMF is to find such lower-rank nonnegative matrices  $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{I \times J}$  and  $\mathbf{X} = [x_{jt}] \in \mathbb{R}^{J \times T}$  that  $\mathbf{Y} \cong \mathbf{AX} \in \mathbb{R}^{I \times T}$ , given the data matrix  $\mathbf{Y}$ , the lower rank  $J$ , and possibly some prior knowledge on the matrices  $\mathbf{A}$  or  $\mathbf{X}$ . Assuming each column vector of  $\mathbf{Y} = [y_{it}] = [\mathbf{y}_1, \dots, \mathbf{y}_T]$  represents a vectorized image from a set of training images, and  $J$  is *a priori* known number of (basis vectors) components, then we can interpret the column vectors of  $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_J]$  as the (basis) parts-based representations, and the column vectors of  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_T]$  as feature vectors that contain the nonnegative coefficients of the linear combinations of the basis vectors. Note that each training image represented by the vector  $\mathbf{y}_t$  should be distinguished by its own feature vector, and the facial images belonging to the same person should have similar feature

vectors. Thus, the classification can be easily performed in the reduced dimension feature space (typically  $J \ll \min\{I, T\}$ ).

A family of NMF algorithms can be derived considering the generalized KL divergence with the constraining terms  $\Psi$  and  $\Phi$ :

$$D(\mathbf{Y}||\mathbf{A}\mathbf{X}) = \sum_{i,t} y_{it} \log \frac{y_{it}}{[\mathbf{A}\mathbf{X}]_{it}} - y_{it} + [\mathbf{A}\mathbf{X}]_{it} + \gamma\Psi - \delta\Phi, \tag{1}$$

where  $\gamma, \delta > 0$ .

In LNMF [2],  $\Psi = \sum_{m=1}^J \sum_{n=1}^J [\mathbf{A}^T \mathbf{A}]_{mn}$ ,  $\Phi = \sum_{j=1}^J [\mathbf{X}\mathbf{X}^T]_{jj}$ ,  $\gamma > 0$  controls the degree of orthogonality between the basis vectors, and  $\delta > 0$  controls the variance of the components (row vectors) in  $\mathbf{X}$ .

The additional penalty (or regularization) terms in DNMF [5,6] are defined as  $\Psi = \text{tr}\{\mathbf{S}_X\}$  and  $\Phi = \text{tr}\{\mathbf{S}_{\bar{X}}\}$ , where  $\text{tr}\{\mathbf{Z}\}$  denotes the trace of  $\mathbf{Z}$ , and

$$\mathbf{S}_X = \sum_{k=1}^K \sum_{\rho=1}^{|\mathcal{N}_k|} (\mathbf{x}_\rho^{(k)} - \bar{\mathbf{x}}^{(k)})(\mathbf{x}_\rho^{(k)} - \bar{\mathbf{x}}^{(k)})^T, \tag{2}$$

$$\mathbf{S}_{\bar{X}} = \sum_{k=1}^K |\mathcal{N}_k| (\bar{\mathbf{x}}^{(k)} - \bar{\mathbf{x}})(\bar{\mathbf{x}}^{(k)} - \bar{\mathbf{x}})^T, \tag{3}$$

where  $K$  is the number of classes,  $|\mathcal{N}_k|$  is the number of facial images in the  $k$ -th class,  $\mathbf{x}_\rho^{(k)}$  is the  $\rho$ -th image in the  $k$ -th class,  $\bar{\mathbf{x}}^{(k)}$  is the mean vector of the  $k$ -th class, and  $\bar{\mathbf{x}}$  is the mean vector over all the column vectors in  $\mathbf{X}$ . The matrix  $\mathbf{S}_X$  defines the inner-class scatter, i.e. the scatter of the coefficients in  $\mathbf{X}$  around their class means. The outer-class scatter is defined by the matrix  $\mathbf{S}_{\bar{X}}$  that informs about the distances between the class means. The matrices  $\mathbf{S}_X$  and  $\mathbf{S}_{\bar{X}}$  are adapted to NMF from LDA, and come from the well-known Fisher discrimination criterion.

Both LNMF and DNMF involve the multiplicative update rules to minimize the cost function in (1). Hence, the algorithms can easily get stuck in local minima, and their limit point may not be a stationary point [7]. In the next section, we propose a different approach to the minimization of the cost function extended with the discriminant terms.

### 3 Sequential Coordinate-Wise Algorithms

Let  $k \in \mathcal{K} = \{1, \dots, K\}$ , where  $\mathcal{K}$  is the set of the indices of  $K$  classes, and  $\mathcal{Z} = \{z_t\}_{t=1, \dots, T}$ ,  $\mathcal{Z} = \mathcal{K}$ , where  $z_t$  denotes the index of the class to which the facial image  $\mathbf{y}_t$  belong. Let  $\mathcal{N}_k = \{t : z_t = k\}$  be the set of indices of the column vectors in  $\mathbf{Y}$  that belong to the  $k$ -th class, and  $|\mathcal{N}_k|$  is the cardinality of the set  $\mathcal{N}_k$ . Let  $\mathbf{M} = [m_{st}] \in \mathbb{R}^{T \times T}$ , and for  $k = 1, \dots, K$ :

$$m_{st} = \begin{cases} \frac{1}{|\mathcal{N}_k|} & \text{if } (s, t) \in \mathcal{N}_k \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

Note that the matrix  $\mathbf{S}_X$  in (2) can be rearranged using the matrix  $\mathbf{M}$  in the following form:

$$\mathbf{S}_X = (\mathbf{X} - \mathbf{X}\mathbf{M})(\mathbf{X} - \mathbf{X}\mathbf{M})^T = \mathbf{X}\mathbf{W}_1\mathbf{X}^T, \tag{5}$$

where  $\mathbf{W}_1 = (\mathbf{I}_T - \mathbf{M})(\mathbf{I}_T - \mathbf{M})^T$ , and  $\mathbf{I}_T \in \mathbb{R}^{T \times T}$  is an identity matrix. Similarly, the matrix  $\mathbf{S}_{\bar{X}}$  in (3) can be rewritten as

$$\mathbf{S}_{\bar{X}} = (\mathbf{X}\mathbf{M} - \mathbf{X}\mathbf{E}_T)(\mathbf{X}\mathbf{M} - \mathbf{X}\mathbf{E}_T)^T = \mathbf{X}\mathbf{W}_2\mathbf{X}^T, \tag{6}$$

where  $\mathbf{E}_T = \frac{1}{T}\mathbf{1}\mathbf{1}^T$ ,  $\mathbf{1} = [1, 1, \dots, 1]^T \in \mathbb{R}^T$ , and  $\mathbf{W}_2 = (\mathbf{M} - \mathbf{E}_T)(\mathbf{M} - \mathbf{E}_T)^T$ .

Thus, the constraining term  $\gamma\Psi - \delta\Phi$  in (1) can be reformulated as  $\gamma\Psi - \delta\Phi = \text{tr}\{\mathbf{X}\mathbf{W}\mathbf{X}^T\} = \|\mathbf{X}\mathbf{W}^{\frac{1}{2}}\|_F^2$ , where  $\mathbf{W} = \gamma\mathbf{W}_1 - \delta\mathbf{W}_2$ .

Assuming the measure of similarity between  $\mathbf{Y}$  and  $\mathbf{A}\mathbf{X}$  is defined by the standard squared Euclidean distance, the cost function penalized with the constraining term  $\gamma\Psi - \delta\Phi$  can be expressed by:

$$D(\mathbf{Y}||\mathbf{A}\mathbf{X}) = \frac{1}{2}\|\mathbf{Y} - \mathbf{A}\mathbf{X}\|_F^2 + \|\mathbf{X}\mathbf{W}^{\frac{1}{2}}\|_F^2, \tag{7}$$

The cost function (7) is quadratic with respect to  $x_{jt}$ , and hence, it can be used to formulate the constrained Quadratic Programming (QP) problem:

$$\min_{\mathbf{x}_t \geq 0} \frac{1}{2}\mathbf{x}_t^T \mathbf{Q}\mathbf{x}_t + \mathbf{c}_t^T \mathbf{x}_t, \quad (t = 1, \dots, T). \tag{8}$$

For the squared Euclidean distance  $\mathbf{Q} = \mathbf{A}^T\mathbf{A}$  and  $\mathbf{c}_t = -\mathbf{A}^T\mathbf{y}_t$ . The QP problem (8) can be also expressed in the equivalent matrix form:

$$\min_{\mathbf{x} \geq 0} \frac{1}{2} \text{tr} \{ \mathbf{X}^T \mathbf{Q} \mathbf{X} \} + \text{tr} \{ \mathbf{C}^T \mathbf{X} \}, \tag{9}$$

where  $\mathbf{C} = -\mathbf{A}^T\mathbf{Y}$ . For the penalized cost function in (7), the penalized QP problem is as follows:

$$\min_{\mathbf{x} \geq 0} \frac{1}{2} \text{tr} \{ \mathbf{X}^T \mathbf{A}^T \mathbf{A} \mathbf{X} \} - \text{tr} \{ \mathbf{Y}^T \mathbf{A} \mathbf{X} \} + \text{tr} \{ \mathbf{X} \mathbf{W} \mathbf{X}^T \}. \tag{10}$$

There are many efficient algorithms for solving QP problems, however, due to the nonnegativity constraints, penalized term, and easy separability of the variables, the SCWA proposed by V. Franc et al. [10] seems to be a good choice. This algorithm has been already applied to the NMF problems in [11, 8] with satisfactory results. The SCWA solves the QP problem given by (8) updating only a single variable  $x_{jt}$  at each iterative step. Thus, assuming  $\mathbf{W} = [w_{pr}]$ ,

$p, r \in \{1, \dots, T\}$ ,  $n, m = \{1, \dots, J\}$ , we rewrite the cost function in (7) in the following form:

$$\begin{aligned}
 D(\mathbf{Y} \|\mathbf{A}\mathbf{X}) &= \frac{1}{2} \sum_{p,n,m} x_{np}x_{mp}(\mathbf{A}^T\mathbf{A})_{nm} - \sum_{p,n} x_{np}(\mathbf{A}^T\mathbf{Y})_{np} + \sum_{n,p,r} x_{np}w_{pr}x_{nr} \\
 &= \frac{1}{2} \left( (\mathbf{A}^T\mathbf{A})_{jj} + 2w_{tt} \right) x_{jt}^2 \\
 &\quad + \left( \sum_{n \neq j} x_{nt}(\mathbf{A}^T\mathbf{A})_{nj} - (\mathbf{A}^T\mathbf{Y})_{jt} + \sum_{r \neq t} w_{tr}x_{jt} + \sum_{p \neq t} w_{pt}x_{jp} \right) x_{jt} \\
 &\quad + \frac{1}{2} \sum_p \sum_{n \neq j} \sum_{m \neq j} x_{np}x_{mp}(\mathbf{A}^T\mathbf{A})_{nm} - \sum_p \sum_{n \neq j} x_{np}(\mathbf{A}^T\mathbf{Y})_{np} \\
 &\quad + \frac{1}{2} \sum_{p \neq t} (\mathbf{A}^T\mathbf{A})_{jj} x_{jp}^2 + \sum_{p \neq t} \left( \sum_{n \neq j} x_{np}(\mathbf{A}^T\mathbf{A})_{nj} - (\mathbf{A}^T\mathbf{Y})_{jp} \right) x_{jp} \\
 &\quad + \sum_{p \neq t} \sum_{r \neq t} x_{jp}w_{pr}x_{jr} + \sum_{n \neq j} \sum_{p,r} x_{np}w_{pr}x_{nr} \\
 &= x_{jt}^2 \alpha_{jt} + x_{jt} \beta_{jt} + \gamma_{jt}. \tag{11}
 \end{aligned}$$

The optimization of  $D(\mathbf{Y} \|\mathbf{A}\mathbf{X})$  in (11) with respect to the selected variable  $x_{jt}$  gives the following analytical solution:

$$\begin{aligned}
 x_{jt}^* &= \arg \min_{x_{jt}} D(\mathbf{Y} \|\mathbf{A}[x_{jt}]) = \arg \min_{x_{jt}} x_{jt}^2 \alpha_{jt} + x_{jt} \beta_{jt} + \gamma_{jt} \\
 &= \max_{x_{jt}} \left( 0, -\frac{\beta_{jt}}{\alpha_{jt}} \right), \tag{12}
 \end{aligned}$$

where

$$\alpha_{jt} = \frac{1}{2} (\mathbf{A}^T\mathbf{A})_{jj} + w_{tt}, \tag{13}$$

$$\beta_{jt} = \sum_{n \neq j} x_{nt}(\mathbf{A}^T\mathbf{A})_{nj} - (\mathbf{A}^T\mathbf{Y})_{jt} + \sum_{r \neq t} w_{tr}x_{jt} + \sum_{p \neq t} w_{pt}x_{jp}. \tag{14}$$

Since  $\sum_{r \neq t} w_{tr}x_{jr} = \underline{\mathbf{w}}_t \underline{\mathbf{x}}_j^T - w_{tt}x_{jt}$ , and  $\sum_{p \neq t} w_{pt}x_{jp} = \underline{\mathbf{x}}_j \mathbf{w}_t - w_{tt}x_{jt}$ , where  $\underline{\mathbf{w}}_t = [w_{t1}, \dots, w_{tT}] \in \mathbb{R}^T$  and  $\underline{\mathbf{x}}_j = [x_{j1}, \dots, x_{jT}] \in \mathbb{R}^T$  denote the corresponding  $t$ -th row vector of the matrix  $\mathbf{W}$  and the  $j$ -th row vector of the matrix  $\mathbf{X}$ , we can rewrite  $\beta_{jt}$  in (14) as

$$\beta_{jt} = (\mathbf{A}^T\mathbf{A}\mathbf{X} - \mathbf{A}^T\mathbf{Y})_{jt} - (\mathbf{A}^T\mathbf{A})_{jj}x_{jt} + \underline{\mathbf{w}}_t \underline{\mathbf{x}}_j^T + \underline{\mathbf{x}}_j \mathbf{w}_t - 2w_{tt}x_{jt}. \tag{15}$$

Inserting (13) and (15) to (12), the update rule for  $x_{jt}$  is given by

$$x_{jt} \leftarrow \max \left( \epsilon, x_{jt} - \frac{g_{jt} + \underline{\mathbf{w}}_t \underline{\mathbf{x}}_j^T + \underline{\mathbf{x}}_j \mathbf{w}_t}{(\mathbf{A}^T\mathbf{A})_{jj} + 2w_{tt}} \right), \tag{16}$$

where  $g_{jt} = (\mathbf{A}^T \mathbf{A} \mathbf{X} - \mathbf{A}^T \mathbf{Y})_{jt}$ , and  $\epsilon \cong 10^{-16}$  is a small positive constant to avoid numerical instabilities. The update rule in (16) can be regarded as the extension to the standard HALS algorithm [8].

For updating the matrix  $\mathbf{A}$ , we used the regularized Euclidean distance:

$$D(\mathbf{Y} \parallel \mathbf{A} \mathbf{X}) = \frac{1}{2} \|\mathbf{Y} - \mathbf{A} \mathbf{X}\|_F^2 + \frac{\lambda^{(s)}}{2} \|\mathbf{A}\|_F^2. \quad (17)$$

The regularization term constraints the Frobenius norm of  $\mathbf{A}$ , where the degree of constraining is controlled by the exponentially decaying regularization parameter  $\lambda^{(s)} = \bar{\lambda} + \lambda_0 \exp\{-\tau s\}$ , where  $\bar{\lambda} > 0$ ,  $\lambda_0 > 0$ , and  $\tau > 0$  stand for parameters, and  $s$  is the current number of iterative step in the alternating scheme for NMF. Applying the FNMA algorithm [12] to (17), the update rule for  $\mathbf{A}$  is given by

$$\mathbf{A} \leftarrow \max\{\epsilon, \mathbf{A} - \mathcal{Z}_+[\mathbf{P}_A]\}, \quad (18)$$

where  $\mathbf{P}_A = \mathbf{G}_A \mathbf{H}_A^{-1}$ ,  $\mathbf{G}_A = (\mathbf{A} \mathbf{X} - \mathbf{Y}) \mathbf{X}^T + \lambda^{(s)} \mathbf{A}$  and  $\mathbf{H}_A = \mathbf{X} \mathbf{X}^T + \lambda^{(s)} \mathbf{I}_J$  are the gradient and Hessian of (17) with respect to  $\mathbf{A}$ , and  $\mathbf{I}_J \in \mathbb{R}^{J \times J}$  is an identity matrix. The operator

$$\mathcal{Z}_+[\mathbf{P}_A] = \begin{cases} p_{ij}^{(A)} & \text{if } (i, j) \notin \mathcal{A} \\ 0 & \text{otherwise} \end{cases} \quad (19)$$

denotes the projection of  $\mathbf{P}_A$  onto the passive set which is the complement of the active set  $\mathcal{A} = \{i, j : a_{ij} = 0, [\mathbf{G}_A]_{ij} > 0\}$  defined by the KKT conditions. To relax the intrinsic scale ambiguity, the columns of  $\mathbf{A}$  are scaled to the unit  $l_1$ -norm in each alternating step.

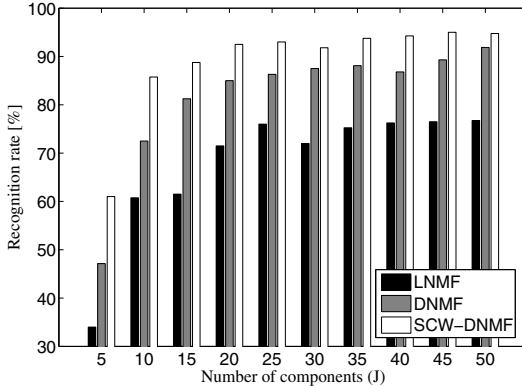
## 4 Experiments

The discussed algorithms are extensively tested under the software developed by L. Klaskala [13]. In this paper, we present only the results obtained for the ORL database<sup>1</sup> that contains 400 frontal face images of 40 people (10 pictures per person). The images were taken at different times (between April 1992 and April 1994 at the AT&T Laboratories Cambridge), varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position. The whole set is randomly divided into 320 training images containing all the classes and 80 testing images. To test the robustness against noisy perturbations, the testing images are corrupted with a strong additive Gaussian zero-mean noise with the variance adjusted to  $SNR = -5$ [dB]. The negative entries in the testing data matrix are replaced with zero-entries.

The experiments are performed according to the scheme proposed in [2, 3]. The matrix of the training images is decomposed into a matrix of parts-based

<sup>1</sup> <http://people.cs.uchicago.edu/~dinoj/vis/orl/>





**Fig. 1.** Recognition rate versus the number of components  $J$  obtained with LNMf, DNMF, and SCW-DNMF. The testing images are corrupted with a strong additive Gaussian noise ( $SNR = -5$ [dB]).

representations (the matrix  $\mathbf{A}$ ) and a matrix of features (the matrix  $\mathbf{X}$ ). Then, the testing images are projected on the feature space using the matrix of parts-based representations with the same algorithm as for the decomposition. The Euclidean distance measure is used to compare the testing feature vectors with the training feature vectors.

The NMF algorithms are initialized with uniformly distributed random matrices, and tested for various values of the related parameters. For each analyzed case, the training process is repeated 10 times with different initial random matrices. Fig. 1 presents the mean recognition rate versus the number of components (parameter  $J$ ) obtained with different NMF algorithms. For the SCW-DNMF, we found the optimal parameters:  $\gamma = 10^{-3}$ ,  $\delta = 10^{-6}$ ,  $\bar{\lambda} = 10^{-12}$ ,  $\lambda_0 = 10^8$ ,  $\tau = 1$ ,  $s = 30$  (number of alternating steps), and the number of inner iterations<sup>2</sup> for updating the matrix  $\mathbf{X}$  (the updating rule (16)) is equal to 100.

## 5 Conclusions

The recognition rate obtained with the proposed SCW-DNMF algorithm is much higher than obtained with LNMf, moderately higher than with DNMF, and quite stable with respect to the number of the components. For the noise-free testing images, the recognition rate increases by about 3 % for the SCW-DNMF, and it is still better than for the LNMf and DNMF. It is thus obvious that the discriminant information under the form of the inner- and outer-class scatters is essential to obtain high performance, especially when a number of part-based representation vectors is small. Our algorithm outperforms DNMF since it is

<sup>2</sup> After performing many tests, we notice that the SCWA must have several inner iterations to reach the high performance [8].

based on the much more robust optimization technique – the SCW QP instead of the multiplicative technique. The use of the exponentially decaying parameter  $\lambda$  in (18) is also very important to avoid getting stuck in local minima.

In the future research, we will test our algorithm for another databases such as Yale, CMCL, and FERET. The problem of estimating the associated parameters will be also studied.

## References

- [1] Lee, D.D., Seung, H.S.: Learning of the parts of objects by non-negative matrix factorization. *Nature* 401, 788–791 (1999)
- [2] Li, S.Z., Hou, X.W., Zhang, H.J., Cheng, Q.S.: Learning spatially localized, parts-based representation. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, vol. 1, pp. I207–I212 (2001)
- [3] Guillaumet, D., Vitrià, J.: Classifying faces with nonnegative matrix factorization. In: *Proc. 5th Catalan Conference for Artificial Intelligence, Castello de la Plana, Spain* (2002)
- [4] Buciu, I., Nikolaidis, N., Pitas, I.: Nonnegative matrix factorization in polynomial feature space. *IEEE Transactions on Neural Networks* 19(6), 1090–1100 (2008)
- [5] Wang, Y., Jia, Y., Hu, C., Turk, M.: Fisher nonnegative matrix factorization for learning local features. In: *Proc. 6th Asian Conf. on Computer Vision, Jeju Island, Korea* (2004)
- [6] Zafeiriou, S., Tefas, A., Buciu, I., Pitas, I.: Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification. *IEEE Transactions on Neural Networks* 17(3), 683–695 (2006)
- [7] Lin, C.J.: On the convergence of multiplicative update algorithms for non-negative matrix factorization. *IEEE Transactions on Neural Networks* 18(6), 1589–1596 (2007)
- [8] Cichocki, A., Zdunek, R., Phan, A.H., Amari, S.I.: *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. Wiley and Sons, Chichester (2009)
- [9] Kotsia, I., Zafeiriou, S., Pitas, I.: Discriminant non-negative matrix factorization and projected gradients for frontal face verification. In: Schouten, B., Juul, N.C., Drygajlo, A., Tistarelli, M. (eds.) *BIOID 2008*. LNCS, vol. 5372, pp. 82–90. Springer, Heidelberg (2008)
- [10] Franc, V., Hlaváč, V., Navara, M.: Sequential coordinate-wise algorithm for the non-negative least squares problem. In: *Gagalowicz, A., Philips, W. (eds.) CAIP 2005*. LNCS, vol. 3691, pp. 407–414. Springer, Heidelberg (2005)
- [11] Zdunek, R., Cichocki, A.: Fast nonnegative matrix factorization algorithms using projected gradient approaches for large-scale problems. *Computational Intelligence and Neuroscience* 2008(939567) (2008)
- [12] Kim, D., Sra, S., Dhillon, I.S.: Fast Newton-type methods for the least squares nonnegative matrix approximation problem. In: *Proc. 6th SIAM International Conference on Data Mining, Minneapolis, Minnesota, USA* (2007)
- [13] Klaskala, L.: *Image recognition with nonnegative matrix factorization*. M.Sc. thesis (supervised by Dr. R. Zdunek), Wrocław University of Technology, Poland (2009) (in Polish)

# A New Image Mixed Noise Removal Algorithm Based on Measuring of Medium Truth Scale

Ning-Ning Zhou<sup>1,2</sup> and Long Hong<sup>1,2</sup>

<sup>1</sup> School of Computing, Nanjing University of Posts and Telecommunications,  
Nanjing 210003, PRC

<sup>2</sup> State Key Laboratory of Software Development Environment, Beihang University,  
Beijing 100191, PRC

{zhounn, hongl}@njupt.edu.cn

**Abstract.** The medium mathematics system is another mathematical tool which deals with fuzzy and uncertain problem. According to the analysis of the features of the image mixed noise, this paper introduces a new image mixed noise removal algorithm based on measuring of medium truth scale. It uses the distance ratio function to detect the noise pixel and to restore the image. The experimental results demonstrate that the new image mixed noise removal algorithm can do better in smoothing mixed noise and preserving details than the classical ones do in subjective aspect and objective aspect, which will lead to its practicable and effective applications in mixed noise removal and image restoration.

**Keywords:** Image mixed noise, measuring of medium truth scale, distance ratio function, noise removal, image restoration.

## 1 Introduction

Image noise removal algorithm is the focal area in the field of digital image processing. Because of their simplicity, being well-established and easy for computation, traditional filters such as Mean filter, Median filter, Wiener filter and so on, are often used subject to certain rectification. But to the mixed noise, traditional filter usually generate poor results. Therefore, some new mathematical and computational methods such as wavelet transformation [1], neural net [2] and genetic algorithm [3] have been widely adopted in image mixed noise removal and restoration. Nevertheless, due to their complexities, effective mixed noise removal methods are yet to be seen.

Imaging process is affected by a variety of factors. Because of the complexity of image information and the strong relations among image pixels, problems with uncertainty and inaccuracy will appear in the image processing. Some scholars introduced fuzzy mathematics into image noise removal and proposed fuzzy noise removal algorithms [4],[5]. Fuzzy mathematics, a mathematical tool which deals with fuzzy and uncertain problems, has yielded excellent results in image noise removal. But the result of fuzzy noise removal algorithm is highly dependent on the membership function which is decided by subjective experience.

The medium mathematics system is another mathematical tool which deals with fuzzy and uncertain problem. This paper introduces the quantize method of measuring of medium truth scale into the image noise removal and presents a new medium mixed noise removal algorithm. The experimental results demonstrate that the new image mixed noise removal algorithm can do better in smoothing mixed noise and preserving details than the classical ones do in subjective aspect and objective aspect.

## 2 The Medium Mathematics System

Medium principle was established by Chinese scholars Zhu Wu-jia and Xiao Xi-an in 1980s.

### 2.1 Basic Symbols of Medium Mathematics System

In medium mathematics system [6], [7], predication (conception or quality) is represented by  $P$ , any variable is denoted as  $x$ , with  $x$  completely possessing quality  $P$  being described as  $P(x)$ . The “ $\neg$ ” symbol stands for inverse opposite negative and it is termed as “opposite to”. The inverse opposite of predication is denoted as  $\neg P$ . Then the concept of a pair of inverse opposite is represented by both  $P$  and  $\neg P$ . Symbol “ $\sim$ ” denotes fuzzy negative which reflects the medium state of “either-or” or “both this-and that” in opposite transition process. The fuzzy negative profoundly reflects fuzziness; “ $\prec$ ” is a truth-value degree connective which describes the difference between two propositions.

### 2.2 Measuring of Medium Truth Scale

According to the concept of super state [8], the numerical value area of generally applicable quantification is divided into five areas corresponding to the predication truth scale, namely  $\neg^+P$ ,  $\neg P$ ,  $\sim P$ ,  $P$ , and  $^+P$ , as shown in Fig. 1. In “True” numerical value area T,  $\alpha_T$  is  $\varepsilon_T$  standard scale of predication  $P$ ; In “False” numerical value area F,  $\alpha_F$  is  $\varepsilon_F$  standard scale of predication  $\neg P$ .

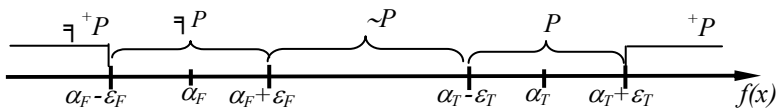


Fig. 1. Relation between numerical value areas and predication

Individual truth scale in each numerical value area can be calculated by the distance ratio  $h_T(f(x))$  (or  $h_F(f(x))$ ) [8] which relates to  $P$  (or  $\neg P$ ).

## 3 Medium Image Mixed Noise Removal Algorithm

The ever-present mixed noise model in the image is the noise composed by Salt-Pepper noise and Gaussian noise. According to the different features of the two kinds

of noise set, the idea of the new algorithm discussed in this paper is as follows: First detect and remove the Salt-Pepper noise by measuring the truth scale of one pixel related to a noise. Then reduce the Gaussian noise by symmetric-mean filtering.

### 3.1 Detection and Elimination of the Salt-Pepper Noise

#### (1) The new Salt-Pepper noise removal algorithm

Noise can be regarded as a disturbance of gray in a gray image. The grey level of pixel at coordinate  $(i,j)$  is expressed as  $x(i,j)$ . Let predication  $Q(x)$  represents that the pixel  $x(i,j)$  is a normal pixel, transition  $\neg Q_L(x)$  and  $\neg Q_R(x)$  represent that the pixel  $x(i,j)$  is a noise, and  $\sim Q_L(x)$  and  $\sim Q_R(x)$  represent that the pixel  $x(i,j)$  is a pixel between the normal pixel and the noise. Make the every gray level of the multi-levels image relate to the different truth area of the predication ( $\neg Q_L$ ,  $\sim Q_L$ ,  $Q$ ,  $\sim Q_R$  and  $\neg Q_R$ ), and establish the standard scales  $\alpha_{FL}, \alpha_{TL}$  which relate to  $Q$  and  $\neg Q_L$ , and the standard scales  $\alpha_{FR}, \alpha_{TR}$  which relate to  $Q$  and  $\neg Q_R$ , as shown in Fig. 2.  $[a,b]$  is the gray level area of normal pixels in the neighborhood of the pixel  $x(i,j)$ .

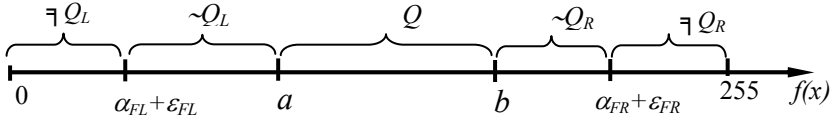


Fig. 2. Relation between image gray level area and predication normal pixel

According to Fig. 3, the distance ratio function  $h(x(i,j))$  can be expressed as follows:

$$h(x(i,j)) = \begin{cases} 0 & x(i,j) \text{ located in the area of } \neg Q_L \\ \frac{|x(i,j) - (\alpha_{FL} + \epsilon_{FL})|}{|a - (\alpha_{FL} + \epsilon_{FL})|} & x(i,j) \text{ located in the area of } \sim Q_L \\ 1 & x(i,j) \text{ located in the area of } Q \\ \frac{|x(i,j) - (\alpha_{FR} + \epsilon_{FR})|}{|b - (\alpha_{FR} + \epsilon_{FR})|} & x(i,j) \text{ located in the area of } \sim Q_R \\ 0 & x(i,j) \text{ located in the area of } \neg Q_R \end{cases} \quad (1)$$

The value of the distance ratio function  $h(x(i,j))$  determines the degree between the pixel and the normal pixel.

When  $h(x(i,j))=1$ , it shows that the pixel  $x(i,j)$  is a normal pixel. When the image is restored, the value of the pixel  $x(i,j)$  should be reserved;

When  $0 < h(x(i,j)) < 1$ , it shows the pixel  $x(i,j)$  is between a normal pixel and a noise. When the image is restored, the value of restored image at the point  $x(i,j)$  is replaced by the weight sum of the original gray value of the pixel  $x(i,j)$  and the mean of the gray levels of the normal pixels in neighborhood of that pixel;

When  $h(x(i,j))=0$ , it shows the pixel  $x(i,j)$  is a noise. When the image is restored, the value of restored image at the point  $x(i,j)$  is replaced by the mean of the gray levels of the normal pixels in neighborhood of that pixel.

To sum up, the new image restoration algorithm can be expressed as follows:

$$x'(i, j) = h(x(i, j)) \times x(i, j) + (1 - h(x(i, j))) \times \bar{x}(s, t) \quad (2)$$

$$(s, t) \in [a, b] \quad \text{and } s \neq i, t \neq j$$

Here  $x'(i, j)$  is the gray value of the restored image at the point  $x(i, j)$ ,  $\bar{x}(s, t)$  is the mean of the gray levels of the normal pixels in neighborhood of that pixel.

## (2) Computation of the new Salt-Pepper noise removal algorithm

The computation of the new algorithm is implemented in following steps:

① Select the noise window.

Select a child window of  $(2n+1) \times (2n+1)$  pixels whose center is at  $(n, n)$  in the image gray matrix. The gray level of pixels in this child window is a domain of set X. The gray level of pixel at coordinate  $(i, j)$  is expressed as  $x(i, j) \in X$ . Let set  $Y = X - x(n, n)$ , that is to say that the set Y is the neighborhood of the pixel  $x(i, j)$ . The center pixel  $x(n, n)$  is the considered point which is decided to be a noise or a normal pixel.

② According to the type of the noise, decide the gray level area of normal pixels.

As to the Salt-Pepper noise, let B is the set of pixels which are in set X but except the maximum pixel and the minimum pixel. That is:

$$B = X - \max(X) - \min(X)$$

Let A is the gray level area of normal pixels, that is:

$$A = [\min(B - \max(B) - \min(B)), \max(B - \max(B) - \min(B))], \text{ denoted as } [a, b].$$

③ Decide the similarity degree between the center pixel  $x(n, n)$  and the normal pixel.

The gray image only includes the brightness information. On the basis of the colorimetric theory, the minimal brightness difference discriminated by eye is nearly 3 [9]. So let  $\alpha_{FL} + \varepsilon_{FL} = 3$ ,  $\alpha_{FR} + \varepsilon_{FR} = (255 - 3) = 252$ . Then according to expression (1), compute the value of the distance ratio function and get the similarity degree between the center pixel  $x(n, n)$  and the normal pixel.

④ According to expression(2), compute the value of restored image at the point  $(n, n)$ .

⑤ Traverse the  $M \times N$  image gray level matrix, repeat the above steps in every child window of  $(2n+1) \times (2n+1)$  pixels and get a restored image of denoising the Salt-Pepper noise.

## 3.2 Elimination of the Gaussian Noise

The Gaussian noise is different from the Salt-Pepper noise which only corrupts some parts of an image, while every pixel of an image corrupted by Gaussian noise is likely to be affected. Furthermore, in the same level of gray, there are obvious differences in the degree of corruption caused by Gaussian noise. Then symmetric-mean filtering can be applied to reduce the effect of the Gaussian noise.

The computation of the symmetric-mean filtering is implemented in following steps:

- ① Select the child window.

Select a child window of  $(2n+1) \times (2n+1)$  pixels whose center is at  $(n,n)$  in the image gray matrix. The gray level of pixel at coordinate  $(i,j)$  is expressed as  $x(i,j)$ . Except the center point, there are  $[(2n+1) \times (2n+1) - 1] / 2$  pairs of symmetric points in this child window, as shown in Fig. 3. For example, pixel  $x(n-1,n-1)$  and pixel  $x(n+1,n+1)$  is a pair of symmetric points, pixel  $x(n-2,n+1)$  and pixel  $x(n+2,n-1)$  is a pair of symmetric points,.....

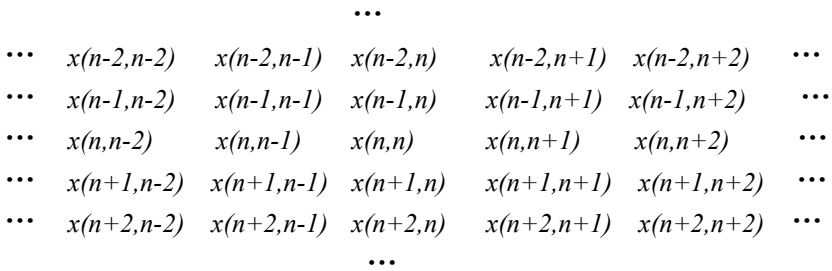


Fig. 3. The child window of  $(2n+1) \times (2n+1)$  pixels

- ② Select the pixel of very pair of symmetrical points whose gray level value is proximate to the  $x(n,n)$ , and is denoted as  $s(1), s(2), \dots, s([(2n+1) \times (2n+1) - 1] / 2)$ .

- ③ Compute the mean of  $s(1), s(2), \dots, s([(2n+1) \times (2n+1) - 1] / 2)$  to replace the value of gray level value at the center. It can be expressed as follows

$$x'(n,n) = \frac{1}{[(2n+1) \times (2n+1) - 1] / 2} \sum_{i=1}^{(2n+1) \times (2n+1) - 1} s(i) \quad (n \geq 1) \quad (3)$$

Here,  $x'(n,n)$  is the value of restored image at the point  $(n,n)$ .

- ④ Traverse the  $M \times N$  image gray level matrix, repeat the above steps in every  $(2n+1) \times (2n+1)$  child window and get a restored image of denoising the Gaussian noise.

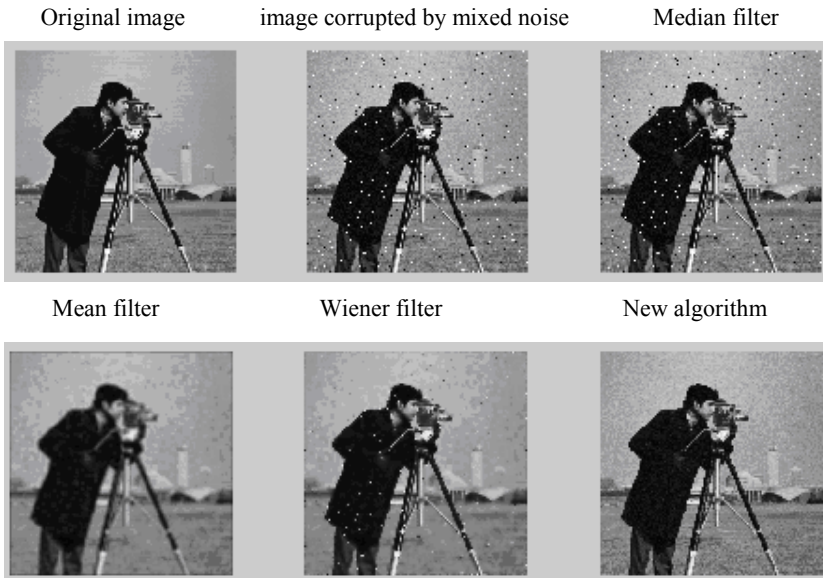
### 3.3 Experimental Result Analysis

Lena image and Cameraman image corrupted by the mixed noise are chosen as experimental samples to verify the noise removal algorithm base on the measuring of medium truth scale described in this paper. Mean filter, Median filter, Wiener filter are used in comparison in the experiment in order to evaluate the performance of the new algorithm in mixed noise removal of the Lena image and the Cameraman image. Set  $n=2$ , the results of the experiment are shown in Fig. 4 and Fig. 5.

The performance of the new algorithm can be evaluated through both subjective visual and objective quality.



**Fig. 4.** Experimental result of mixed noise removal of Lena image ( $\sigma^2 = 0.01$ )



**Fig. 5.** Experimental result of mixed noise removal of Cameraman image ( $\sigma^2 = 0.01$ )

### (1) Subjective visual effect

The experimental results as shown in Fig. 5 and Fig. 6 reveal that the visual effect of mixed noise removal of the new algorithm is better than that of the others such as



Mean filter, Median filter, Wiener filter. There are more image details being preserved by the new algorithm than the other filters. This makes the image become more smooth and distinct.

**(2) Better objective quality**

Peak-Value Signal-to-Noise (PSNR) is a classical evaluation method to noise removal and restoration of image.

The PSNR method is used to evaluate the new algorithm in comparison with Mean filter, Median filter, Wiener filter. The results are shown in Table 1 and Table 2.

**Table 1.** PSNR of algorithms to Lena image with mixed noise

| $\sigma^2$ | Mean filter | Median filter | Wiener filter | New algorithm |
|------------|-------------|---------------|---------------|---------------|
| 0.002      | 12.96       | 52.65         | 57.11         | 58.06         |
| 0.004      | 12.88       | 49.56         | 52.84         | 57.43         |
| 0.006      | 12.83       | 47.45         | 50.19         | 56.33         |
| 0.008      | 12.73       | 45.33         | 47.95         | 54.22         |
| 0.01       | 12.66       | 43.74         | 46.11         | 52.48         |

**Table 2.** PSNR of algorithms to Cameraman image with mixed noise

| $\sigma^2$ | Mean filter | Median filter | Wiener filter | New algorithm |
|------------|-------------|---------------|---------------|---------------|
| 0.002      | 12.79       | 50.37         | 55.93         | 55.36         |
| 0.004      | 12.69       | 47.74         | 52.29         | 53.80         |
| 0.006      | 12.65       | 45.94         | 50.00         | 52.90         |
| 0.008      | 12.57       | 44.46         | 47.94         | 51.95         |
| 0.01       | 12.56       | 44.36         | 47.90         | 51.85         |

The results of the tables show that the PSNR of new algorithm is higher than the classical ones.

The experimental results demonstrate that the new image mixed noise removal algorithm based on measuring of medium truth scale can do better in smoothing mixed noise and preserving details than the classical ones do in subjective aspect and objective aspect, which will lead to its practicable and effective applications in mixed noise removal and image restoration.

## 4 Conclusions

This paper presents a new mixed noise removal algorithm based on measuring of medium truth scale. Conclusions about the new algorithm can be drawn as follows.

(1) According to the analysis of the characteristics of Salt-Pepper noise and Gaussian noise, this paper discusses a new image mixed noise removal method which is different from other mathematical methods.

(2) Use the distance ratio function to detect and restore the Salt-Pepper noise corrupting image.

(3) Apply the symmetric-mean filtering to remove the Gaussian noise.

(4) The experimental results show that the new algorithm is more practicable and effective in image mixed noise removing than other classical algorithms.

Further research is to apply the algorithm to the image segmentation and image recognition.

## Acknowledgment

This work was supported by National Basic Research Program of China (973 Program) (No. 2005CB321901), Open Fund of the State Key Laboratory of Software Development Environment, Beihang University, (No.BUAASKLSDE-09KF-03).

## References

1. Hu, X.D., Peng, X., Yao, L.: Study of wavelet domain Gaussian mixture model with median filtering mixed image denoising. *J. Guangzi Xuebao/Acta Photonica Sinica* 36(12), 2381–2385 (2007) (in Chinese)
2. Xu, Y.N., Liu, L.P., Zhao, Y., Jin, C.F., Sun, X.D.: Hopfield neural network-based image restoration with adaptive mixed-norm regularization. *J. Chinese Optics Letters* 7(8), 686–689 (2009) (in Chinese)
3. Lalush, D.S.: Binary encoding of multiplexed images in mixed noise. *J. IEEE Transactions on Medical Imaging*. 27(9), 1323–1332 (2008)
4. Morillas, S., Gregori, V., Hervás, A.: Fuzzy peer groups for reducing mixed Gaussian-impulse noise from color images. *J. IEEE Transactions on Image Processing*. 18(7), 1452–1466 (2009)
5. Chen, D.L., Xue, D.Y., Gao, D.X.: New efficient fuzzy weighted mean filter approach for removal of mixed noise. *J. Xitong Fangzhen Xuebao / Journal of System Simulation* 19(3), 527–530 (2007) (in Chinese)
6. Zhu, W.J., Xiao, X.A.: Propositional Calculus System of Medium Logic(I). *J. Nature*. 8, 315–316 (1985) (in Chinese)
7. Zhu, W.J., Xiao, X.A.: A system of medium axiomatic set theory. *J. Science in China* 11, 1320–1335 (1988)
8. Hong, L., Zhu, W.J., Xiao, X.A.: Measure of Medium Truth Scale and Its Application. *J. Journal of Computer*. 29, 2186–2193 (2006) (in Chinese)
9. Lu, Y.: Research on Image Quality Evaluation of the Digital Video System. Master thesis of An-Hui University (2005) (in Chinese)

**Part IV**

**Bioinformatics and Medical  
Applications**

# Clinical Examples as Non-uniform Learning and Testing Sets

Piotr Augustyniak

AGH University of Science and Technology, 30 Mickiewicza Ave.  
30-059 Krakow, Poland  
august@agh.edu.pl

**Abstract.** Clinical examples are widely used as learning and testing sets for newly proposed artificial intelligence-based classifiers of signals and images in medicine. The results obtained from testing are usually taken as an estimate of the behavior of automatic recognition system in presence of unknown input in the future. This paper investigates and discusses the consequences of the non-uniform representation of the medical knowledge in such clinically-derived experimental sets. Additional challenges come from the nonlinear representation of the patient status in particular parameters' domain and from the uncertainty of the reference provided usually by human experts. The presented solution consists of representation of all available cases in multidimensional diagnostic parameters or patient status spaces. This provides the option for independent linearization of selected dimensions. The recruitment to the learning set is then based on the case-to-case distance as selection criterion. In result, the classifier may be trained and tested in a more suitable way to cope with unpredicted patterns.

## 1 Introduction

In medical research, the size of study cases population is an important, and usually the only considered factor of the result's reliability. However, even an intuitive approach says that two similar cases don't significantly enrich the learning or testing sets. Despite only a little influence the clinical researcher has on the available examples, each paper on clinical data-based research specifies only the cases count and neglects the features distribution therefore silently assuming it is gaussian [7 3].

This observation and lack of justified guidelines for learning sets composition motivated us to the research on the intelligent recruitment. The presented method is proposed as an alternative for the random choice in the recruitment of cases to the learning set from all available medical examples. It is noteworthy that the human education, particularly in medicine, is also based on purposely preselected examples. Unlike the learning set, that determines the volume of competence of the AI classifier, the most natural method of recruitment test set members is the random choice.

The paper is organized as follows: In section 2 two alternative representations of medical cases are presented. The transformations between the representations

and linearization of selected domains are also concerned in that section. Section 3 introduces the definition of the case-to-case distance and two methods of recruitment the cases as learning set members. Section 4 is dedicated to the description of conditions and results of tests of the basic QRS complex types recognition in the electrocardiogram with use of backpropagation neural network and both proposed learning set recruitment methods.

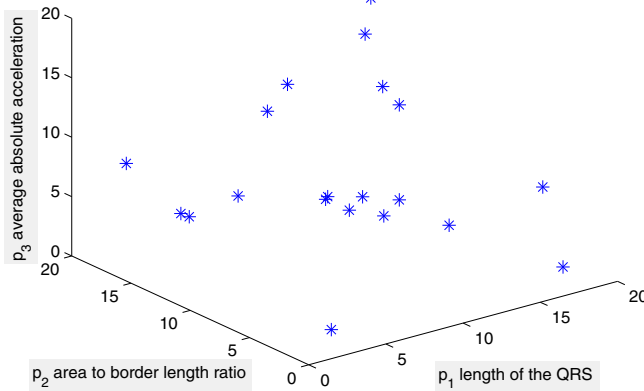
## 2 Management of Cases Representation

### 2.1 Parameter-Domain Representation of Cases

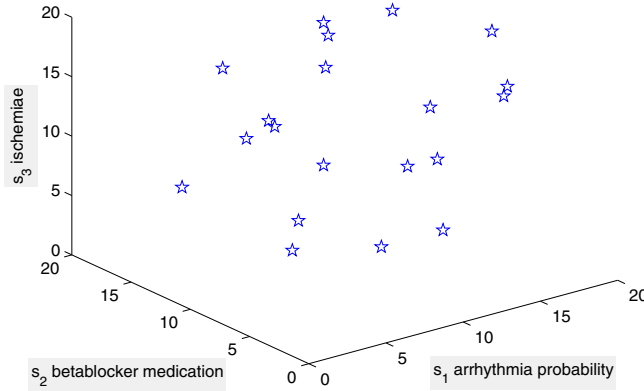
Detailed description of the patient on the cell level is rarely practical in the clinic for the reasons of huge amount of data and lack of the organ’s physiology description. Health records usually provide several organ-specific descriptors and principal global parameters describing the whole organism in aspects representing it as organ’s environment influencing its functionality. Each subject  $S$  can be described by the set of diagnostic parameters  $S_p = \{p_1 \cdots p_N\}$ , considered as the projection of his physiological state on the modality-dependent  $N$ -dimensional state space  $\mathbf{S}^N$ . The projection is limited due to restrictions on the count  $N$  of values available for measurement and inaccurate due to measurement errors  $\epsilon_N$  and additive interferences  $\delta_N$  [8].

### 2.2 Disease-Domain Representation of Cases

The description of the disease  $D$  usually involves a set of symptoms  $D_s = \{s_1 \cdots s_M\}$ , being characteristic patterns of  $M$  selected parameters [9]. Their coincidence is defined as a set of conditions  $C_D\{M\}$ , also called *disease templates* allowing the doctor to make evidence of certain pathology. Comparing to



**Fig. 1.** Parameter-domain representation of cases. Example dimensions are: length of the QRS, area to border length ratio and average absolute acceleration.



**Fig. 2.** Disease-domain representation of cases. Example dimensions are: arrhythmia probability, betablocker medication and ischemiae.

the *parameter-domain representation* being an initial quantitative description of the subject, the *disease-domain representation* is a result of the interpretation process and the final outcome of the diagnostics determining the subject treatment.

### 2.3 Transformation of Cases Representation

The medical diagnosis may be considered as matching of the parameter-domain and the disease-domain case descriptions. The subject  $S \in \mathbf{S}$  is qualified to a certain category defined in the disease-domain space  $\mathbf{D}^M$  and described as having a disease  $D \in \mathbf{D}$  by means of his diagnostic parameters  $S_p$  best matching to recommended and fulfilling the essential criteria of the disease pattern  $D_s$  (eqn. 1). Although the patient may meet the criteria  $C_D\{M\}$  for several diseases  $D_1, D_2, D_3 \dots$ , only up to two, three of them, considered as most important are diagnosed and treated in medical practice.

$$p(D) = |\langle S_p, D_s \rangle| \text{ where } \forall m \in M D_s \subset C_D\{M\} \tag{1}$$

The patient’s status available in his health record in the parameter-domain representation may be transformed to the normalized diseases space in which the probability of each disease is represented independently (2).

$$\mathbf{S}^N \rightarrow \mathbf{D}^M: 0 \leq p(D) = C_D\{M\} \leq 1 \tag{2}$$

This transformation is based on the quantitative measure of correlation between the subject’s record and the *disease template*. The transformation is only partially mutually unambiguous, since - due to the data reduction - guessing the diagnostic parameters of a subject from the disease he or she has is hazardous. The transformation is performed by the human medic during the diagnosis process. If the available  $N$  parameters are not sufficient to separate two pathologies,

the representation of the subject may be completed by the complementary diagnosis yielding  $N-1$  new parameters and interpreted in an iterative way.

## 2.4 Domain Linearization of Cases Representation

Representation of the subject's state in the multidimensional parameter space assumes the independence of any two particular variables. In spite this is not always fulfilled, such representation opens the opportunity for linearization of the dimensions [1], where the nonlinearity considerably influences the distance calculations. The parameter  $p_n$  whose values have to be piecewise expanded or compressed in order to correctly represent the differences between particular diseases is transformed accordingly to (3):

$$p'_n = f(p_n) \quad (3)$$

where  $f$  is a nonlinear projection of the dimension  $p$  to  $p'$ . The projection  $f$  is a piecewise continuous function defined in all the domain of the parameter  $p_n$  in a heuristic way in order to provide equal separation of the normal and abnormal cases regardless the parameter's value.

## 3 Distance of the Case-Representation Space

### 3.1 Definition of the Case-to-Case Distance

In order to consider distance-based similarity of case representations, the notion of distance has to be defined in parameter-domain and diseases-domain spaces. Provided all dimensions of the space are linear, the following Cartesian definition of the distance is applicable (4):

$$d(p_N, p_K) = \sqrt{w_1 (d_1^N - d_1^K)^2 + \dots + w_m (d_m^N - d_m^K)^2} \quad (4)$$

where  $w_m$  is the  $m$ -th weighting coefficient for each dimension in the disease state space.

### 3.2 Distance-Based Case Recruitment

Aiming at the generation of possible rich and representative learning set from a given set of available cases, the selection of appropriate cases may follow one of the following paradigm:

- maximum hiperspace volume, or
- equidistant hiperspace support.

First approach tends to *maximize the volume* of the competence hiperspace by selecting the cases most distant in the space as learning set members. This can be achieved by calculating first the gravity center of the cloud of all available

medical cases. First candidate is recruited as the most peripheral case. Next candidates are cases most distant from all of the previously selected, therefore the learning set consists of most atypical cases. The procedure ends after having recruited a given number of cases, or when cases exceeding the given distance are no longer available.

Second approach assumes the *equal distance* between the learning set samples. The procedure starts with the calculation of the distance between any two cases in the space. If the distance histogram is unimodal, the distance range is determined around its maximum population bin. First candidate is then recruited at random from cases belonging to the most typical distance bin. The recruitment of next candidates is based on how their distance to the already recruited case matches with the mid-range value. The procedure ends when the only remaining cases show out-of-range distance.

## 4 Test Conditions and Results

Both presented recruitment methods were implemented in the Matlab environment. Based on objects description consisting of up to 25 parameters each, they are capable to select from the initial population a subset complying with given distance or population criteria. Despite the method is designed to provide purposely selected learning sets for a wide class of artificial intelligence algorithms, we tested it on a three-layers backpropagation neural network [6], [10], [5], [12] applied to heartbeat types (QRS) classification in the electrocardiogram (ECG). The ECG signal originated from the MIT-BIH Arrhythmia Database [4] and was sampled with the frequency of 360 Hz.

Each QRS section was represented in normalized parameter- and disease-spaces. The versors of the parameter space were:

- length of the QRS,
- area to border length ratio,
- average absolute acceleration.

The versors of the disease space were:

- arrhythmia probability,
- betablocker medication,
- ischaemiae (insufficient oxygenation, ST segment changes).

Each dimension was quantized to 20 levels, therefore the input layer contains 60 neurons. The middle layer is composed of 16 neurons, and the output layer contains 4 neurons, accordingly to the recognition of four basic QRS morphologies: normal, supraventricular, ventricular and undetermined.

From 1500 cases of heartbeats available from the database with medical annotations, 150 cases were randomly selected as the test set, and other 150 cases were recruited accordingly to the presented methods as the learning set. For the purpose of reference, the random recruitment was also used as an option.



The results of classification accuracy for the parameter-space heartbeat representation are displayed in tab. 1

The results of classification accuracy for the disease-space heartbeat representation are displayed in tab. 2.

**Table 1.** Parameter-space heartbeat representation. Percentage of correct heartbeat classification for the same test set and different recruitment method for learning set cases.

| learning set recruitment method | random | maximum volume | equidistant support |
|---------------------------------|--------|----------------|---------------------|
| normal                          | 93     | 97             | 98                  |
| supraventricular                | 63     | 93             | 95                  |
| ventricular                     | 87     | 98             | 98                  |
| undetermined                    | 51     | 78             | 71                  |

**Table 2.** Disease-space heartbeat representation. Percentage of correct heartbeat classification for the same test set and different recruitment method for learning set cases.

| learning set recruitment method | random | maximum volume | equidistant support |
|---------------------------------|--------|----------------|---------------------|
| normal                          | 78     | 85             | 88                  |
| supraventricular                | 60     | 90             | 91                  |
| ventricular                     | 81     | 85             | 91                  |
| undetermined                    | 53     | 77             | 74                  |

## 5 Discussion

The artificially prepared learning set (in case of both recruitment methods and independently in both representation domains) led to a considerably better result of the network learning, expressed by a better recognition result achieved in the test phase with use of the same randomly selected test set.

When the method *maximizing the volume* of competence space was used, the recognition of undetermined beats was slightly better comparing to the competitors. This was caused by the representation of more atypical beats within the learning set. The features of these beats lie far from the gravity center of the cases cloud in the parameter space.

On the contrary, the use of *equidistant hyperspace support* refines the allotment of beats into basic categories (normal/supraventricular/ventricular) at a price of few more atypical beats erroneously falling into the 'undetermined' category. Such behavior is most expected in real electrocardiogram interpreters and the proposed regularization of the non-uniform representation of the medical knowledge has been proven as the efficient method for ameliorating the learning set adequacy.

When the disease-space heartbeat representation is used, the percentage of correct heartbeat classification drops dramatically for two reasons:

- disease-domain representation is determined based on the electrocardiogram context wider than a single heartbeat,

- transformation of cases representation simplifies the cases description thus for the determination of the heartbeat types, the disease-domain is less representative than the parameter-domain.

In fact the determination of heartbeat types based on the disease-domain representation is a reversal of the typical diagnostic order. The regular interpretation first calculates the parameters, then based on parameter values determines the beat types, which in turn and in the context of neighboring results are used for assignment the disease-domain representation [11].

The presented investigation supported by the example of application to the problem of heartbeat classification, well known in electrocardiology, demonstrates that the recruitment of learning set members determines the quality of AI-based recognition systems. The selection method should consider:

- possibly diverse examples spanning the hyperspace of competence which volume is maximal,
- possibly regular distribution of the samples of medical knowledge along the particular dimensions of that hyperspace.

## Acknowledgment

Scientific work supported by the Polish State Committee for Scientific Research resources in years 2009-2012 as a research project No. N N518 426736.

## References

1. Aldroubi, A., Feichtinger, H.: Exact Iterative Reconstruction Algorithm for Multivariate Irregularly Sampled Functions in Spline-like Spaces: the  $L^p$  Theory. Proc. Amer. Math. Soc. 126(9), 2677–2686 (1998)
2. Augustyniak, P.: Automatic Understanding of ECG Signal. In: Kopotek, A., Wierzhon, S.T., Trojanowski, K. (eds.) Intelligent Information Processing and Web Mining, pp. 591–597. Springer, Heidelberg (2005)
3. Haussler, D.: Quantifying Inductive Bias: AI Learning Algorithms and Valiant's Learning Framework. Artificial Intelligence 36, 177–221 (1988)
4. Moody, G.B., Mark, R.G.: The MIT-BIH Arrhythmia Database on CD-ROM and Software for Use with it. In: Computers in Cardiology 1990, pp. 185–188 (1990)
5. Osowski, S.: Neural Networks for Information Processing. WUT Publishing House, Warsaw (2000) (in Polish)
6. Rutkowski, L., Tadeusiewicz, R. (eds.): Neural Networks and Soft Computing. Polish Neural Network Society (2000)
7. Stanis, A.: Accessible Course of the Statistics with STATISTICA PL and Examples from Medicine. StatSoft Poland, Krakow (2006) (in Polish)
8. Straszecka, E., Straszecka, J.: Distance Based Classifiers and their Use to Analysis of Data Concerned Acute Coronary Syndromes. Image Processing & Communications 9(3-4), 53–69 (2003)
9. Straszecka, E., Straszecka, J.: Interpretation of Medical Symptoms Using Fuzzy Focal Element. In: Kurzynski, M., et al. (eds.) Computer Recognition Systems. Springer, Heidelberg (2005)

10. Tadeusiewicz, R.: *Neural Networks*. RM Academic Publishing House, Warsaw (1993) (in Polish)
11. Tadeusiewicz, R., Augustyniak, P.: Information Flow and Data Reduction in the ECG Interpretation Process. In: *IEEE 27 Annual EMBS Conf.*, paper 88 (2005)
12. Tadeusiewicz, R., Ogiela, L.: Selected cognitive categorization systems. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) *ICAISC 2008*. LNCS (LNAI), vol. 5097, pp. 1127–1136. Springer, Heidelberg (2008)

# Identifying the Borders of the Upper and Lower Metacarpophalangeal Joint Surfaces on Hand Radiographs

Andrzej Bielecki<sup>1</sup>, Mariusz Korkosz<sup>2</sup>, Wadim Wojciechowski<sup>3</sup>,  
and Bartosz Zieliński<sup>1</sup>

<sup>1</sup> Institute of Computer Science, Jagiellonian University,  
Lojasiewicza 6, 30-348 Cracow, Poland  
bielecki@ii.uj.edu.pl, bartosz.zielinski@uj.edu.pl

<sup>2</sup> Division of Rheumatology, Departement of Internal Medicine and Gerontology,  
Jagiellonian University Hospital,  
Śniadeckich 10, 31-531 Cracow, Poland  
mariuszk@mp.pl

<sup>3</sup> Department of Radiology, Jagiellonian University Hospital,  
Kopernika 19, 31-531 Cracow, Poland  
wadim@mp.pl

**Abstract.** In this paper, the next stage of our studies concerning the computer analysis of hand X-ray digital images is described. An algorithm identifying the borders of the upper and lower joint surfaces on hand radiographs is proposed. It is based on local segmentation and profile plots analysis. The described algorithm achieved high efficiency - mean distance distribution signature for complete borders was equal to 0.118mm. Therefore, it can be applied both to trace development of hand joints diseases and analysis of bone contour shapes using syntactic methods.

## 1 Introduction

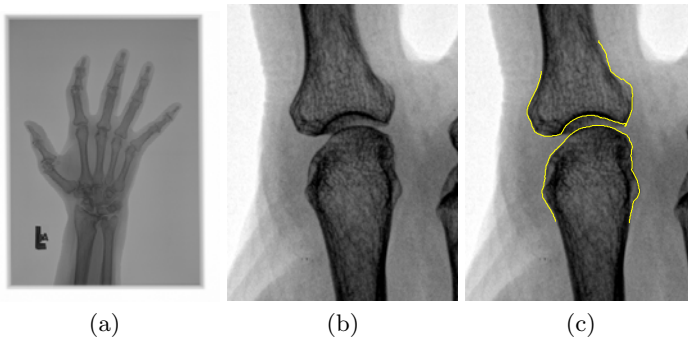
Hand radiographs are an important source of clinical information in the case of inflammatory and non-inflammatory diseases [6]. To give a diagnosis, an X-ray is taken of the patient's hand and symmetric metacarpophalangeal joint spaces and interphalangeal joint spaces are analyzed. Detecting and tracing pathological changes, like joint space narrowing, as soon as possible is a crucial point in medical diagnosis. Hence, the difference between metacarpophalangeal joints width equals to about 0.5mm in the two following X-ray pictures is significant and supports important information for the estimation of therapy efficiency. However, such small changes are difficult to detect if an X-ray picture is examined by a human expert. Therefore, the possibility of such analysis performed by a computer system, which enables not only the detection of tiny changes in X-ray picture, but also interpret them from a medical point of view, is a key point for diagnosis support. Studies concerning the possibilities of such systems

implementation have been the topic of numerous publications [4,5,7] (see more in [2]).

This paper is a continuation of the studies described in [1,2,8,9]. In the previous papers, picture preprocessing and joint location algorithms, efficient in around 97% of cases, were presented (see [2,9]). In this paper, the successive step of the development of the computer system for hand diseases diagnoses support is presented. The algorithm identifying the borders of the upper and lower joint surfaces on hand radiographs has been implemented. Obtained borders will be utilized in the implementation of the bones shape description and interpretation algorithm presented in [1].

## 2 Identifying the Borders of the Upper and Lower Surfaces

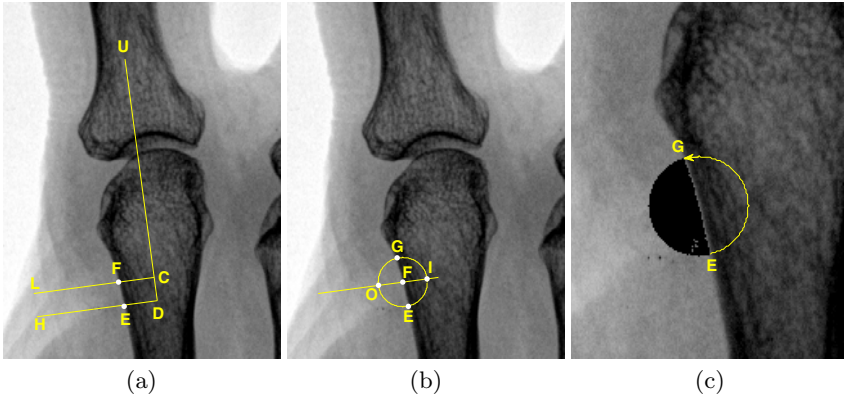
The algorithm for identifying the borders of the upper and lower surfaces requires the knowledge that all analyzed metacarpophalangeal joints are condyloid (elipsoid) joints. Moreover, every joint is placed between two bones called upper surface and lower surface. Due to the specific character of these joint types, the borders of the upper and lower surfaces should be considered differently.



**Fig. 1.** Input image (a) joint's region of interest (b) and ideal upper and lower surfaces borders (c)

It should be stressed that the upper surface of every considered joint contains an indentation, resulting in the special location of its border, running through the bottom of the area, which contains the maximum valued pixels (upper line in Fig. 1c). Unlike the upper surface, the bottom bone surface does not contain an indentation, which results in the border running through the edge of the bone (lower line in Fig. 1c).

The main idea of the proposed strategy is to compute the path running through the borders of both surfaces, like the bright lines in Fig. 1c, using two points - one from the upper and one from the lower surface. Those two points can be positioned manually by the operator or can be located automatically, using the joint space location method described in [2].



**Fig. 2.** Initially analyzed line segments  $DH$  and  $CL$  (a), circle containing inner and outer pixels of the surface (b) and proceeding counter-clockwise circle points analysis, starting from border point  $E$  (c)

### 2.1 Initial Border Points Location

Let  $U$  and  $D$  be points located respectively for the upper and lower surface and let  $DU$  be the line segment created between them. Let region of interest (**ROI**) be part of the image in Fig. 1a which contains line segment  $DU$  running through the corresponding parts of the joint surfaces (see Fig. 1b). Moreover, let  $DH$  and  $CL$  be two initial line segments (**ILS**) perpendicular to line segment  $DU$ , such that  $C$  is a point located on the line segment  $DU$  in the neighborhood of point  $D$  (see Fig. 2a).

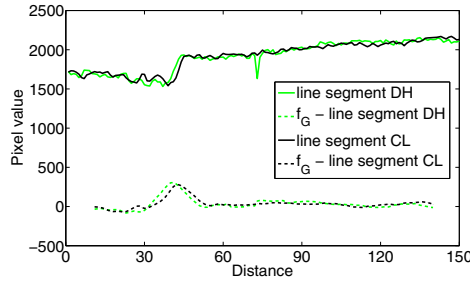
To compute border points lying on the line segments  $DH$  and  $CL$ , their profile plots have to be analyzed using special function  $f_G$ , defined as follows:

$$f_G(x_n) = \text{mean}\{f(x_{n+i})|i = \{1..k\}\} - \text{mean}\{f(x_{n-i})|i = \{1..k\}\}, \quad (1)$$

where  $f$  is profile plot and  $\text{mean}\{f(x_{n+i})|i = \{1..k\}\}$  is average value of the  $k$  successive arguments. The initial border point (**IBP**) is assigned to the argument, for which function  $f_G$  reaches its global maximum. Accurate location of IBP is certain due to the fact that the values of the pixels corresponding to diaphysis (mid section of the bone) strongly contrast with the neighbouring pixels, corresponding to the background. Profile plots and function  $f_G$  for line segments  $DH$  and  $CL$  are presented in Fig. 3. Points corresponding to IBP are marked in Fig. 2a as  $E$  and  $F$ .

### 2.2 Local Segmentation

Initial border points  $F$  and  $E$ , obtained in the previous step are used to create a circle with a center in point  $F$  and radius equal to the distance from  $F$  to  $E$ . That kind of circle surrounds two kinds of pixels, corresponding to surface and background, divided by a border running through the points  $F$  and  $E$  (Fig. 2b).



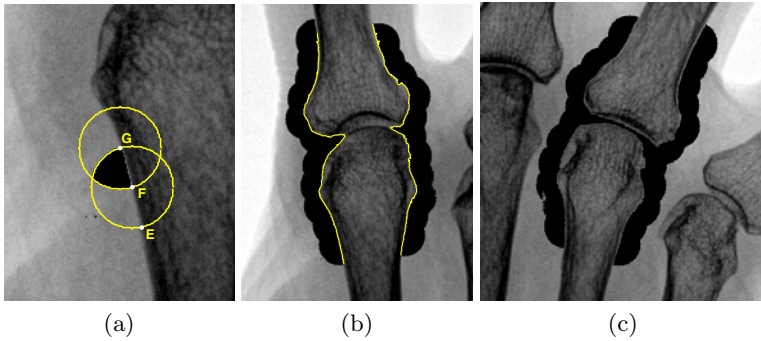
**Fig. 3.** Profile plots of line segments from Fig 2a and analogous functions  $f_G$

Moreover, this circle intersects with ILS in two points considered as inner intersect point (**IIP**) and outer intersect point (**OIP**) - see point  $I$  and  $O$  in Fig 2b. To segment the region surrounded by the given circle, the algorithm starts by computing mean local background value (**MLBV**), which is an average value of pixels from the neighborhood of OIP. Due to the fact that OIP is located out of the surface, calculated value corresponds to the value of the local background pixels. Then, algorithm uses MLBV to analyze successive pixels lying inside the circle, starting from OIP and proceeding with the four neighbours of OIP, then with the neighbors of OIP's neighbours and so on. Let *MaxBackgroundDeviation* be a constant value which corresponds to maximal deviation of the background pixels values from the *MLBV*. If the value of the analyzed pixel is greater than  $MLBV - MaxBackgroundDeviation$ , then this pixel is marked as a background pixel and its neighbors are analyzed as well. Otherwise, the pixel is marked as surface and its neighbors are not analyzed.

As a result of local segmentation, the algorithm returns two consistent parts of the circle, divided by the border (see Fig 2c).

This local circle segmentation (**LCS**) is now used to compute the next point of local segmentation. It is made possible by the fact that the circle runs through two points of the border. One of those points was already set and used to create the circle (point  $E$  in Fig 2c). The second border point is unknown (point  $G$  in Fig 2c), but can be computed by analyzing successive points from the circle, starting from point  $E$  and running counter-clockwise as long as the actual point belongs to a surface (see Fig 2c). The second border point is set as the last point which meets this condition. Point  $G$  is then used as the center point of the successive circle and local segmentation is performed once more. However, in contrast to the first local segmentation, *MLBV* is calculated as the average value of background pixels obtained from the last segmentation and lying inside the successive circle (black pixels in Fig 4a).

The algorithm operates as long as the second border point is inside the ROI and at the end can produce two kinds of results. If the upper and lower surface do not overlap one another, the algorithm produces two borders - one for the upper and one for the lower surface (Fig 4c). In this case, the lower surface border is the final border, however the upper surface border still has to be analyzed, due



**Fig. 4.** Successive circles applied in local segmentation (a) and result of local segmentation for surfaces overlapping one another (b) and separated surfaces (c)

to the fact that sometimes it runs through the rear edge of the bone, instead of running through the bottom of the area, which contains the maximum valued pixels. When surfaces overlap one another, the algorithm produces left and right borders for both surfaces, although it does not separate one surface from another (Fig 4b).

### 2.3 Surface Disconnection

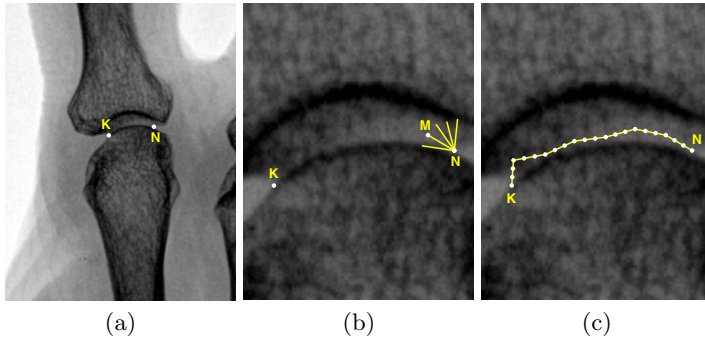
This step of the algorithm applies only to the ROI with overlapping surfaces.

For both the left and right border obtained in the previous stage, the algorithm searches for two points which have the shortest distance to line segment DU. These points correspond to exact joint space locations (**EJSL**) - see points K and N in Fig 5a.

The next issue is to connect the left and right EJSL by curve, which separates two surfaces and does not intersect with any of their borders. Such a curve, called an inner path (**IP**) is created by outgoing points from point N to point K in Fig 5a. Let  $\{P_i\}_{i=1}^n$  be a set of successive IP points, such that  $P_1 = N$  and  $P_n = K$ . Each given point  $P_i$  is used to calculate successive point  $P_{i+1}$ , by analyzing five line segments  $\{P_i P_{i+1}^\alpha\}_{\alpha=\{10,20,30,40,50\}}$ , where  $\alpha$  is the angle between line segments  $P_i K$  and  $P_i P_{i+1}^\alpha$ . End point of the line segment with the highest mean value of underlying pixels is then chosen as  $P_{i+1}$  (point M of line segment NM in Fig 5b). The algorithm stops operating if analyzed point  $P_i$  is in the neighborhood of point K, which eventually leads to obtain the inner path (see Fig 5c).

In the final step, IP is analyzed to obtain the final borders of the upper and lower surfaces. However, as has been mentioned above, it is probable that the border generated by the local segmentation for the upper surface is not accurate. Due to this fact, IP is extended to points from the left and right border, which lie in the neighbourhood of EJSL. Extended inner path (**EIP**), established to contain all required inner points, is shown in Fig 6a. Then, the algorithm applies the modification of the method proposed and described by authors in [2]. The





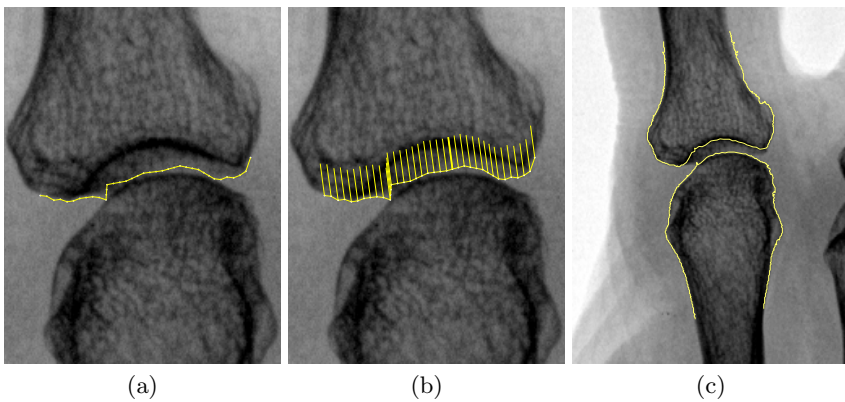
**Fig. 5.** Exact joint space locations (a), line segments analyzed in successive steps (b) and inner path (c)

main idea of the proposed strategy is to create a profile plot for each line segment, parallel to line segment  $DU$ , with the lower endpoint located in the corresponding inner point - see Fig. 6b. For each profile plot achieved in this manner, the analysis has to be conducted.

Denote by  $x_{min}$ , the argument for which the function  $f$  reaches its global minimum. Let  $CDiff$  be a fixed parameter, which refers to half of the average difference between pixels located on the indentation and pixels located on a bone without indentation. The argument corresponding to the upper border is given by the formula:

$$x_{up} = x_i : f(x_i) - f(x_{min}) > CDiff \wedge \bigwedge_{min < j < i} f(x_j) - f(x_{min}) < CDiff \quad (2)$$

This action enables us gaining of the border points of the upper surface using knowledge about inner point locations. Obtained border points connected with



**Fig. 6.** Extended inner path (a), line segments parallel to line segment  $DU$  (b) and complete border of the upper and lower surfaces (c)

border points attained previously in the local segmentation result in a complete border of the upper surface, as in Fig 6t.

Upper surface border is then used to compute the border of the lower surface, by creating profile plots for each line segment parallel to line segment  $DU$ , in such a way that the upper endpoint is placed in the corresponding point of the upper surface border.

Denote by  $x_{min}$ , the argument for which the derivative of function  $f$  reaches its local maximum. Let  $ADiff$  be a fixed parameter, which refers to the maximal difference between successive arguments  $x_{min}^{prev}$  and  $x_{min}$ . The local minimum from the interval  $[x_{min}^{prev} - ADiff, x_{min}^{prev} + ADiff]$  is the argument corresponding to the lower border. This final action produces unknown points of the lower surface border which can be joined with border points obtained previously by local segmentation, resulting in a complete border of the lower surface, like lower line in Fig 6c.

### 3 Experiments

In total, 40 joints were analyzed in 10 images (for 5 subjects) included in the test set, acquired through the offices of the University Hospital in Cracow, Poland. The tested images presented bones with small pathological changes, rheumatoid arthritis being the most prevalent cause of inflammatory disorder. Each analyzed image was a 12-bit grayscale image taken in anterior-posterior position,  $2920 \times 2320$  pixels resolution, given in a DICOM file format.

An experienced radiologist examined all joints from the test set and manually marked borders using tablet Dell XPS 2. The analyzed images were presented with the help of a graphical user interface and the radiologist could draw or correct the contours as well as zoom in or zoom out of the image to see more or less detail. The borders marked by the expert were used as the ground truth to examine borders extracted automatically. The efficiency of the algorithm was examined with the distance distribution signature proposed in [3].

Denote by  $B_I$  and  $B_A$ , ground truth border and extracted border, respectively. A distance distribution signature from set  $B_I$  to set  $B_A$ , denoted by  $\mathcal{D}_{B_I}^{B_A}$ , is a discrete function whose distribution characterizes the discrepancy, measured in distance from  $B_I$  to  $B_A$ . Define the distance from arbitrary point  $x$  in set  $B_A$  to set  $B_I$  as the minimum absolute distance from  $x$  to all the points in  $B_I$ ,  $d(x, B_I) = \min\{d_E(x, y) | y \in B_I\}$ , where  $d_E(x, y)$  denotes the Euclidean distance between points  $x$  and  $y$ . Distance distribution signature is given by the formula  $\mathcal{D}_{B_I}^{B_A} = \frac{1}{|B_A|} \sum_{x \in B_A} d(x, B_I)$ .

Mean distance distribution signature for complete borders was  $0.118mm$  (same for lower and upper borders) and its standard deviation was  $0.057mm$  ( $0.046mm$  for upper borders and  $0.067mm$  for lower borders). The minimal and maximal distance distribution signature in test set were equal to  $0.055mm$  and  $0.3mm$  (from  $0.064mm$  to  $0.3mm$  for upper borders and from  $0.055mm$  to  $0.28mm$  for lower borders), respectively. Mean  $\max_{x \in B_A} \{d(x, B_I)\}$  was also computed and was equal to  $0.634mm$ .

## 4 Concluding Remarks

The described algorithm achieved high efficiency - mean distance distribution signature was equal to  $0.118mm$ . In comparison with algorithms described in literature [4], where average accuracy of the obtained borders was about  $1.2mm$ , the presented one turned out to be more accurate - the achieved accuracy corresponding to  $0.4mm$ . This means that the algorithm is able to detect the mentioned changes of  $0.5mm$  (see discussion in section [1]). This allows us to trace the development of both degenerative and inflammatory diseases in metacarpophalangeal and interphalangeal joint regions. Furthermore, such precisely obtained borders will enable exact analysis of the bone contour shapes using syntactic methods ([2]), which will allow erosion detection in bones. Such analysis is crucial for medical diagnosis support, enabling discrimination between degenerative changes and inflammatory changes.

## References

1. Bielecka, M., Skomorowski, M., Zieliński, B.: A fuzzy shape descriptor and inference by fuzzy relaxation with application to description of bones contours at hand radiographs. In: Kolehmainen, M., et al. (eds.) ICANNGA 2009. LNCS, vol. 5495, pp. 469–478. Springer, Heidelberg (2009)
2. Bielecki, A., Korkosz, M., Zieliński, B.: Hand radiographs preprocessing, image representation in the finger regions and joint space width measurements for image interpretation. *Pattern Recognition* 41(12), 3786–3798 (2008)
3. Huang, Q., Dom, B.: Quantitative methods of evaluating image segmentation. In: International Conference on Image Processing, vol. 3, pp. 53–56 (1995)
4. Kauffman, J.A., Slump, C.H., Bernelot Moens, H.J.: Segmentation of hand radiographs by using multi-level connected active appearance models. In: Proceedings of the SPIE, vol. 5747, pp. 1571–1581 (2005)
5. Sharp, J., Gardner, J., Bennett, E.: Computer-based methods for measuring joint space and estimating erosion volume in the finger and wrist joints of patients with rheumatoid arthritis. *Arthritis & Rheumatism* 43(6), 1378–1386 (2000)
6. Staniszevska-Varga, J., Szymańska-Jagiello, W., Luft, S., Korkosz, M.: Rheumatic diseases atlas. Practical Medicine Publishing House, Cracow (2003) (in Polish)
7. Tadeusiewicz, R., Ogiela, M.: Picture languages in automatic radiological palm interpretation. *International Journal of Applied Mathematics and Computer Science* 15(2), 305–312 (2005)
8. Zieliński, B.: A Fully-Automated Algorithm Dedicated to Computing Metacarpophalangeal and Interphalangeal Joint Cavity Widths. *Schedae Informaticae* 16, 47–67 (2007)
9. Zieliński, B.: Hand radiograph analysis and joint space location improvement for image interpretation. *Schedae Informaticae* 17/18, 45–61 (2009)

# Decision Tree Approach to Rules Extraction for Human Gait Analysis

Marcin Derlatka<sup>1</sup> and Mikhail Ihnatouski<sup>2</sup>

<sup>1</sup> Faculty of Mechanical Engineering Bialystok Technical University,  
Wiejska Street 45C, 15-351 Bialystok, Poland  
mder@pb.edu.pl

<sup>2</sup> The Research Center of Resources Saving Problems  
National Academy of Sciences of Belarus,  
Tizenhauza sq., 7,230023 Grodno, Belarus  
mii\_by@mail.ru

**Abstract.** The article presents the application of decision tree techniques to exploring barometric information obtained with instruments measuring the pressure of the human plantar onto contact surface while walking. The investigation was carried out on a group of 28 typical subjects as well as the subjects affected by Pes Planovalgus and Cerebral Palsy. The decision tree has been inducted by means of the vector of 255 values describing single stride with 51 samples per each of five zones of the human foot. The classification made by the resultant decision tree was correct for more than 94% strides. This allows to point the parameters which are the best discriminators between the investigated types of human gait.

**Keywords:** human gait, barometric system, decision tree.

## 1 Introduction

Data mining is the set of methods which enables to manage a huge and multidimensional set of measured data. Data mining provides for fast and efficient analyzing of the data and finding new, sometimes unexpected, connections between various parameters [8].

Biomedical engineering is a very particular and important field of data mining application. First of all, biomedical engineering helps in improving the quality of human life. Second, biomedical data possess special features such as high inter- and intrasubject variability, high nonlinear dependency between some parameters, multidimensionality etc., which cause difficulties in analyzing cases of particular subjects with conventional methods [3].

One of the important problems in biomedical engineering is an automatic instrumented human gait analysis. Gait is a basic human activity. It enables us to relocate our body. Gait is also a very complex human activity. It is described by enormous number of parameters which include:

- kinematics;
- kinetics;
- anthropometrics;
- electromyographics;
- others.

The measurement of some of those parameters is necessary to perform the quantitative human gait assessment. The assessment of human gait is a very important task to for instance, evaluate the level of disease and quantifying the effects of rehabilitation process or surgical intervention. The success of taking the classical approach to human gait by clinician is strongly limited by his ability. Good clinician should be familiar with the technical and medical side of the investigation. He should also be able to handle large sets of data and to make proper expertise based on his knowledge and experience. Nowadays, the methods of automatic human gait analysis are very popular, because they break the limitations of manual evaluation of the data concerning gait [2, 3, 4, 6, 7, 16, 17]. The artificial intelligence methods for automatic gait analysis are as follows: neural networks [17], fuzzy logics [18] and others [4, 12].

One of the most promising techniques of data mining is decision tree. Decision trees enable to extract the knowledge hidden in the data and presenting it in a very vivid way. They provide very simple conditions in the tree nodes and lead to conclusion (class) on the lowest level of the tree. It is very important that the results are easy for interpretation and could be used by the staff with neither mathematical nor engineering background. This makes decision trees useful tool for clinical applications. One of the most interesting properties of decision trees is no apriori assumptions. Moreover, to work properly, the decision trees do not need as much data as neural networks. It is especially important in biomedical applications, where the number of subjects is very limited. Of course, a bigger set of data give more accurate and more trustworthy results. Decision trees have already been successfully used in the human gait analysis in clinical applications [1, 11]. However, in recent papers the main destiny of decision trees is a classification task.

The first step in the common procedure of automatic human gait analysis is based on one of the following methods:

1. creating new diagnostic parameters describing the phenomena of human gait in a synthetic way, which are based on:
  - (a) charts of time-dependant measured variables [10, 13] or,
  - (b) new feature spaces resulting from the reduction of the input vector's dimensionality by means of some mathematical transformations [4, 15, 17],
2. choosing some values of some selected parameters describing events in human gait (like maximum of the limb-loading phase for vertical components of the ground reaction force or an ankle angle in the initial contact phase) [5, 14];

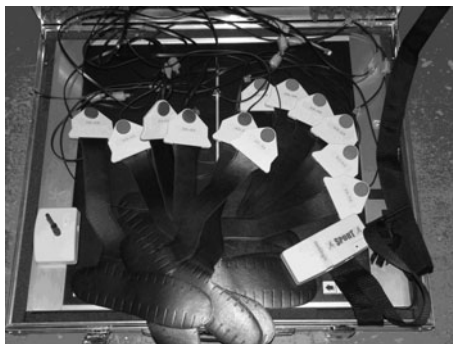
All methods presented above are somewhat controversial. All of them involve a significant reduction of information in the measured data. Moreover, method

1b gives the parameters which have no physical interpretation. The most suitable method should be based on all real data but should select only the most important information for the investigated case.

This paper describes the application of decision tree to the analysis of biomechanical signals from a barometric measuring system. The main aim of the analysis of those signals is to demonstrate the possibility of the diagnosis of diseases, which utilizes recorded signals describing the gait of the investigated subject by means of decision tree. The correct diagnosis is very important in choosing appropriate methods for gait improvement. However, in this paper the inducted decision tree is used not only for classification but also for pointing the main parameters which are the best discriminators between the investigated types of human gait.

## 2 Barometric Measuring System

The foot pressure measuring system is one of the most common devices for human gait analysis. Its main advantages are: the possibility to record sequence of strides and easiness in use. The devices of this type are often used in foot pathologies. During the investigation the barometric insole of an appropriate size is put into the subject's shoes (Fig 1). The insoles consist of many barometric sensors, which allow to record the distribution of pressure of the human plantar onto the contact surface while walking. The subjects walk along a pathway in a comfortable self-determined way. Thus, for the single subject many strides of human gait have been recorded (tab. 1). The acquisition of data has been made with frequency of 150 Hz.



**Fig. 1.** The set of barometric insoles

During cycle  $T$  of measuring plantar pressure  $P^t$  the system may be found at any moment  $t$  ( $t=1, 2, \dots, T$ ) in one of the states characterized by a set of instantaneous pressure values  $p_i^t$  ( $P^t = p_1^t, p_2^t, \dots, p_N^t$ ) transmitted from the insole sensors  $N$  and characterized by index  $i=1, 2, \dots, N$ , via direct measurements. The

set of instantaneous values  $p_i^t$  is a quantitative parameter that depends on the subject’s mass ( $m$ ) and the real contact area of both feet with the surface ( $s$ ):

$$P^t \propto f(m, s, t) \tag{1}$$

It is necessary to group the insole sensors into zones ( $Z$ ) to simplify and make the clinical gait analysis possible. The set of sensors of the insole has been divided into five anatomic zones, as it is presented in Fig 2 [9]. Zone 1 is equivalent to toes, 2 metatarsal bones, 3 cuboidal bone, 4 navicular bone and 5 is heel bone.

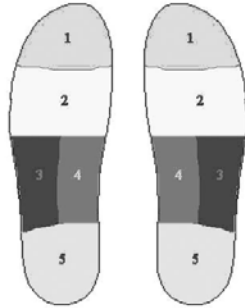


Fig. 2. The sensors insole divided into anatomic zones

### 3 Material and Method

In the paper the authors decided to build an input vector which consists of all the measured data. The measured data have been normalized in time because each stride could be done with different velocity. All data has been interpolated by 51 samples describing one stride for each zone separately (every 2% of the gait stride). The first sample presents the value of the initial contact of the investigated leg, while the fifty-first describes the next initial contact of the same leg. The measured values depend on the weight of the subject, so they have been normalized by dividing the total foot pressure for the investigated lower limb. Finally, we obtained a vector  $V$  of 255 numbers. The numbers describe one stride of the investigated subject.

$$V = [Z1S1 \quad Z1S2 \quad Z1S3 \dots \quad ZiSj \dots \quad Z5S51] \tag{2}$$

where:  $ZiSj$  - denotes a normalized value of foot pressure in  $i$ -th zone during the gait described by  $j$ -th sample.

It is important to note that presenting the vector  $V$  with all samples from all zones at the input of the classifier should highlight which values are crucial in the analysis of human gait in the cases selected for the investigation. The division of the investigated subjects into classes was made by decision tree based on the CART (Classification And Regression Trees) algorithm. A 10-fold cross validation was used to prevent from over fitting of the decision tree.

**Table 1.** Recorded data

|                    | <i>Typical</i> | <i>PesPlanovalgus</i> | <i>CerebralPalsy</i> |
|--------------------|----------------|-----------------------|----------------------|
| number of subjects | 10             | 10                    | 8                    |
| number of strides  | 77             | 94                    | 94                   |

The measurements were made in The Research Center of Resources Saving Problems of the National Academy of Sciences of Belarus in Grodno on a group of 28 subjects by means of barometric system. The barometric insole of appropriate size was put into the subject's shoes (Fig 1). The obtained data contain pressures both typical and pathological gaits. They represent the following types of gait:

- typical: subjects who have not experienced injuries or abnormalities affecting their gait;
- Pes Planovalgus (PP);
- Cerebral Palsy (CP).

It is important to note that the subjects from the second and the third group were affected by diseases at different levels.

Pes Planovalgus is one of the most frequently appearing foot disease. The frequency of this pathology is estimated at the level ranging from a few to several dozen percent of primary schoolchildren, depending on references. In this case the foot has a small, if any, longitudinal arch while loading. It is a result of the muscular-ligament system failure. Moreover, in pes planovalgus the heel bone is in the pronation position which results from the foot being more flattened. The results of pes planovalgus are foot deformation and pain.

Cerebral Palsy is the most common neurologically based disorder. Symptoms of the Cerebral Palsy are spasticity and difficulties in coordination in the musculoskeletal system, caused by a brain damage around the time of birth. People with Cerebral Palsy have difficulties in bipedal locomotion beginning from small problems to not walking at all, depending on the level of disorder. One of the characteristic traits of the moving pattern of the people affected by Cerebral Palsy is starting the initial contact phase with toes.

## 4 Results

The result of using the CART algorithm with the 10-fold cross-validation was a decision tree with 11 nodes (6 conclusion nodes) (fig. 3). The resultant decision tree presented rules extracted from the data. It is really easy to present rules in the 'if-then' algorithm:

- \* if ( $Z3S2 \leq 0.001034$ ) and ( $Z5S30 \leq 0.044715$ ) then Cerebral Palsy
- \* if ( $Z3S2 \leq 0.001034$ ) and ( $Z5S30 > 0.044715$ ) or  
( $Z3S2 > 0.001034$ ) and ( $Z1S30 > 0.198468$ )  
and ( $Z2S20 \leq 1.632567$ ) and ( $Z1S34 \leq 1.061607$ ) then Pes Planovalgus
- \* otherwise Typical.



The result of the classification of the barometric data based on the inducted decision tree is presented in table 2. The classification accuracy is really high (more then 94%). However, we should remember that the classification accuracy depends on the classification task while comparing with the results of other authors who reported application the data mining methods in the human gait analysis [1] (where classification accuracy was equal 81%). It is a very important to note that the authors of this paper have not found any reports of the automatic human gait analysis approach based on the baropodometric system. So, there is no possibility to compare our results to other authors working on the same type of data.

In our approach the highest percentage of correct classification concerns for the subjects affected by Pes Planovalgus disease. The decision tree produces much worse results for the remaining two groups (typical, Cerebral Palsy). It is easy to notice that Cerebral Palsy cases incorrectly classified have been treated almost equally both as typical and Pes Planovalgus cases. This results from the small homogeneity of Cerebral Palsy group which includes cases at different levels of pathology. Among them there are cases which are similar to both typical or Pes Planovalgus group. The analysis of the percentage of correct classification for typical subjects and subjects affected by Pes Planovalgus leads to conclusion that the decision tree 'decided' to fit closer Pes Planovalgus subjects because the group of typical subjects is smaller. This proves that special attention should be paid to the quantity of each group.

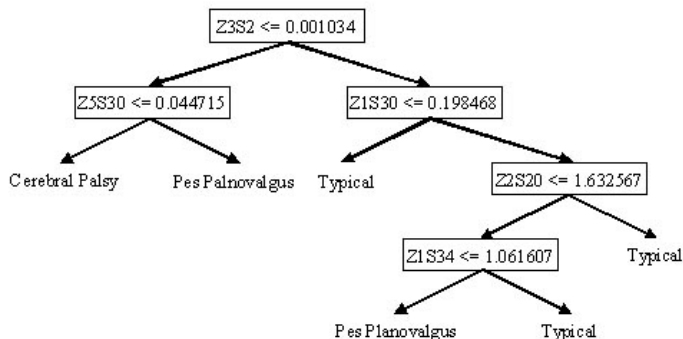
**Table 2.** Results of the classification by means of the decision tree

|         | predicted as |    |    | CCC    | OCC   |
|---------|--------------|----|----|--------|-------|
|         | typical      | CP | PP |        |       |
| Typical | 67           | 1  | 9  | 87.01% |       |
| CP      | 5            | 82 | 7  | 87.23% | 94.9% |
| PP      | 0            | 1  | 93 | 98.94% |       |

where CCC - denotes the percentage of correct classification in cases, OCC - the overall percentage of correct classifications.

A more detailed analysis of the decision tree (see Fig. 3) shows that the most important in decision making are the following values: samples 30 and 34 for the first zone (toes), sample 20 for the second zone (metatarsal bones), sample 2 for the third zone (cuboidal bone) and sample 30 for the fifth zone (heel bone). The results prove no significance of the navicular bone. At first glance, this could be really surprising because in Pes Planovalgus we should obtain much bigger foot pressure in the medial site of the foot than for typical subjects! In fact 75% of typical cases are discriminated from Pes Planovalgus cases by the 30th sample (58% of stride - toe off phase) in zone 1, when the pronation position of the foot affected a bigger pressure under hallux.

The main discriminator between Cerebral Palsy and other case is the second sample (the initial contact phase) of the third zone. It is obvious if we remember



**Fig. 3.** Decision tree - CART algorithm with cross validation

that the initial contact for Cerebral Palsy subjects with spasticity starts from toes.

## 5 Conclusions

The novel approach to automatic human gait analysis demonstrates that decision trees are a powerful technique which could be successfully applied in biomechanics. A decision tree could manage vector of a many numbers of real parameters as an input and point the values which are main discriminators. The advantage of employing decision trees is the easiness of interpretation so it could be applied successfully to clinics. Moreover, decision trees could improve the understanding of human gait phenomena and could lead to the selection of more suitable methods for human gait improvement. It is really important that the same procedure could be used in any data describing a human activity independent of the clinical problem.

## Acknowledgments

This paper was supported by the grant W/WM/1/09 from Bialystok Technical University.

## References

1. Armand, S., Watelain, E., Roux, E., Mercier, M., Lepoutre, F.X.: Linking clinical measurements and kinematic gait patterns of toe-walking using fuzzy decision trees. *Gait and Posture* 25, 475–484 (2007)
2. Begg, R., Kamruzzaman, J.: A machine learning approach for automated recognition of movement patterns using basic, kinetic and kinematic gait data. *Journal of Biomechanics* 38, 401–408 (2005)
3. Chau, T.: A review of analytical techniques for gait data. Part 1: fuzzy, statistical and fractal methods. *Gait and Posture* 13, 49–66 (2001)

4. Derlatka, M.: Application of Kernel principal component analysis in human gait. *Journal of Vibroengineering* 7(3), 27–30 (2005)
5. Derlatka, M., Ihnatouski, M., Lahkovski, V.: Biomechanics and correction of the foot's dysfunctions. In: Sviridenok, A.I., Lahkovski, V.V. (eds.) Grodno State University, Grodno (2009)
6. Derlatka, M., Pauk, J.: Data Mining in Analysis of Biomechanical Signals. *Solid State Phenom.*, vol. 147-149, pp. 588–593 (2009)
7. Ghoussayni, S., Stevens, C., Durham, S., Ewins, D.: Assessment and validation of a simple automated method for the detection of gait events and intervals. *Gait and Posture* 20, 266–272 (2004)
8. Hand, D., Mannila, H., Smith, P.: Principles of data mining. Polish edition. WNT, Warsaw (2005)
9. Ihnatouski, M., Sviridenok, A., Lashkovski, V., Krupicz, B.: Biomechanical analysis of antropometric and functional zones on human plantar walking. *Acta mechanica et automatica* 2(4), 19–23 (2008)
10. Jaworek, K.: Indicies metod of assessing human gait and run. In: IBIB PAN, Warsaw, vol. 32 (1992)
11. Mikut, R., Jakel, J., Groll, L.: Interpretability issues in data-based learning of fuzzy systems. *Fuzzy Sets and Systems* 150, 179–197 (2005)
12. Olney, S.J., Griffin, M.P., McBride, I.D.: Multivariate examination of data from gait analysis of persons with stroke. *Phys. Ther.* 78(8), 814–828 (1998)
13. Pauk, J., Jaworek, K.: Parametric identification of lower limbs during walking of a man. In: *Design Nature*, pp. 361–366. WIT Press, Southampton (2002)
14. Pretkiewicz - Abacjew, W., Erdmann, W.S.: Kinematics of walking of 6 year old children. *Journal of Human Kinetics* 3, 115–130 (2000)
15. Romei, M., et al.: Use of the normalcy index for the evaluation of gait pathology. *Gait and Posture* 19, 85–90 (2004)
16. Wolf, S., Loose, T., Schablowski, M., Doderlein, L., Rupp, R., Gerner, H.J., Bretthauer, G., Mikut, R.: Automated feature assesment in instrumented gait analysis. *Gait and Posture* 23, 331–338 (2006)
17. Wu, J., Wang, J., Liu, L.: Kernel-Based method for automated walking patterns recognition using kinematics data. In: Jiao, L., Wang, L., Gao, X.-b., Liu, J., Wu, F. (eds.) ICNC 2006, part II. LNCS, vol. 4222, pp. 560–569. Springer, Heidelberg (2006)
18. Yardimci, A.: Fuzzy Logic Based Gait Classification for Hemiplegic Patients. In: Berthold, M.R., Shawe-Taylor, J., Lavrač, N. (eds.) IDA 2007. LNCS, vol. 4723, pp. 344–354. Springer, Heidelberg (2007)

# Data Mining Approaches for Intelligent E-Social Care Decision Support System

Darius Drungilas<sup>1,3</sup>, Antanas Andrius Bielskis<sup>1</sup>, Vitalij Denisov<sup>1</sup>,  
and Dalė Dzemydienė<sup>2,3</sup>

<sup>1</sup> University of Klaipėda, Manto str. 84, Klaipėda, LT-92294, Lithuania  
dorition@gmail.com, {andrius.bielskis,vitalij.denisov}@ik.ku.lt

<sup>2</sup> Mykolas Romeris University, Ateities str. 20, Vilnius, LT-08303, Lithuania  
daledz@mruni.lt

<sup>3</sup> Institute of Mathematics and Informatics, Akademijos str. 4, Vilnius,  
LT-08663, Lithuania

**Abstract.** Large-scale of multidimensional recognitions of emotional diagnosis of disabled persons often generate large amount of multidimensional data with complex recognition mechanisms. The problem is to reveal main components of diagnosis and to construct flexible decision making support system. Sensors can easily record primary data, however the recognition of abnormal situations, clusterization of emotional stages and resolution for certain type of diagnosis is oncoming issue for bio-robot constructors. This paper analyses the possibilities of integration of different knowledge representation techniques, especially data mining methods, for development of the reinforcement framework with multiple cooperative agents for recognition of the prediction criteria of diagnosis of emotional situation of disabled persons. The research results present further development of model of framework with integration of the evaluation of data mining methods for wheelchair type robots working in real time by providing movement support for disabled individuals.

**Keywords:** emotion recognition, distributed information systems, multilayer perceptron, self-organizing maps, teacher noise.

## 1 Introduction

Computers and robots are rapidly entering areas of our lives that typically involve socio-emotional content, such as telephone computerized receptionist, service robots in hospitals, homes, and offices, internet-based patient advising (where patients read textual information about their diseases), internet-based health chat lines and computer mediated patient monitoring and caring. Many similar applications are in the making [4], [7], [11] and [14].

We recognize the possibilities to develop the integration of different types of knowledge representation techniques in the bio-robot system with working on-line sub-systems of complex mechanisms of cooperation of multi-agent's activities for human's affect sensing [3].

Our research area is concerning methods of integration in the developing of the adaptive, user-friendly e-health care services for people with movement disabilities. The decision support system under development depends upon the possibility of extracting emotion without interrupting the user during human-computer interaction (HCI) and using this information for patient monitoring and caring [2] and [5] as appropriate emotional state could be a key indicator of the patient’s mental or physical health status [7].

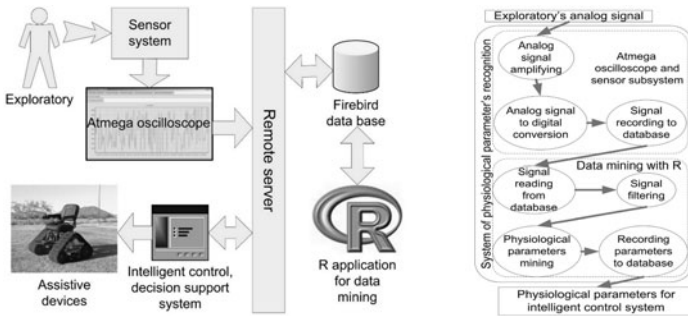
The features of continuous physiological activity of disabled person are becoming accessible by use of intelligent bio-sensors coupled with computers. Such sensors provide information about the wearer’s physical state or behaviour. They can gather data in a continuous way without having to interrupt the user and may include sensors for detecting of: Skin Conductance (SC), Blood Volume Pulse (BVP), Electrocardiogram (ECG), Respiration, Electromyogram (EMG), Body temperature (BT), and Facial Image Comparison (FIC). A number of wearable systems have been proposed with integrated wireless transmission, GPS (Global Positioning System) sensor, and local processing. Commercial systems are also becoming available [7] and [9].

In this article, we focus on hardware and software design for physiological parameters recognition based on continuous SC measuring. We propose methods for automatic emotional state recognition using data filtering, self-organizing maps (SOM) and multilayer perceptron (MLP).

## 2 Emotion Recognition Using Data Mining Methods

The proposed framework of intelligent emotion recognition and non-invasive human-machine interaction system is based on distributed information systems containing bio-data of exploratory and approaches capable to transform these data into valuable information for intelligent control, decision support system.

The main concept of this framework is shown in Fig. 1. In this case, the sensor system contains SC biometric sensor. The amplified SC physiological signal is



**Fig. 1.** The main components of hardware and software system interaction during the recognition processes of physiological parameters (left) and concept model of physiological parameters recognition system (right)

digitized and recorded into Firebird database via remote server by Atmega oscilloscope [5]. R tool connected to Firebird database via remote server was used in order to extract useful information from collected data. R, as widely used for statistical software development and data analysis, is used for data filtering and physiological parameters mining. The extracted information as a result is recorded to Firebird database, so that any intelligent control, decision support system connected to Firebird database could use this information.

The main process of physiological parameters recognition is to transform exploratory’s analog signal into physiological parameters so that they could be used by any intelligent control, decision support system, in order, to take patient monitoring and caring. The concept model of the main processes in the system of physiological parameters recognition is shown in Fig. 1. The signal’s filtering and physiological parameter’s mining mechanisms are used as components of physiological parameters recognition process.

### 2.1 Extraction of Physiological Parameters

The phase of recognition of physiological stages of persons deal with problem of extraction of important physiological parameters. The SC distribution data are analyzed (in Fig. 2, we can see typical SC curve). From stimulus point (when emotional change occurs), four characteristics can be extracted from SC data: latency, rise time, amplitude and half recovery time (see Fig. 2).



Fig. 2. SC characteristics by [13] (left) and distribution of emotional states by [10] (right)

If we assume, that  $y(i)$  is  $i^{th}$  digital SC data value, these parameters can be extracted from filtered SC data by such algorithm:

- Stimulus calls the calculations.
- While  $y(i+1)-y(i) \leq 0$  (curve does not rise), count the time – latency (Lat).
- While  $y(i+1)-y(i) > 0$  (curve rises) count the time and increasing SC value (RT and A respectively).
- Fix the curve maximum – MAX.
- While  $MAX-y(i) < A/2$  count the half recovery time (HRT).

After extraction of physiological parameters we need an approach to transform them to the set of emotional states, based on arousal and valence dimensions [10], where discrete emotional state could be allocated (see Fig. 2).

## 2.2 Emotional State Recognition

There are many different methods of recognizing physiological state by using data of wearer's emotion recognition sensors [8] and [9]. In this paper, we used multilayer perceptron (MLP) to transform physiological parameters to discrete emotional state. It was constructed by topology shown in Fig. 3. It is feed forward neural network containing two hidden layers. There are four neurons in input layer for SC physiological parameters and 8 neurons in output layer representing predictable emotional states shown in Fig. 2.

Adaptive gradient descend with momentum algorithm was used to train MLP. By this algorithm the weights are updated as:

$$w_{ij}^l(t) = w_{ij}^l(t-1) + \Delta w_{ij}^l(t) \quad (1)$$

$$\Delta w_{ij}^l(t) = -\gamma(t) \frac{\delta E_S(t)}{\delta w_{ij}^l(t-1)} + \lambda \Delta w_{ij}^l(t-1) \quad (2)$$

where  $w_{ij}^l(t)$  is the weight from node  $i$  of  $l^{th}$  layer to node  $j$  of  $(l+1)^{th}$  layer at time  $t$ ,  $\Delta w_{ij}^l(t)$  is the amount of change made to the connection,  $\gamma(t)$  is the self-adjustable learning rate,  $\lambda$  is the momentum factor,  $0 < \lambda < 1$ , and  $E_S$  is the criterion function. Minimizing the  $E_S$  by adjusting the weights is the object of training neural network.

The criterion function  $E_S$  usually consists of a fundamental part and an extended part. The fundamental part is defined as a differentiable function of relevant node outputs and parameters at appropriate time instants. The extended part is a function of derivatives of node output that is related to evaluation of criterion function. Therefore, the part is related to some notions that cannot be represented by the fundamental criterion, such as, smoothness, robustness, and stability. Here, the fundamental part is only considered.

$$E_S(t) = \frac{1}{2} \sum_{i=1}^S \sum_{j=1}^2 (y_j(t) + \hat{y}_j(t))^2 \quad (3)$$

where  $S$  is the total number of training samples.

The learning rate  $\gamma(t)$  is usually initialized as a small positive value and it is able to be adjusted according to the information presented to the network. The training process repeats until  $E_S$  is either sufficiently low or zero [6].

$$\gamma(t) = \begin{cases} \gamma(t-1)a_1, & 0 < a_1 < 1, E_S(t) \geq E_S(t-1) \\ \gamma(t-1)a_2, & a_2 > 1, E_S(t) < E_S(t-1) \end{cases} \quad (4)$$

The confusion matrix is often used for classification analysis, where a  $C \times C$  matrix ( $C$  is the number of classes) is created by matching the predicted values (in columns) with the desired classes (in rows). From the matrix, several metrics can be used to access the overall classification performance, such as the accuracy and precision.

The  $k$ -fold cross-validation is a robust estimation procedure, where the data sample is divided into  $k$  partitions of equal size. One subset is tested each time

and the remaining data are used for fitting the model. The process is repeated sequentially until all subsets have been tested. Therefore, under this scheme, all data are used for training and testing.

### 2.3 Teacher’s Noise Elimination

In order to collect experiment data (for MLP training and validation), the small application “State”, based on arousal valence space of Fig. 2 was developed. While tracking the physiological signal the exploratory fixes the stimulus time and discrete emotional state as identification number (see Fig. 3)

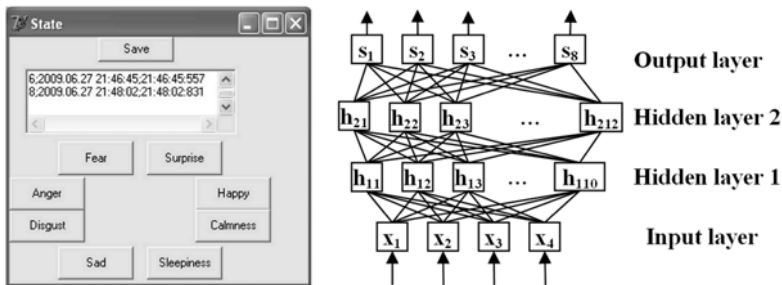


Fig. 3. Application for collection the experimental data (left) and MLP topology (right)

Sometimes, because of overlay, it is difficult to discriminate the emotional state. So the errors could come from labeling the data points (teacher noise). Classifying data into somewhat similar clusters can lead to noise reduction, and therefore, higher accuracy [1].

SOM, unsupervised self-learning algorithm, was used for clustering, that discovers the natural association found in the data. SOM combines an input layer with a competitive layer where the units compete with one another for the opportunity to respond to the input data. The winner unit represents the category for the input pattern. Similarities among the data are mapped into closeness of relationship on the competitive layer [12]. The SOM here defines a mapping from the input data space  $R^4$  onto a two-dimensional array of units. Each unit in the array is associated with a parametric reference vector weight of dimension four. Each input vector is compared with the reference vector weight  $w_j$  of each unit. The best match, with the smallest Euclidean distance is defined as response, and the input is mapped onto this location. Initially, all reference vector weights are assigned to small random values and they are updated as:

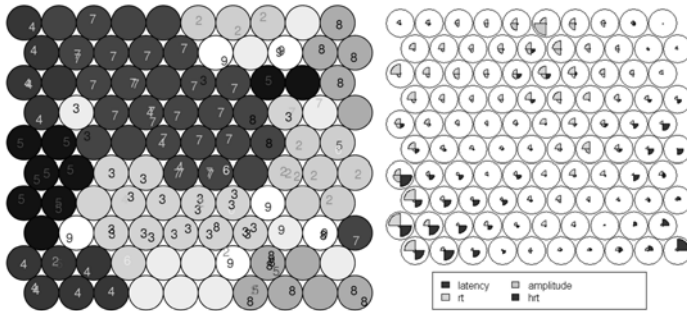
$$\Delta w_j = \alpha_n(t)h_j(g, t)(x_i - w_j(t)) \tag{5}$$

where  $\alpha(t)$  is the learning rate at time  $t$  and  $h_n(g, t)$  is the neighborhood function from winner unit neuron  $g$  to neuron  $n$  at time  $t$ . In general, neighborhood function decreases monotonically as a function of the distance from neuron  $g$  to neuron  $n$ . This decreasing property is a necessary condition for convergence.



### 3 Empirical Results and Discussion

All experiments reported in this work were written in R. First of all we will try to eliminate teacher noise with SOM. In Fig. 4, we can see  $10 \times 10$  SOM grids, where each unit contains  $R^4$  weight vector that groups SC parameters by similarities. The numbers represent training data classes, and color tone - different clusters after training. The SOM's units on competitive layer are arranged by similarities i.e. by Euclidean distance, so the training is measured as mean distance to the closest unit on each iteration of training.



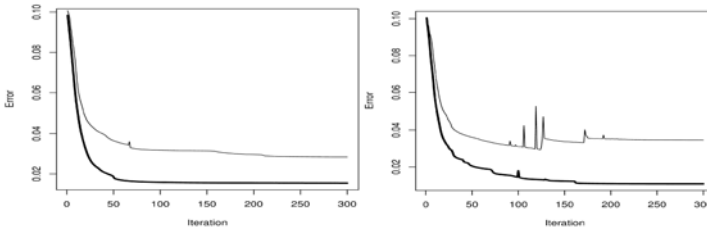
**Fig. 4.** Clustering SC parameters using SOM (left) and influence of SC parameters on each neuron of the SOM (right)

The clustering accuracy of SOM can be calculated by:

$$A(h|X) = \frac{\sum_{t=1}^N h(x^t) = r^t}{N} 100\% \tag{6}$$

where  $h(x)$  is hypothesis of assigning  $x$  to appropriate class,  $r^t$  - experts indicated (desired) class,  $N$  - classification sample,  $h(x^t) = r^t$  is equal to 1, when  $x^t$  is classified as  $r^t$ , and it is equal to 0 otherwise. The clustering accuracy calculated by (6) is 79.55%. So the classes of parameters of different states are distinguishable enough to make emotional state recognition. In order to know which factor is most important for emotional state classification, we will make clustering with SOM by each factor and calculate clustering accuracy by (6).

So the clustering accuracies by latency, rise time, amplitude and half recovery time are 44.70%, 52.27%, 52.27% and 48.48% respectively. The rise time and amplitude correlates with emotional states the most, and latency is least significant parameter for emotional state recognition. However all four SC parameters combined together give 27.28% higher accuracy (79.55%) than the best clustering (52.27%) by separate SC parameters. In Fig. 4, we can see the influence of SC parameters on each neuron, because of the clustering of emotional states described in Fig. 4 (left), has been made.



**Fig. 5.** MLP training progress for 2 of 50 experimental configurations

Further, we will use classified data by SOM for MLP training (the SOM outputs became inputs to the MLP) to find out if teacher noise elimination is important in our experiment. So we made two samples: first training sample was made from SOM's predicted data, and second – from data not processed by SOM.

To evaluate MLP classification, we adopted 10 runs of 5-fold cross-validation, in a total of  $10 \times 5 = 50$  experiments for each tested configuration. Statistical confidence will be given by the *t*-student test at the 90% confidence level.

MLP training progress for 2 of 50 experiment configurations, is shown in Fig. 5. It is given for the first and the second training samples as bold and thin lines respectively. As we see, convergence is more faster for first trainings sample. So it was useful to preprocess MLP's training sample with SOM, as, by using this approach, MLP easier finds the pattern.

Finally, after 5-fold cross-validation, the classifying accuracies were calculated:  $35.78 \pm 1.91\%$  and  $32.11 \pm 8.07\%$  for the processed and not processed with SOM training samples respectively. So we see that, using training sample preprocessed with SOM, the classifying accuracy increases by 3.67% and the training process is more stable.

## 4 Conclusion

An approach of integration of some data mining techniques for intelligent e-health care environment are proposed. The process of physiological parameters recognition is based on measurements of physiological signals taken from electrodes noninvasively attached on human body. The amplified SC signal is used in the model for physiological parameters recognition and emotional state clustering. The data sample of physiological parameters extracted from SC signals was preprocessed by SOM in order to reduce teacher noise that leads to higher accuracy of classification physiological parameters to discrete emotional state. It was shown that using data sample preprocessed with SOM, in MLP training the learning process is more faster than using not preprocessed data sample. The preprocessing also increases classification accuracy using MLP by 3.67%.

## References

1. Alpaydin, E.: *Introduction to Machine Learning*. The MIT Press, Cambridge (2004)
2. Bielskis, A.A., Denisovas, V., Drungilas, D., Gričius, G., Ramašauskas, O.: Modelling of intelligent multi-agent based e-health care system for people with movement disabilities. *Electronics and Electrical Engineering Kaunas: Technologija* 86(6), 37–42 (2008)
3. Bielskis, A.A., Dzemydienė, D., Denisov, V., Andziulis, A., Drungilas, D.: An approach of multi-agent control of bio robots using intelligent recognition diagnosis of persons with moving disabilities. *Technological and Economic Development of Economy* 15(3), 377–394 (2009)
4. Dzemydienė, D., Maskeliūnas, S., Dzemyda, I.: Interoperability of information system components for monitoring of sewage and intelligent analysis of water resources. *Technological and Economic Development of Economy* 14(3), 260–278 (2008)
5. Gričius, G., Drungilas, D., Šliamin, A., Lotužis, K., Bielskis, A.A.: Multi-agent-based e-social care system for people with movement disabilities. In: *Technologijos Mokslo Darbai Vakarų Lietuvoje*, pp. 67–77. Klaipėdos universiteto leidykla, Klaipėda (2008)
6. Han, M., Wang, Y.: Analysis and modeling of multivariate chaotic time series based on neural network. *Expert Systems with Applications* 36(2), 1280–1290 (2009)
7. Lisetti, C., Nasoz, F., LeRouge, C., Ozyer, O., Alvarez, K.: Developing multimodal intelligent affective interfaces for tele-home health care. *Int. J. Hum.-Comput. Stud.* 59(1-2), 245–255 (2003)
8. Mandryk, R.L.: *Modeling User Emotion in Interactive Play Environments: A Fuzzy Physiological Approach*. PhD thesis (2005)
9. Pentland, A.: Healthwear: medical technology becomes wearable. *IEEE Computer* 37(5), 42–49 (2004)
10. Russell, J.A.: A circumplex model of affect. *Journal of Personality and Social Psychology* 39(6), 1161–1178 (1980)
11. Sfakiotakis, M., Tsakiris, D.P.: Biomimetic centering for undulatory robots. *The International Journal of Robotics Research* 26(11-12), 1267–1282 (2007)
12. Talbi, M.L., Charef, A.: PVC discrimination using the QRS power spectrum and self-organizing maps. *Computer Methods and Programs in Biomedicine* 94(3), 223–231 (2009)
13. Wang, P., McCreary, H.: EDA sensor (2006), [http://courses.cit.cornell.edu/ee476/FinalProjects/s2006/hmm32\\_pjw32/index.html](http://courses.cit.cornell.edu/ee476/FinalProjects/s2006/hmm32_pjw32/index.html)
14. Zavadskas, E.K., Naimavičienė, J., Kaklauskas, A., Krutinis, M., Vainiūnas, P.: Multi-criteria decision support system of intelligent ambient assisted living environment. In: *25th International Symposium on Automation and Robotics in Construction, ISARC 2008, Selected papers*, pp. 717–724 (2008)

# Erythematous-Squamous Diseases Diagnosis by Support Vector Machines and RBF NN

Vojislav Kecman<sup>1</sup> and Mirna Kikec<sup>2</sup>

<sup>1</sup> Virginia Commonwealth University, SoE, CS Department, Richmond, VA, USA

<sup>2</sup> Ustanova za hitnu medicinsku pomoc, Zagreb, Croatia

**Abstract.** The paper presents the results of using Support Vector Machines (SVMs) and Radial Basis Function Neural Networks (RBF NNs) for diagnosing erythematous-squamous diseases which represent difficult dermatological problems. The data sets contains 358 data pairs of 34 dimensional input records of patients with six known diagnosis (outputs). Thus, data set is sparse and it is fairly unbalanced too. The paper also discusses the strategies for training SVMs. Both networks design six different one-against-other classifier models which show extremely good performance on previously unseen test data. The training and the test sets are obtained by random splitting the dataset into two groups ensuring that each group contains at least one patient for each disease. 100 random split trials (equivalent to performing 10-fold-crossvalidation 10 times independently) were carried out for estimating the tests error rates.

## 1 Introduction

Two related machine learning models Support Vector Machine (SVM) and Radial Basis Function Neural Network (RBF NN) are applied to the data set used in [1], [2], and [3] describing six dermatological diseases. Data set can be downloaded from <http://www.cormactech.com/neunet/sampdata/dermatology.html> [4]. There are 358 data pairs from the known patients containing 34 dimensional inputs (34 features i.e., attributes) and one dimensional output denoting one-out-of-six diseases. (There are actually 366 data pairs in [4], but 8 data pairs with missed age information have been discarded here). The dermatological diseases contained in a data set are as follows - psoriasis (112 instances), seboric dermatitis (61), lichen planus (72), pityriasis rosea (49), cronic dermatitis (52) and pityriasis rubra (20). Each skin disease is labeled with a single number corresponding to the order as given above. Thus, each output will have one of the labels from 1 to 6. Obviously, in addition to being sparse, the data set is not balanced either because there is, for example, 6 times more patients having psoriasis than the ones with pityriasis rubra. Both characteristics (sparseness and misbalance) usually make the problem of a good classifier design very difficult. As for the inputs (features, attributes), they can be separated into two basic groups - clinical and histopathological. Features corresponding to the clinical group are - 1: erythema, 2: scaling, 3: definite borders, 4: itching, 5: koebner phenomenon,

6: polygonal papules, 7: follicular papules, 8: oral mucosal involvement, 9: knee and elbow involvement, 10: scalp involvement, 11: family history. The next 22 attributes belonging to the histopathological group are - 12: melanin incontinence, 13: eosinophils in the infiltrate, 14: PNL infiltrate, 15: fibrosis of the papillary dermis, 16: exocytosis, 17: acanthosis pilaris, 18: hyperkeratosis, 19: parakeratosis, 20: clubbing of the rete ridges, 21: elongation of the rete ridges, 22: thinning of the suprapapillary epidermis, 23: pongiform pustule, 24: munro microabcess, 25: focal hypergranulosis, 26: disappearance of the granular layer, 27: vacuolization and damage of basal layer, 28: spongiosis, 29: saw-tooth appearance of retes, 30: follicular horn plug, 31: perifollicular parakeratosis, 32: inflammatory mononuclear infiltrate, 33: band-like infiltrate. Finally, the last i.e., 34th input is the age of the patient. All the features (clinical and histopathological), except the family history and age, was given a degree in the range of 0 to 3, where 0 indicates that the feature was not present, 3 indicates the largest amount possible, and 1, 2 indicate the relative intermediate values. The family history feature can have two values only; 1 if any of these diseases has been observed in the family, and 0 otherwise. The age attribute represents the age of the patient which was scaled between 0 and 3 here. (A more detailed description of the data set characteristics can be found at the site given above [4]).

## 2 Modeling and Experimenting with a Data Set

### 2.1 Previous Experiments and Results

The data set introduced has been subject of extensive modeling by using various data mining tools showing relatively similar results [1], [2], [3]. In [1] a new classification algorithm, called VFI5 (for Voting Feature Intervals-5), is developed and applied to a problem of differential diagnosis of erythemato-squamous diseases. By using a 10-fold cross-validation (CV) technique an average error of 96.2% accuracy has been achieved. The result mentioned was improved to 99.2% by adapting the weights and by using genetic algorithms.

In the research report [2] three different data modeling tools have been used within an educational visual software - in addition to the VFI5 algorithm, the Nearest Neighbor Classifier and Nave Bayesian Classifier using Normal Distribution have been implemented but the error rates of the models are not given. This hints that the results are of similar accuracy as the ones given in [1].

An entirely different approach is presented in [3], namely the application of a particular neuro-fuzzy system, named KERNEL, to the same problem of differential diagnosis of erythemato-squamous diseases ' which represents a major problem in dermatology. A multi-step learning strategy is adopted to obtain, starting directly from available data, a fuzzy rule base that can be used to identify the particular disease. The obtained classification results at the end of a two-phase experimental session are reported'. Same as in [1] a 10-fold CV has been adopted. The paper presents results of the unsupervised and supervised learning algorithms. The best reported average result for a supervised learning, after discarding the age feature, was 94.47% accuracy.

## 2.2 RBF NN and SVM Classification (Diagnosis) Models

In this paper, two related classification models have been used for developing a diagnosis tool - the RBF neural network and the SVM model [5] and [6]. In fact, if SVM is using Gaussian kernels, the SVM can be looked at as the RBF NN trained by a slightly different learning rule. Namely, at the end of a SVM's learning two tasks are performed - the selection of the Gaussian hyper-bells and the calculation of their corresponding weights. As for the RBF NN, these two problems are connected in the sense that the data miner must first pick up both the number and the position of the Gaussian basis functions and only then calculate the weights by using a standard pseudo-inverse operation of a design matrix in solving a least squares problem. Such a resemblance of the RBF NN and SVM ends if one uses polynomial kernels in the SVM's design. In addition to the two models mentioned, a standard linear classifier is also designed here, to compare it with a SVMs' models using linear kernel.

As for the experiments performed, the learning has been much stricter in this paper than in [1], [2], and [3] where a single 10-fold CV has been used (10 runs). Here, 100 experimental runs have been executed with data sets randomly split up into the training sets (90% of data pairs) and the test ones (10% data pairs). In other words this corresponds to 10 independent 10-fold CV runs. As already mentioned the problem is to build a diagnose system for six ( $K = 6$ ) dermatological diseases. Thus, one is solving a multi-class supervised classification problem here. The most common approach for multi-class learning is to transform the  $K$  classes problem into a set of  $K$  two-class problems, which is also known as one-against-others method, or one-versus-all (OVA) classification. A recent extensive paper, [7], has shown by that an OVA approach is as good as any other more sophisticated approach for multiclass classification. In such an approach, the first model separates and classifies the psoriasis (labeled as +1) against all the other five diseases (labeled as -1), the second model classifies the seboreic dermatitis (labeled as +1) against all the other five diseases (labeled as -1), and so on. Once all six classifiers are designed, the new input is supplied to all six models and the winner will be the one producing the highest value at the output.

**RBF NN Diagnosis Model Design:** This part of the paper will be relatively short not because the RBF NNs are generally bad tool, but due to the fact that for this particular six skin diseases diagnosis example, SVMs have shown much better performance. In fact, SVMs models have displayed perfect accuracy with 100% correct classification on the unseen test data randomly obtained in 100 experimental runs, while RBF NN performed very good only.

Three basic design issues for RBF classifier are - where to place the RBF functions, how many of them to use and what should be the value of the width parameter of an RBF [5]. (In the case of using Gaussian RBF, width parameters are the values of the covariance matrix). Once the three sets of parameters (number, positions and shapes of Gaussians) are known, a least squares problem for finding the output layer weights should be solved. Two basic ways have been used for placing the Gaussian basis functions - clustering and placing them at

each  $k$  training data points. Both methods gave similar results ending with a lower end of average errors over the unseen test data points around 2%. The elapsed CPU time on a 1.6Ghz laptop having 1.5Gb of memory and making 100 runs in MATLAB's R2006a version, for 7 different number of RBFs in hidden layer ([30 33 36 40 45 51 60]) and for 9 different Gaussian width (i.e., ) values [5 6 7 8 9 10 11 12.5 15] was 4010 seconds. Minimal average test error was 1.94%.

**SVM Diagnosis Model Design:** The SVM classifier was designed by using two different kernels - Gaussian and polynomial ones. As it is very well known, a training stage of SVMs' classifiers involves solving the QP problem where the symmetric positive definite Hessian matrix is an  $(N, N)$  matrix, and  $N$  is the number of training data pairs.

There are many different solving routines for SVMs falling in three basic groups [6]. If the problem is small enough to be stored completely in memory (on current PC hardware up to approximately 5000 data), interior point methods are suitable. They are known to be the most precise QP solvers but have a memory consumption of  $O(N^2)$ . For very large data sets on the other hand, there is currently no alternative to working-set methods (decomposition methods) like SMO [8], ISDA [9] and [10] or similar strategies. This class of methods has basically a memory consumption of  $O(N)$  and can therefore cope even with genuinely large scale problems. Active-set algorithms [6] are appropriate for medium-size problems because they need  $O(N_f^2 + N)$  memory where  $N_f$  is the number of free (unbounded) SV variables. Although  $N_f$  is typically much smaller than the number of the data, it dominates the memory consumption for large data sets due to its quadratic dependency. Common SVM software packages rely on working-set methods because  $N$  is often large in practical applications. However, in some situations this is not the optimal approach, e.g., if the problem's Hessian matrix is ill-conditioned, if the SVM penalty parameter  $C$  is not chosen carefully, or if high precision is needed [6]. Active-set algorithms are the classical solvers for QP problems. They are known to be robust, but they are sometimes slower and (as stated above) require more memory than working-set algorithms. Their robustness is in particular useful for cross-validation techniques where the SVM parameters are varied over a wide range. For data set used here the active-set algorithm has been implemented as a C MEX-file under MATLAB R2006a for classification. This routine has been developed by M. Vogt in [6] and it has been adapted for solving a multi-class skin diseases problems by the author here. It can handle both fixed and variable bias terms  $b$ . The variable bias  $b$  has been used in all simulations here.

There are few basic SVM's design parameters while building the classifier. The penalty parameter  $C$  as well as the shape parameter for Gaussian kernels and order of the polynomial for the polynomial ones respectively.

SVM classifier performed perfectly over the broad range of parameters' values (i.e., the test errors have been equal zero for a range of SVM's parameters) for both kernels. Fig. 1 shows the performance of the Gaussian kernels SVM calculated for the following eight different values of the penalty parameter  $C = [0.5 1 10 50 100 250 500 1000]$  and for 11 different shape parameters of Gaussian

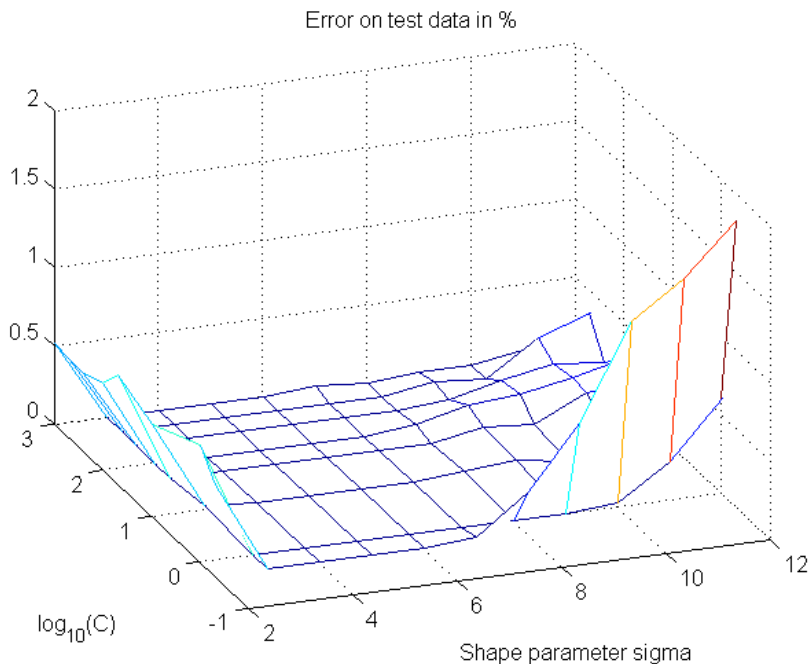
bells = [2 3 4 5 6 7 8 9 10 11 12]. As it can be seen from Fig. 1 perfect classification (diagnosis) after 100 random runs has been achieved for all C values whenever was equal to [3 4 5 6]. Also, for  $C = 10$  perfect diagnosing has been achieved for all values except for  $\sigma = 2$  when the test error was 0.67.

These results on the skin diseases data used here are superior to any other published results known at the moment of writing the paper.

The CPU time elapsed for performing 100 experimental runs 88 times has been 13,175 seconds = 3.66 hours.

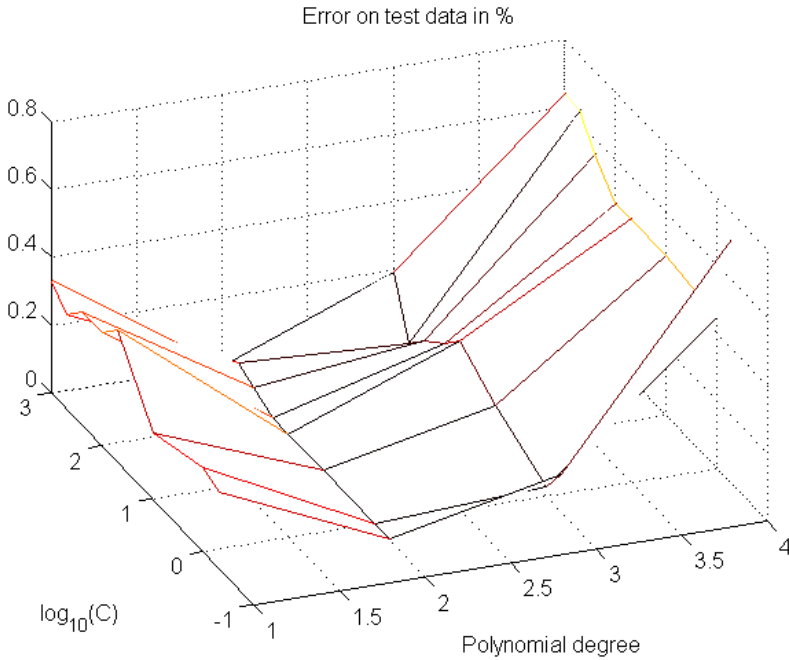
In addition to perform perfectly, the SVM can give more information and hints about the character of the problem. The average number of support vectors (SVs) in six models may give hints about the difficulties in diagnosing each particular disease. For the results shown in Fig. 1 the average number of SVs has been as follows 45.75 for pityriasis rubra, 49 for cronic dermatitis, 50.21 for psoriasis, 50.31 for lichen planus, 54 for pityriasis rosea and 59.64 for seboreic dermatitis. The smaller number of SVs, the easier diagnosis. Thus, results shown hint that diagnosing pityriasis rubra is easier than classifying other skin diseases. In particular, it seems to be easier than diagnosing pityriasis rosea and seboreic dermatitis.

Similar results are also present in Fig. 2 where the performance of the polynomial SVM calculated for the same eight values of the penalty parameter  $C = [0.5 \ 1 \ 10 \ 50 \ 100 \ 250 \ 500 \ 1000]$  is shown. The polynomials of the 1st, 2nd, 3rd



**Fig. 1.** An average error of the SVM using Gaussian kernels on test data after 100 random runs





**Fig. 2.** An average error of the polynomial SVM on test data after 100 random runs

and 4th order are used. As it can be seen from Fig. 2 that perfect classification (diagnosis) after 100 random runs has been achieved by 2nd order polynomial for almost all C values. Test error equals zero for  $C = [ 1 \ 10 \ 50 \ 100 \ 500 ]$ , and for all other orders and C values it's never bigger than 0.7% only.

Final models for both RBF NN and SVMs when Gaussian kernel has been used is given in (1). Both models are in a form of a weighted sum of basis functions. The fundamental difference is in the norm (i.e., cost) function used which reflects in two different learning algorithms for finding weights  $v_i$ . For RBF NN sum of errors squares was a cost function used while for SVMs a structural risk minimization principle has been implemented. Additionally, for RBF NN the position and number of basis functions is given by model builder while in the SVM design they are results of a learning algorithm.

$$y_{out} = v_1 e^{-\frac{1}{2} \left( \frac{x_{in} - x_1}{\sigma} \right)^2} + v_2 e^{-\frac{1}{2} \left( \frac{x_{in} - x_2}{\sigma} \right)^2} + \dots + v_{N_{SV}} e^{-\frac{1}{2} \left( \frac{x_{in} - x_{N_{SV}}}{\sigma} \right)^2} + b \quad (1)$$

Similar to Gaussian SVM models polynomial ones also performed perfectly for the range of parameters values and they can also give more information and hints about the character of the problem. For the results shown in Fig. 2 the average number of SVs has been as follows 15.52 for pityriasis rubra, 19.1 for lichen planus, 21.35 for chronic dermatitis, 21.36 for psoriasis, 27.93 for seboreic dermatitis and 28.26 for pityriasis rosea. Results shown for polynomial SVM also

hint that diagnosing pityriasis rubra might be easier than classifying other skin diseases. And again, classifying pityriasis rubra seems to be easier than diagnosing seboric dermatitis and pityriasis rosea. Polynomial SVM learns about 10 times faster than the Gaussian SVM for the data set used here. This is due to two facts - first, calculation of Hessian matrix involved in solving the QP learning problem is much quicker (because the computation of an exponential function is avoided) and second, polynomial SVM needs much less SVs which also speeds the training significantly up. The fact that the 2nd order polynomial decision functions have shown perfect diagnosing (classification) properties for the broad range of penalty parameter  $C$  hints that the six skin diseases may be normally distributed within the 34-dimensional input space having only different covariance matrices.

There is one more interesting characteristics of polynomial SVMs in classification. As it can be seen from Fig. 2 even the SVM using the first order (i.e., linear) polynomial has very small diagnostic errors. Namely, the maximal error on test data is not exceeding 0.44 %. Therefore, it may be interesting to compare the performance of such a linear SVM to the standard linear classifier minimizing the sum of error squares. Here, the standard linear classifier is slightly changed by using the Tychonov regularization. Thus, a series of regularized linear classifiers has been designed by using a broad range for a regularization parameter  $\lambda$ , but the error on previously (during the training) unseen test data, could have dropped only to around 2% which is much worse in comparisons to the linear SVM as shown in Fig. 2. This result just confirms the great potential of the SVM learning algorithm which is based on the maximization of a margin between the classes in input domain, and not on a minimization of the sum of error squares in the output one.

### 3 Conclusions

The paper shows an application of RBF NN and SVM for diagnosing skin diseases by transforming a  $K$ -class problem into  $K$  two class problems (one-vs-all approach for multiclass problems). It has been shown that SVM models perform better than RBF NN ones, and that the SVM models using both and polynomial kernels can perfectly classify, during the training unseen, test data. For a given data set SVMs using the polynomials of second order were particularly efficient and accurate. They use only between 5% and 10% of training data as the support vectors achieving perfect, error-less, diagnosis. The SVM was trained by using an active set method for solving SVMs' QP based learning problem. The results shown are the best known to date for diagnosing erythemato-squamous diseases which represent difficult dermatological problems. In addition, the number of support vectors used in each one out of  $K=6$  SVM models hints about which of diseases may be easy to diagnose (classify) and which may be more difficult diagnosing task.

## References

1. Guvenir, H.A., Demiroz, G., Ilter, N.: Learning differential diagnosis of erythematous-squamous diseases using voting feature intervals. *Artificial Intelligence in Medicine* 13(3), 147–165 (1998)
2. Emeksiz, N., Guvenir, H.A.: Application of machine learning techniques to differential diagnosis of erythematous-squamous diseases, report bu-ceis-9910. Bilkent University (Ankara) (1998)
3. Castellano, G., Castiello, C., Fanelli, A.M.: Diagnosis of dermatological diseases by a neuro-fuzzy approach. In: *International Conference in Fuzzy Logic and Technology (EUSFLAT 2003)*, Zittau, Germany (2003)
4. Database, D.: <http://www.cormactech.com/neunet/sampdata/dermatology.html>
5. Kecman, V.: *Learning and Soft Computing - Support Vector Machines, Neural Networks, Fuzzy Logic Systems*. The MIT Press, Cambridge (2001), [www.support-vector.ws](http://www.support-vector.ws)
6. Vogt, M., Kecman, V.: Active-Set Methods for Support Vector Machines. In: Wang, L. (ed.) *Support Vector Machines: Theory and Applications*. *Studies in Fuzziness and Soft Computing*, vol. 177, pp. 133–158. Springer, Berlin (2005)
7. Rifkin, R., Klautau, A.: In defense of one-vs-all classification. *Journal of Machine Learning Research* 5, 101–104 (2004)
8. Platt, J.C.: Fast training of support vector machines using sequential minimal optimization. In: Schoelkopf, B., Burges, C.J.C., Smola, A.J. (eds.) *Advances in Kernel Methods - Support Vector Learning*. The MIT Press, Cambridge (1999)
9. Kecman, V., Huang, T.M., Vogt, M.: Active-Set Methods for Support Vector Machines. In: Wang, L. (ed.) *Support Vector Machines: Theory and Applications*. *Studies in Fuzziness and Soft Computing*, vol. 177, pp. 255–274. Springer, Berlin (2005)
10. Huang, T.M., Kecman, V., Kopriva, I.: *Kernel Based Algorithms for Mining Huge Data Sets, Supervised, Semi-supervised and Unsupervised Learning*. Springer, Berlin (2006), [www.learning-from-data.com](http://www.learning-from-data.com)

# Neural Network-Based Assessment of Femur Stress after Hip Joint Alloplasty

Marcin Korytkowski<sup>1,2</sup>, Leszek Rutkowski<sup>1,3</sup>, Rafał Scherer<sup>1,3</sup>,  
and Arkadiusz Szarek<sup>4</sup>

<sup>1</sup> Department of Computer Engineering, Częstochowa University of Technology  
al. Armii Krajowej 36, 42-200 Częstochowa, Poland  
marcin.korytkowski@kik.pcz.pl, lrutko@kik.pcz.czyst.pl,  
rafal@ieee.org

<http://kik.pcz.pl>

<sup>2</sup> Olsztyn Academy of Computer Science and Management  
ul. Artyleryjska 3c, 10-165 Olsztyn, Poland

<http://www.owskiiz.edu.pl/>

<sup>3</sup> Academy of Management (SWSPiZ), Institute of Information Technology,  
ul. Sienkiewicza 9, 90-113 Łódź, Poland

<http://www.swspiz.pl/>

<sup>4</sup> Institute of Metal Working and Forming, Quality Engineering and Bioengineering  
Czestochowa University of Technology

szarek@iop.pcz.pl

<http://iop.pcz.pl/>

**Abstract.** Neural networks are a practical tool for solving various problems of approximation, classification, prediction or control. In the paper we use multi-layer perceptrons to determine the character of stress in healthy femur and after endoprosthesoplasty. Inserting metal prosthesis to the bone changes the stress character what can lead to local decalcification and weakening of its strength in certain areas. Dynamic bone load resulting from non-anatomical load can cause fracture in the weak area. Neural network was learned with the data obtained from numerical simulations using the finite element analysis. The input to the network was stress state in twelve points of femur and body mass.

## 1 Introduction

Hip joint prostheses are one of the most widely used prostheses in the surgery. It results from the fact that human hip joint transmits the highest forces and is often subjected to disease processes. The mechanical injuries caused by e.g. car accidents are also of a significant importance and their number is still increasing. Manufactured prostheses differ from each other with their geometry, method of stem fixation (cemented, uncemented) as well as the geometry and the method of socket fixation [1]. The clinical results for total cemented and uncemented hip joint replacement deteriorate in the course of time due to different reasons. Most of the failures are the results of the aseptic loosening i.e. slow, but progressive process, often coexisting with the bone defect. The reason for loosening of the prosthesis is a number of still not fully recognized factors, but undoubtedly the

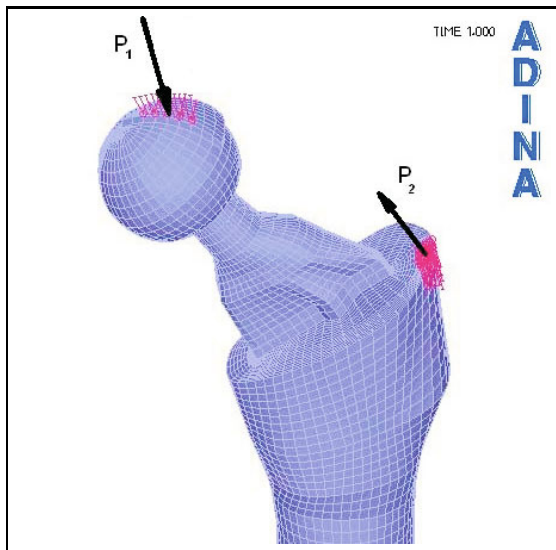
most decisive influence on such unfavourable phenomena is from the forces transmitted by a hip joint, causing significant (and undetermined) displacements and tensions in the bone-implant system [2]. Further development of hip joint arthroplasty depends on proper selection of the physical and mechanical properties of the prostheses to anatomical and physiological requirements of the osseous tissue. A fundamental assumption here is the design of such implant models which do not generate any stresses causing osteoporosis and therefore initiate their loosening. One of the opportunities to assess the types of prosthesis and the fixation techniques is to perform investigations on the simulation models for the conditions occurring in the human body. Artificial neural networks (NN) [3] [12] [13] are structures consisted of interconnected artificial neurons to solve various problems (regression, classification or approximation) without the necessity to build a formal mathematical model. Neural networks are relatively rarely used in orthopaedics. Grisby et al [5] used NN to predict functional outcomes, length of stay, and costs among orthopedic patients admitted to an inpatient rehabilitation hospital. In this paper, a neural network is used to determine femur stress characteristics of healthy hip joint and the joint after alloplastics. Inserting metal prosthesis to the bone changes the stress character what can lead to local decalcification and weakening of its strength in certain areas. Dynamic bone load resulting from non-anatomical load can cause fracture in the weak areas, thus it is very important to keep stress value map at the correct level.

## 2 Numerical Model

In order to determine locations where bone transformations occur, numerical calculations have been performed, accompanied with a preparation of bone-cement-implant model. Geometrical features of this bone were based on "Standardized Femur" model designed by Istituto Ortopedii Rizzoli in Bologna, where profiles of trabecular and cortical bones were prepared. The bone was modelled as an element of linear-elastic mechanical properties, isotropic, composed of two types of materials with strength parameters determined in references [6] [7] [11]:

- cortical bone - Young's modulus  $E = 2.1 \cdot 10^4 \div 1.68 \cdot 10^4$  [MPa]; Poisson's ratio  $\nu = 0.3$ ;
- trabecular bone - Young's modulus  $E = 1.6 \cdot 10^3 \div 1.1 \cdot 10^3$  [MPa]; Poisson's ratio  $\nu = 0.4$ ;
- bone cement - Young modulus  $E = 1.1 \cdot 10^3$  [MPa]; Poisson's ratio  $\nu = 0.4$ ;

Discrete model of bone-cement-implant comprised 84435 elements with hexahedron shape of 3D Solid type based on 81171 nods. Numerical investigations have been performed by means of the Finite Element Method. In the load model, the interaction of external forces which were placed on the femoral bone head surface and force from abductor muscles which was placed to the surface of small trochanter according to Fig. 1. Value of load forces in the system relates to load during normal walk at the moment of leg touching the ground, where force values are lowest as compared to the whole walking cycle. Nature of stresses has been presented for the whole bone and in six cross-sections determined in Gruen zones, Fig. 2. It has been accepted that



**Fig. 1.** Model of hip joint load

$BW = 569.687$  [N] (Body Weight), thus values of forces which load the system were accordingly  $P1 = 2.47\%BW$  and  $P2 = 1.55\%BW$ . Bone fixation was performed by removing degrees of freedom in femoral bone condyles in knee joint. Mechanical properties of human cortical bone depend on many factors, i.e. age, sex, health state, lifestyle, undergone operations and geometrical factors. Short-term bone strength after twenty years of age reduces by approximately 5% every ten years [10][8][9]. We assumed the following properties of human bone:

- for cortical bone:
  - density  $\rho = 1500 - 2000$  [kg/m<sup>3</sup>],
  - Young's modulus  $E = 5 - 20$  [GPa],
  - Poisson's ratio  $\nu = 0.2 - 0.4$ ,
  - immediate strength  $R_m = 80 - 150$  [MPa],
- for trabecular bone:
  - density  $\rho = 100 - 1200$  [kg/m<sup>3</sup>],
  - Young's modulus  $E = 100 - 5000$  [GPa],
  - Poisson's ratio  $\nu = 0.2 - 0.4$ ,
  - immediate strength  $R_m = 5 - 20$  [MPa],

To prepare data for learning, two groups of age 75 to 85 were used - one group of 70 persons with degenerative hip joint changes and one group of 20 healthy persons. This age interval was chosen as in this age there is the highest percentage of hip arthroplasty. For preparing numerical models, the shape of femoral bone was selected to be similar to the anatomically correct one. According to clinical tests results, diversified bone strength parameters were assumed, which resulted from the decrease of mechanical strength in cortical and trabecular bone with ageing processes.

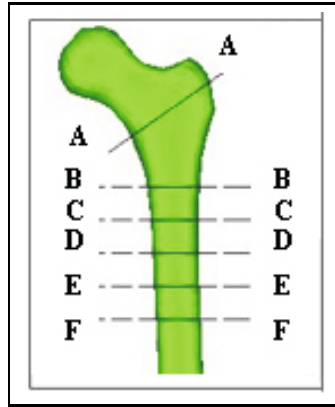


Fig. 2. Division parts in trochanter

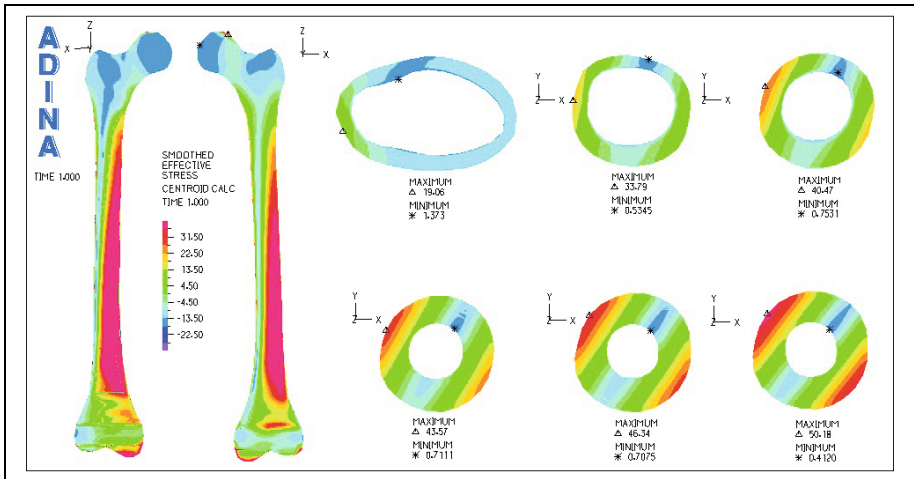


Fig. 3. Distribution of reduced stress  $\sigma_{zr}$  [MPa] in cortical bone. Model of normal human bone.

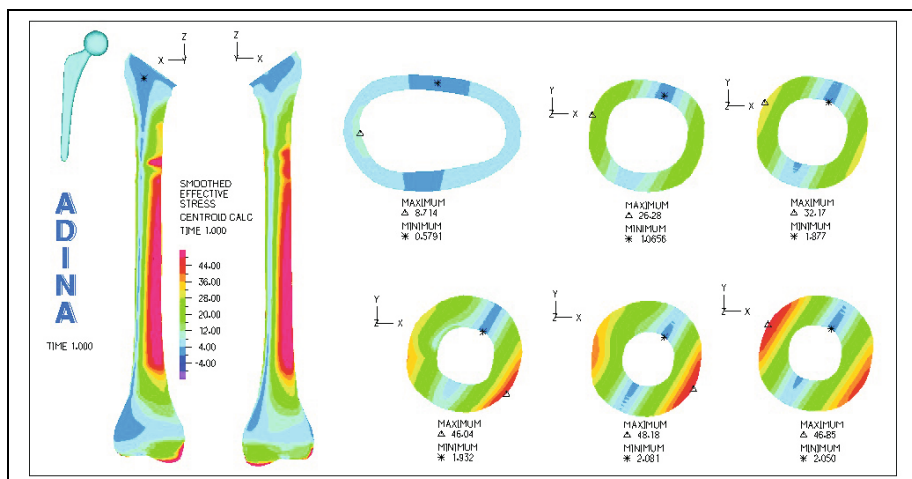
### 3 Numerical Simulations

We used multilayer perceptron and the weight were modified by the backpropagation algorithm [12] [13]. Having given learning data set of pair  $(\bar{x}, d)$  where  $d$  is the desired response of the system, we can use the following error measure

$$Q(\bar{x}, d) = \frac{1}{2} [\bar{y}(\bar{x}) - d]^2 \tag{1}$$

Every neural network, denoted for simplicity as  $w$ , can be determined by minimizing the error measure in the iterative procedure. For every iteration  $t$ , the parameter value is computed by

$$w(t + 1) = w(t) - \eta \frac{\partial Q(\bar{x}, d; t)}{\partial w(t)} \tag{2}$$



**Fig. 4.** Distribution of reduced stress  $\sigma_{zr}$  [MPa] in trabecular bone. Model of bone with implanted Centrament stem.

where  $\eta$  is a learning coefficient, set in simulations to 0.02. Dataset used in experiments was divided randomly into a learning set and a testing set. To train the neural network we used input data obtained on the basis of computer simulations using the finite element analysis to identify the stress state of healthy femoral bone. The research was performed for 70 patients of 75–85 years of age which femur bones were anatomically correctly shaped. The load resulting from the body mass was changed from 70 to 58 kg. On the basis of stress maps, stress values were read in the areas corresponding to Gruen zones, used commonly in orthopaedics, see Fig 2. For such setup, 13 input features were obtained - 12 stress values and body mass. Correct stress values were assumed to be values for healthy femoral bones. Examination of femurs after alloplastics were performed on a group of 20 patients in the age of 75–85 years. The data were divided into two classes - correct and incorrect stress values. We used two-layer nonlinear neural network with 5 neurons in the hidden layer and one output. The dataset was divided into learning and testing sets and we obtained 100% classification accuracy on the testing set.

## 4 Conclusions

Nonlinear neural network was used to classify stress characteristics in healthy femur and femur after endoprosthesis. Inserting metal prosthesis to the bone changes the stress character what can lead to local decalcification and weakening of its strength in certain areas. Dynamic bone load resulting from non-anatomical load can cause fracture in the weak areas, thus it is very important to keep stress value map at the correct level. Neural network was learned with the data obtained from numerical simulations using the finite element analysis. The input to the network was stress state in twelve points of femur and body mass. We obtained 100% classification accuracy of correct and pathological femur stress values.



## Acknowledgments

This work was partly supported by the Polish Ministry of Science and Higher Education (Habilitation Project 2007-2010 Nr N N516 1155 33, Polish-Singapore Research Project 2008-2010 and Research Project 2008-2011) and the Foundation for Polish Science – TEAM project 2010-2014.

## References

1. Amsutz, H.C., Grogoris, P., Dorey, F.J.: Evolution and future of surface replacement of the hip. *Journal Orthop. Sci.* 3(3) (1998)
2. Bernakiewicz, M.: Strain Analysis of Femur. In: *Biology of Sport 1998*, Kokotek k. Lublica, Poland, September 14-16, Vol. 15 (1998) (in Polish)
3. Bishop, C.M.: *Neural Networks for Pattern Recognition*. Oxford University Press Inc., New York (1995)
4. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*, 2nd edn. Wiley, Chichester (2000)
5. Grigsby, J., Kookan, R., Hershberger, J.: Simulated neural networks to predict outcomes, costs and length of stay among orthopedic rehabilitation patients. *Arch. Phys. Med. Rehabil.* 75(10), 1077–1081 (1994)
6. Krzesinski, G., Zagrajek, T.: Modelling Mechanical Properties on Bones. In: *Biology of Sport 1997*, vol. 17(Suppl. 8), pp. 238–243 (1997) (in Polish)
7. Mann, K.A., Bartel, D.L., Wright, T.M., Burstein, A.H.: Coulomb frictional interfaces in modeling cemented total hip replacements: a more realistic model. *J. Biomech.* 28(9), 1067–1078 (1995)
8. Natali, A.N.: Meroi: A review of the biomechanical properties of bone as material. *J. Biomed. Eng.* 11, 266–276 (1989)
9. Nigg, B.M., Herzog, W.: *Biomechanics of the musculoskeletal system*. J.Wiley&Sons, England (1944)
10. Nordin, M., Frankel, V.H.: *Basic biomechanics of the musculoskeletal system*. Lea & Fabiger (1989)
11. Reilly, D.T., Burstein, A.H.: The mechanical properties of cortical bone. *The Journal of Bone and Joint Surgery* 56 (1974)
12. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representations by error propagation. In: Rumelhart, D.E., McClelland, J.L., the PDP Research Group (eds.) *Parallel Distributed Processing. Explorations in the Microstructure of Cognition. Foundations*, vol. 1, pp. 318–362. The MIT Press, Cambridge (1986)
13. Russell, S.J., Norvig, P.: *Artificial Intelligence. A Modern Approach*. Prentice Hall, Englewood Cliffs (1995)

# Automated Detection of Dementia Symptoms in MR Brain Images

Karol Kuczynski<sup>1</sup>, Maciej Siczek<sup>2</sup>, Rafał Stegierski<sup>1</sup>, and Waldemar Suszyński<sup>1</sup>

<sup>1</sup> Maria Curie-Skłodowska University  
Pl. M. Curie-Skłodowskiej 1  
20-031 Lublin, Poland

`karol.kuczynski@umcs.lublin.pl`

<sup>2</sup> Hospital of Ministry of Interior and Administration  
ul. Grenadierów 3  
20-331 Lublin, Poland

**Abstract.** Alzheimer's, Parkinson's and other dementive diseases pose nowadays both medical and social problems. Image data provides diagnostic information on their crucial symptoms. However, mainly due to time constraints and various technical issues, not all useful information is in most cases extracted from the acquired radiological image data. The authors' aim is to project, implement and test a framework that supports and automatizes an in-depth analysis (including various fractal, statistical and volumetric properties) of MR (Magnetic Resonance) images for this purpose. Major elements of this system have been already created.

**Keywords:** medical image analysis, image classification, brain atrophy.

## 1 Introduction

Alzheimer's, Parkinson's and other dementive diseases [1] pose nowadays both medical and social problems. Their diagnose is not straightforward. It is based mainly, but not solely, on image data. Radiological images provide valuable information on the crucial symptoms. Brain atrophy is the most important one.

Modern imaging techniques like CT (Computed Tomography) or MRI (Magnetic Resonance Imaging) are available even in many local hospitals. Combined Computed Tomography/Positron Emission Tomography examination is particularly interesting. However there are nowadays only six such units in Poland [2] for a population of almost 40 million and their availability is probably not going to be significantly improved soon.

Unfortunately, fast image analysis performed by a radiologist visually only does not extract all useful information included in an image. Brain atrophy is associated with dementive disorders, but also results from normal ageing processes. That is why it is important to estimate atrophy level, its kind, and especially its time progress objectively. In case of progressive diseases, including those related to dementia, analysis of integrated images of the same patient acquired in different time (and sometimes also different place) may also provide valuable content.

Such analyses are both technically complicated and time-consuming. Because of these factors, they are not often performed thoroughly.

The authors' aim is to project, implement and test a highly automated software framework that helps radiologists to perform an in-depth analysis of image series in order to diagnose dementive disorders reliably. The proposed system consists of the following main modules:

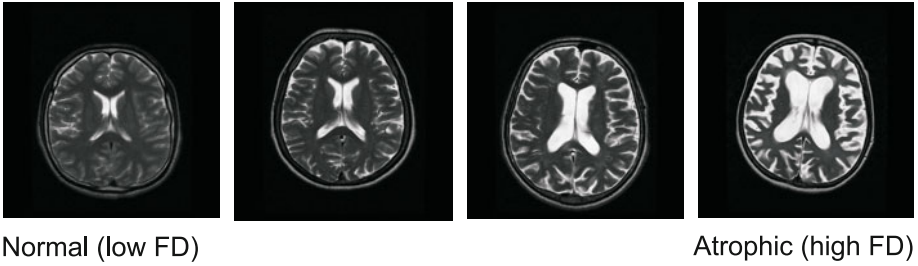
- image registration procedure (maximization of mutual information),
- brain extraction and tissue segmentation,
- analysis of the integrated images (fractal, statistical and volumetric properties).

They have been already partly implemented and evaluated by an experienced radiologist. The employed methods are generally known. However, the tests' results that can be found in literature are often performed with carefully selected datasets or datasets acquired in well controlled conditions for the purpose of image processing and analysis. The problem to be solved is to select appropriate algorithms, combine and tune them in order to work reliably with not always perfect images, acquired during routine medical procedures.

## 2 Materials and Methods

In order to perform a comparative analysis of two or more 3-dimensional image datasets, it is necessary to register them. It is the most time-consuming and probably the most problematic element of the framework. The images are registered by maximization of mutual information [3,4] (algorithm variation by Mattes et. al [5]), using affine transformation. Because of nature of the optimization criterion (numerous local extrema) and the process (regular step gradient descent [3]), correct localization of the global extreme is never guaranteed. In order to maximize likelihood of finding the exact registration parameters and to accelerate the whole process, a number of heuristic techniques has been implemented (multi-resolution approach, multi-start, eyes' localization and preregistration [6] in order to find a reasonable starting point). The registration framework originally projected, implemented and thoroughly tested by the authors [6] has been lately redesigned and rewritten. Its general structure remains unchanged, but nowadays it is based on the Insight Toolkit (ITK) [3] library. Replacement of the proprietary framework with the popular open-source ITK platform fosters and accelerates its future development. ITK also offers dozens of ready to use and well documented algorithms for performing image registration and segmentation.

After the registration process, it is advisable to remove non-brain tissue from the images. Presence of non-brain tissue helps to achieve correct image registration but could be disruptive for the successive analysis steps. Brain extraction is relatively easy in CT images. In case of MR (Magnetic Resonance) images this step is not trivial. Numerous algorithms have been developed to perform it automatically. A survey of the most popular ones can be found in [7]. In the presented system, BET (Brain Extraction Tool) [8] has been utilised. It is accurate



**Fig. 1.** Fractal-dimension-based atrophy measure

enough and fast (processing time is usually shorter than 1 min. on a standard PC for a typical head MR dataset).

Because of fractal properties of many natural objects, fractal analysis is a reasonable choice in applications where natural objects are dealt with, including medical image processing and analysis. It is known that brain cortex images are self-similar in a way referred to as being a fractal, with a fractal dimension  $D = 2.60$  [9] (the results vary and depend on calculation method). It is also known that value of fractal dimension corresponds to brain atrophy level [10] (Fig. 1).

If  $A$  is the union of  $N_r$  non-overlapping copies of itself scaled down (or up) by a factor  $r$ , the fractal dimension is given by:

$$D = \frac{\log(N_r)}{\log(\frac{1}{r})}, \quad (1)$$

where  $1 = N_r r^D$ . In image processing and analysis it has to be estimated phenomenologically. A survey of fractal dimension calculation methods can be found in [11]. Different variations of box-counting methods are the most popular. It is relatively easy to calculate for many reasonably regular sets. In the simplest variant, a binary image is placed on a grid of square blocks. The number of blocks  $N_r$  occupied by a part of the image is then calculated. The procedure is repeated for various grid sizes ( $r$ ). It is expected that increasing the resolution of the grid,  $N_r$  should increase, too. The slope of linear regression of the  $\log(N_r)$  versus  $\log(1/r)$  is the fractal dimension estimation.

This approach has a relevant drawback. Images have to be binary ones, but MR image segmentation is not a trivial task. This process and selection of tissue to be segmented (white matter, grey matter, etc.) has a significant impact on the fractal dimension calculation result. The authors suggest using a variation of box-counting method, proposed by Sarkar and Chaudhuri (differential box-counting) [12]. It operates directly on grey scale images and does not depend on any special preprocessing scheme. An image of size  $M \times M$  is scaled down to  $s \times s$ . Then  $r = s/M$ . The 2D image is treated as a 3D image, where  $(x, y)$  denotes 2D position and  $z$  denotes a grey level. A column of boxes  $s \times s \times s'$  is obtained. If the total number of grey levels is  $G$ , then  $G/s' = M/s$ . If the

minimum and maximum grey levels of the image in the grid  $(i, j)$  fall in the box  $k$  and  $l$  respectively, then [12]

$$n_r = l - k + 1 \quad (2)$$

is a contribution of the grid  $(i, j)$  to  $N_r$ :

$$N_r = \sum_{i,j} n_r(i, j). \quad (3)$$

$N_r$  is calculated for various values of  $r$  as in simple box-counting. Fractal dimension  $D$  is then estimated from least square linear fit of  $\log(N_r)$  against  $\log(1/r)$ . It has been confirmed that this measure can be successfully used to classify normal and abnormal (atrophic) brain structures [13,14].

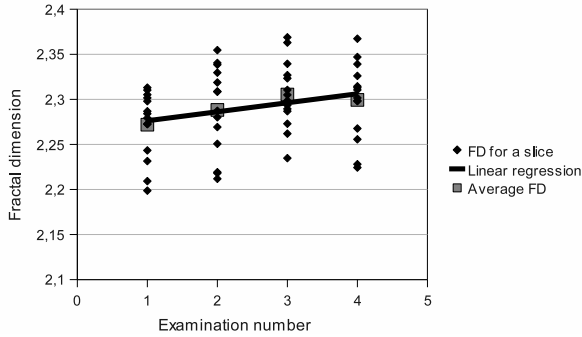
SIENA [15,16] is the next analysis tool (part of FSL library for MR image analysis [16,17]). It is able to perform two-time-point (longitudinal) analysis of brain change (volumetric loss of brain tissue). In particular, it can be used for quantitative estimation of atrophy level shift. Having performed tissue-type segmentation [18], perpendicular tissue edge displacement (between the two time points) is estimated at these edge points. Finally, the mean edge displacement is converted into a global estimate of percentage brain volume change between the two time points. SIENA (sienax tool) can be also used for total brain tissue volume estimation, from a single image. Detailed, technical information on SIENA package and the employed algorithms (including the MR segmentation procedure) can be found on the project's web page [19].

T1-weighted MR images of a human head were the main point of interest (though, other MR and CT images were also used). The images used for the tests come from two sources. The first one is the ELUDE collection (Efficient Longitudinal Upload of Depression in the Elderly) from the mBIRN Data Repository (mBDR, Project Accession Number 2007-BDR-6UHZ1) [20]. A MR scan of each subject was obtained every 2 years for up to 8 years. Multiple datasets of 30 randomly selected patients were used for the experiments. The second source was the Hospital of Ministry of Interior and Administration in Lublin (Poland). 19 subjects that were examined more than once during the last a few years were acquired from the hospital PACS (Picture Archiving and Communication System). Besides, numerous images of both normal and pathological subjects from single examinations were used.

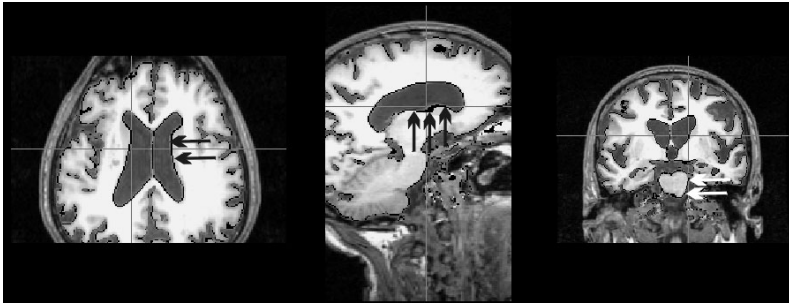
### 3 Results

The whole image processing and analysis process (image registration, brain extraction, segmentation, analytical procedures) is quite time-consuming. However, fractal dimension alone can be calculated very fast (less than 0.5s per single  $512 \times 512$  MR slice) using differential box-counting (described in the previous section).

Fig. 2 presents results of its calculation for quadruple MR examination of the same patient (with two-year-long intervals), on 14 centrally located axial slices



**Fig. 2.** Fractal dimension growth during brain atrophy progress (quadruple MR examination of the same, single patient with two-year-long intervals; 14 centrally located axial slices in the region of interest)



**Fig. 3.** Atrophic areas detected with SIENA (marked black), interesting places marked with black or white arrows

(the region of interest). The head images were not preprocessed at all. The slope of linear regression (thick line) of fractal dimension versus time corresponds to significant (according to a radiologist's opinion) atrophy progress. Such calculations were performed for all available datasets.

All images from the ELUDE collection were registered successfully (regardless of the software used: either ITK-based or FSL-based program). Two (out of nineteen) image pairs from Lublin hospital were problematic, due to presence of artefacts or extremely untypical (probably due to medical constraints) patient location inside MR scanner. Apart from intra-modal registration (MRI T1), CT-MRI registration was also executed if CT scans were available.

The brain extraction procedure was performed using BET (Brain Extraction Tool) [8]. Generally, it worked automatically. In sparse cases slight manual modifications of fractional intensity threshold were necessary. Then percentage brain volume change (PBVC) between corresponding MR T1-weighted images was calculated, and places with detected atrophy were marked (Fig. 3) in this

particular case estimated PBVC was about  $-1.4\%$ ), using SIENA package. The full processing pipeline for one image pair requires one hour or more, using a typical PC.

## 4 Discussion

As mentioned above, the whole process is time-consuming. On the other hand, it is advisable to obtain at least approximate image analysis result within a short time. Then it can be used for making an introductory diagnose or during screening assays (with large amounts of data). The proposed method of fractal dimension calculation seems to be a good candidate. It is fast, fully automatic and requires no preprocessing. It was performed both with and without brain extraction procedure. Interestingly, it has been observed that presence of non-brain tissue does not significantly disrupt the fractal dimension calculation, so the brain extraction procedure could be skipped. This method lets large amounts of data be roughly classified regarding atrophy level [13][14]. Potentially, it can be used to keep track of a disease progress (as shown in Fig. 2). Unfortunately, if two examinations of the same patient are not very time-distant or there are only subtle differences between them, this method is not sensitive enough. The fractal feature based method is promising but needs further investigation, using an extended image collection. The currently available one does not include enough images with apparent atrophic changes, to perform relevant statistical tests. It is still a problem to obtain a large enough, representative collection of multiple images of the same patients with atrophy progress, acquired at various times, for the research purpose.

Image registration is an especially valuable, but still seldom used tool for a doctor. Currently available algorithms (usually using various variants of maximization of mutual information and heuristics) make it possible to perform intra- or intermodal registration of standard medical datasets within reasonable time (5 – 15 minutes). Only a small part of images can be problematic, when using a carefully tuned registration procedure [6]. Usually the problem can be solved by setting a reasonable starting point.

The Brain Extraction Tool in most cases worked correctly, however sometimes it was necessary to alter the default values of the parameters. It is not useful as a standalone tool for a doctor, but is a necessary element of a head image processing system.

Atrophic changes detection performed with SIENA was especially appreciated by the radiologist. In case of the patient (with Alzheimer's disease diagnosed) whose head image is presented in Fig. 3, the time between two MR examinations was only 3 months. The atrophy progress was completely unseen when the image pair was inspected visually by an expert (even after a proper registration process). SIENA not only estimates the two-time-point percentage brain volume change, but also precisely marks brain tissue edge displacements. Information about specific brain parts that undergo atrophy is very diagnostically valuable.

## 5 Conclusions

Despite the existence of commercial image processing and analysis software, it still seems both possible and desirable to significantly support radiologists' job with the proposed framework. A vital part of information provided by medical images is hidden (and unavailable for a doctor inspecting images visually only) but can be extracted with appropriate algorithms. The presented algorithms' set for image registration (a necessary step for comparative analysis), brain extraction (required for further brain image analysis) and brain volume change estimation (SIENA) is mature enough to be used in clinical conditions. It is only necessary to combine the algorithms into a consistent processing system (with a suitable graphical user interface), compatible with hospital PACS. The main elements of the system are now being integrated and tested. An effort is made to decrease the processing time. Automated detection of dementia symptoms is still a challenge. Fractal-dimension-based classification is promising but still experimental. The authors are trying to construct a robust atrophy measure composed of both fractal and volumetric properties. Then it is planned to build an expert system for classification of atrophic changes, using also some external information (like demographic data).

## References

1. World Health Organization, International Statistical Classification of Diseases and Related Health Problems. 10th Revision (ICD-10), Chapter V Mental and behavioural disorders F00–F07 (2007)
2. New PET/CT center to be opened in Poland to reduce cancer surgeries (2008), <http://HealthImaging.com>
3. National Library of Medicine Insight Segmentation and Registration Toolkit (ITK) Documentation (2009), <http://www.itk.org/Doxygen/html/>
4. Viola, P., Wells, W.M.: Alignment by maximization of mutual information. *IJCV* 24(2), 137–154 (1997)
5. Mattes, D., Haynor, D.R., Vesselle, H., Lewellen, T.K., Eubank, W.: PET-CT image registration in the chest using free-form deformations. *IEEE Trans. on Medical Imaging* 22(1), 120–128 (2003)
6. Kuczyński, K., Mikołajczak, P.: Medical image registration a study of accuracy, performance and applicability of the procedure. *Polish Journal of Environmental Studies* 16(4A), 144–147 (2007)
7. Boesen, K., Rehm, K., Schapera, K., Stoltzner, S., Woods, R., Ldersc, E., Rottenberg, D.: Quantitative comparison of four brain extraction algorithms. *NeuroImage* 22(3), 1255–1261 (2004)
8. Smith, S.M.: Fast robust automated brain extraction. *Human Brain Mapping* 17(3), 143–155 (2002)
9. Majumdar, S., Prasad, R.: The fractal dimension of cerebral surfaces using magnetic resonance imaging. *Comput. Phys.*, 69–73 (1988)
10. Czarnecka, A., Siasidek, M.J., Hudyma, E., Kwaśnicka, H., Paradowski, M.: Computer-Interactive Methods of Brain Cortical Evaluation. In: Pietka, E., Kawa, J. (eds.) *Information Tech. in Biomedicine*, ASC 47, pp. 173–178. Springer, Heidelberg (2008)



11. Bisoi, A.K., Mishra, J.: On calculation of fractal dimension of images. *Pattern Recognition Letters* 22, 631–637 (2001)
12. Sarkar, N., Chaudhuri, B.B.: An efficient differential box-counting approach to compute fractal dimension of image. *IEEE Trans. Systems, Man, Cybernet.* 24(1), 115–120 (1994)
13. Kuczyński, K., Mikołajczak, P.: Magnetic Resonance Image Classification Using Fractal Analysis. In: Pietka, E., Kawa, J. (eds.) *Information Tech. in Biomedicine*, ASC 47, pp. 173–178. Springer, Heidelberg (2008)
14. Kuczyński, K., Buczko, O., Mikołajczak, P.: Fractal-dimension-based classification of radiological images. *Polish Journal of Environmental Studies* 17(3B), 198–202 (2008)
15. Smith, S.M., Zhang, Y., Jenkinson, M., Chen, J., Matthews, P.M., Federico, A., De Stefano, N.: Accurate, robust and automated longitudinal and cross-sectional brain change analysis. *NeuroImage* 17(1), 479–489 (2002)
16. Smith, S.M., Jenkinson, M., Woolrich, M.W., Beckmann, C.F., Behrens, T.E.J., Johansen-Berg, H., Bannister, P.R., De Luca, M., Drobnjak, I., Flitney, D.E., Niaz, R., Saunders, J., Vickers, J., Zhang, Y., De Stefano, N., Brady, J.M., Matthews, P.M.: Advances in functional and structural MR image analysis and implementation as FSL. *NeuroImage* 23(S1), 208–219 (2004)
17. Woolrich, M.W., Jbabdi, S., Patenaude, B., Chappell, M., Makni, S., Behrens, T., Beckmann, C., Jenkinson, M., Smith, S.M.: Bayesian analysis of neuroimaging data in FSL. *NeuroImage* 45, S173–S186 (2009)
18. Zhang, Y., Brady, M., Smith, S.: Segmentation of brain MR images through a hidden Markov random field model and the expectation maximization algorithm. *IEEE Trans. on Medical Imaging* 20(1), 45–57 (2001)
19. SIENA web page (2010), <http://www.fmrib.ox.ac.uk/fsl/siena/index.html>
20. mBDR Home Page (2009), <http://mbdr.nbirn.net/>

## Appendix: Acknowledgements

Radiological image data sources used for this study:

1. Biomedical Informatics Research Network (BIRN) Data Repository (<http://www.nbirn.net/bdr>), supported by grants to the BIRN Coordinating Center (U24-RR019701), Function BIRN (U24-RR021992), Morphometry BIRN (U24-RR021382), and Mouse BIRN (U24-RR021760). Testbeds funded by the National Center for Research Resources at the National Institutes of Health, U.S.A.
2. Hospital of Ministry of Interior and Administration in Lublin (Poland).

# Classification of Stabilometric Time-Series Using an Adaptive Fuzzy Inference Neural Network System

Juan A. Lara<sup>1</sup>, Pari Jahankhani<sup>2</sup>, Aurora Pérez<sup>1</sup>, Juan P. Valente<sup>1</sup>,  
and Vassilis Kodogiannis<sup>2</sup>

<sup>1</sup> Technical University of Madrid, School of Computer Science, Campus de Montegancedo,  
28660, Boadilla del Monte, Madrid, Spain

j.lara.torralbo@upm.es, {aurora, jpvalente}@fi.upm.es

<sup>2</sup> University of Westminster, School of Electronic and Computer Science,  
London HA1 3TP, United Kingdom  
{parij, kodogiv}@wmin.ac.uk

**Abstract.** Stabilometry is a branch of medicine that studies balance-related human functions. The analysis of stabilometric-generated time series can be very useful to the diagnosis and treatment balance-related dysfunctions such as *dizziness*. In stabilometry, the key nuggets of information in a time series signal are concentrated within definite time periods known as events. In this study, a feature extraction scheme has been developed to identify and characterise the events. The proposed scheme utilises a statistical method that goes through the whole time series from the start to the end, looking for the conditions that define events, according to the experts' criteria. Based on these extracted features, an Adaptive Fuzzy Inference Neural Network has been applied for the classification of stabilometric signals. The experimental results validated the proposed methodology.

## 1 Introduction

Stabilometry is the branch of medicine responsible for examining balance in human beings. Balance and dizziness disorders are probably two of the most common illnesses that physicians have to deal with. Around 30% of population suffers from any kind of dizziness disorder before reaching the age of 65; for older people, this pathologic symptom occurs in a more frequent rate, and it is responsible for people's falling. In order to examine balance, a device, called posturograph, has been used to measure the balance-related functionalities. The patient stands on a platform and completes a series of tests, as shown in Fig. 1. These tests have been designed to isolate the main sensorial, motor and biomechanical components that contribute to balance. Emphasis has been given to the evaluation of the capacity for each individual components as well as the overall components capacity. The posturograph generates a time-series signal, where the main information normally is confined to specific regions of the series, known as events [1]. Similarities exist also at many other domains. In seismography, for example, the regions of interest are when the time series indicates an earthquake, volcanic activity leading up to the quake or replications.

Initially, stabilometry was considered as a technique measuring only the balance of human beings under certain conditions [2] [3]. Many researchers have studied the effect of closed eyes on balance. [4] and [5]. These works confirmed that the condition of having the eyes closed affects balance due to the fact that balance has a strong visual component.



**Fig. 1.** Patient completing a test on a posturograph

Currently, stabilometry is also considered as a useful tool for diagnosing balance-related disorders like the Parkinson disease [6] or benign vertigo of childhood [7]. Regarding stabilometric data analysis, body sway parameters have been used for analysis balance-related functions [8], [9]. However, it appears that classic posturographic parameters, such as the measure of the sway of the centre of pressure [10] have failed in the detection of balance disorders [11]. The analysis of stabilometric time series using data mining techniques offers new possibilities. Recently, a new method has been developed for comparison of two stabilometric time-series [12]. This method calculates the level of similarity of two time-series and can be applied to compare either the balance of two patients or to study how the balance of one patient evolves with time. Stabilometry also plays an important role in the treatment of balance-related diseases. The NedSVE/IBV system [3] has been utilised for the development of a new method that assists in the rehabilitation of patients who have lost their balance [13].

With the continuously growing demand for models for complex systems inherently associated with nonlinearity, high-order dynamics, time-varying behaviour, and imprecise measurements, such as stabilometric time-series, there is a need for a relevant modelling environment. Efficient modelling techniques should utilise features/pertinent variables extracted from raw data and “transform” them in a highly representative dataset. The models should also be able to take advantage of the existing domain knowledge (such as a prior experience of human observers or operators) and augment it by available numeric data to form a coherent data knowledge modelling entity.

Fuzzy systems accept numeric inputs and convert these into linguistic values (represented by fuzzy numbers) that can be manipulated with linguistic IF-THEN rules and with fuzzy logic operations, such as fuzzy implication and composition rules of inference. However, at present there is no systematic procedure for the design of a fuzzy system. Usually the fuzzy rules are generated by converting human operators’ experience into fuzzy linguistic form directly and by summarizing the system behaviour (sampled input-output pairs) of the operators. During the last years, the

fuzzy neural network approach has gained considerable interest for solving real world problems, including modelling and control of highly complex systems, signal processing and pattern recognition [14].

In this paper, we will consider an Adaptive Fuzzy Inference Neural Network system (AFINN) for the classification of features extracted from stabilometric events. AFINN is made up of Gaussian-membership functions associated with local linear systems. The proposed fuzzy logic system is based on the Sugeno type modified with the introduction of an additional layer of output partitions and constructs its initial rules by clustering while the final fuzzy rule base is determined by competitive learning [15].

## 2 Data Recording

Throughout this research, a static Balance Master posturograph has been used. In a static posturograph, the platform on which the patient stands is static, i.e. does not move. The platform has four sensors, one at each of the four corners: right-front (RF), left-front (LF), right-rear (RR) and left-rear (LR). Each sensor records a datum every 10 milliseconds during the test. This datum is sent to the computer connected to the posturograph. The datum is the intensity of the pressure that the patient is exerting on that sensor. Data are recorded as multidimensional time-series.

The posturograph Balance Master can be used to run a wide range of tests according to a predefined protocol. This research study has focused on the Unilateral Stance (UNI) test that is the most useful for domain experts (physicians) in terms of output information. UNI test aims to measure how well the patient is able to keep his or her balance when standing on one leg with either both eyes open or both eyes closed for 10 seconds. The UNI test generates time-series signals containing events, that is, regions of special interest for experts in the domain. Next section describes the possible events appearing in the time series of UNI test and the features used to characterise these events. Both the events and their features were determined according to the physicians' criteria. The following cases are the four different conditions of UNI test:

- Left leg with Open Eyes: The patient is asked to hold still with his or her left leg on the platform while his or her right leg has to be lifted.
- Right leg with Open Eyes: The patient is asked to hold still with his or her right leg on the platform while his or her left leg has to be lifted.
- Left leg with Closed Eyes: The patient is asked to hold still with his or her left leg on the platform while his or her right leg has to be lifted.
- Right leg with Closed Eyes: The patient is asked to hold still with his or her right leg on the platform while his or her left leg has to be lifted.

Each condition is examined three times (trials) according to the medical protocol. The current research has been carried out on time series from a set of healthy volunteer people aged in the range of 14 to 38, including both genders and sport and non-sport people.

### 3 Method for the Identification and Feature Extraction of Events

The physicians are interested in studying the sway velocity (degrees/second) of the patient during the test. The relative absence of sway means stability while greater sway indicates less stability. The sway velocity throughout a whole test is computed as the average of the angular velocity of the patient between each pair of timestamps, according to Eq. 1. Since the UNI test lasts 10 seconds and the sensors measures a datum every 0.01 seconds, the number of timestamps is 1000.

$$SwayVelocity = \frac{\sum_{i=1}^{1000} AngularVelocity_i}{1000} \quad (1)$$

To compute the angular velocity it is enough to divide the angle  $\theta_i$  that the patient has moved between the timestamp  $i$  and the timestamp  $i+1$  by the time between two consecutive timestamps, that is, 0.01 seconds, as shown in Eq. 2.

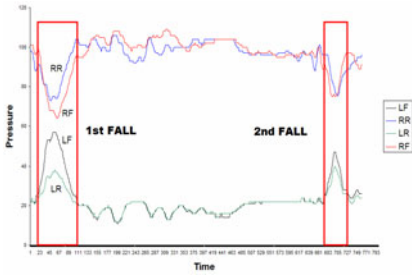
$$AngularVelocity_i = \frac{\theta_i}{0.01} \quad (2)$$

In addition to the sway velocity, physicians are interested in the number of events that occur and their more relevant features. There are two types of events in UNI test. The first one occurs when the patient loses balance and puts the lifted leg down onto the platform. This type of event is known in the domain as a fall. When there is a fall, the respective sensors for the lifted leg will register the pressure increase. Fig. 2 shows the time-series of a patient who has taken the UNI test. The curves at the top of the figure are the values recorded by the RR and RF sensors, that is, the right-leg sensors, the leg the patient was standing on. The curves at the bottom of the figure are the values recorded by sensors LR and LF, that is, the left-leg sensors, the leg that should be lifted. The pressures peaks generated when there is a fall event are highlighted. The features characterising the falls are as follows:

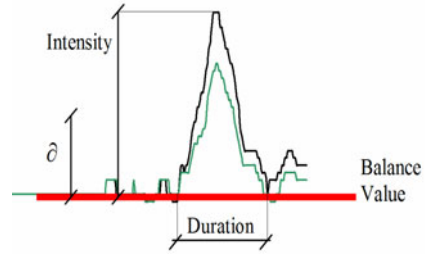
- **Duration:** It is the amount of time between the moment when the patient starts to lose balance and the moment when he or she is stable again, after falling.
- **Intensity:** It is the strength that the patient exerts on the platform when he or she falls down onto it.

The second type of event occurs when the patient losses balance but, before falling, he or she manages to recover upright position. This kind of event is known in the stabilometric domain as an imbalance. To identify and characterise these kinds of events the method proposed in [12] has been used. It is a statistical method that goes through the time series trying to find points where there is an event.

Regarding fall events, this method calculates the average value of the time-series related to the leg that must be lifted (bottom of Fig. 2). This average value represents the balance value as shown in Fig. 3. The method identifies points where there is a local maximum whose distance to the balance value is higher than a certain threshold ( $\delta$ ). That distance is precisely the intensity of the fall. The duration of the fall is then calculated by analysing the two intersections between the time series and the balance value line.



**Fig. 2.** UNI test time series, highlighting two events (falls)



**Fig. 3.** Fall event taken from a stabilometric time series

In order to classify stabilometric time series, a set of balance-related features have been extracted according to the experts’ criteria, as shown in Table 1. These features are as follow:

- Sway Velocity.
- Number of Imbalances.
- Number of Falls.
- Total Duration: It is the sum of the durations of all the falls contained in a time series.
- Maximum intensity: It is the maximum value of intensity of the falls contained in a time series.

These features have been chosen due to the fact that they represent how stable the patient is in a faithful way. The sway velocity is a good overall parameter to study patients’ balance. On the other hand, the number of imbalances, the number of falls and the total duration of falls give a clear idea of the amount of time that the patient was unstable during the test.

**Table 1.** Sample of extracted features

| Time Series | Sway Velocity | Number of falls | Number of imbalances | Total Duration | Maximum Intensity |
|-------------|---------------|-----------------|----------------------|----------------|-------------------|
| 1           | 1.92          | 3               | 2                    | 31             | 15                |
| 2           | 1.18          | 0               | 2                    | 0              | 0                 |
| ...         | ...           | ...             | ...                  | ...            | ...               |
| 56          | 1.85          | 0               | 1                    | 0              | 0                 |

## 4 AFINN Architecture

In this paper, we will consider an Adaptive Fuzzy Inference Neural Network system (AFINN) which is made up of Gaussian-membership functions associated with local linear systems. The proposed fuzzy logic system is based on the Sugeno type modified with the introduction of an additional layer of output partitions. Unlike the

ANFIS system, in which the number of local linear systems is same as that of the number of rules, AFINN provides a means of controlling the growth of the number of local linear systems when the order of the system under consideration increases, so that least-squares estimation can be applied without performance degradation. A clustering algorithm is applied for the sample data in order to organise feature vectors into clusters such that points within a cluster are closer to each other than vectors belonging to different clusters. Then fuzzy rule base is created using results obtained from this algorithm. The fuzzy implication of the fuzzy system is based on fuzzy partitions of the input space directly rather than fuzzy partitions of each dimension of the input space. Thus the membership functions considered in the proposed system are multi-dimensional membership functions. In this sense, there is a similarity with the construction of Gaussian centres in Radial Basis Function networks (RBFN). Since the input space is considered to be partitioned instead of each dimension of the input space, the number of rules can be small and hence the number of local linear systems is also small. In addition, competitive learning technique is applied to locate space partitions according to the clustering of the fuzzy rules at the beginning of training.

The architecture of the proposed neuro-fuzzy network shown in Fig 4 consists of six layers. The first two layers  $L_1$  and  $L_2$  correspond to IF part of fuzzy rules whereas layers  $L_4$  and  $L_5$  contain information about THEN part of these rules and perform the defuzzification task.

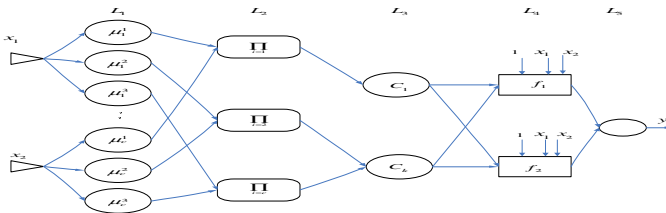


Fig. 4. Structure of AFINN system

The most important problem in fuzzy systems is to find fuzzy rules. Some methods can generate fuzzy rules from input-output pairs [16]. In this paper, the fuzzy rule base is derived using results obtained from a clustering algorithm. The clustering algorithm we apply in this paper at layer  $L_2$  consists of two stages. In the first stage the method similar to Learning Vector Quantisation (LVQ) algorithm generates crisp  $c$ -partitions of the data set. The number of clusters  $c$  and the cluster centres  $v_i, i = 1, \dots, c$ , obtained from this stage are used by FCM (Fuzzy  $c$ -means) algorithm in the second stage. In layer  $L_3$  a mapping between the rule layer and the output layer is performed by a competitive learning process. The local linear systems at  $L_4$  are associated with each term of layer  $L_3$  rather than that of rule base layer  $L_2$ . Thus the size of required matrices for least-squares estimation is considered to be much smaller [15].

## 5 Discussion of Results

Automated diagnostic systems aim to enhance the ability to detect pathological structures in medical examinations and to support evaluation of pathological findings during

the diagnostic procedure. Most techniques developed for automated stabilometric data analysis have focused on the study of the centre of pressure of the patient. However, balance-related events (falls and imbalances) contain useful information for the physicians. In this research, the proposed AFINN network has been implemented for stabilometric time-series classification, employing the most significant features of the events contained in UNI time series. As explained in section 2, the UNI test consists of four different conditions. This case study focused on the second trial of Left leg with Closed Eyes condition. The first and third trials have not been considered because the first trial contains noise and during the third trial the patient has already learnt how to be stable. In this study, 56 stabilometric time series with 1000 timestamps have been used. The data set was divided into two classes according to the gender of patients: MALE and FEMALE. 38 out of the 56 time series belong to male patients while 18 belong to female patients. 5 balance-related features were extracted from each time series.

In our experiments, the training data set was used to train the AFINN model, whereas the testing data set was used to verify the accuracy and effectiveness of the trained AFINN model for classification of the 2 classes of stabilometric time series. The proposed scheme has high classification accuracy with within 5 epochs. The results of the proposed classifier, using 10 different training sets are illustrated at Table 2.

**Table 2.** AFINN performance for stabilometric time series

| System | Rules or Nodes | Epoch | Class 1 (Female) | Class 2 (Male) |
|--------|----------------|-------|------------------|----------------|
| AFFIN  | 7/4            | 5     | 95%              | 94.3%          |

The clustering fuzzification part resulted in 7 rules, while after the competitive layer, the rules were reduced to 4, which resulted in fewer consequent parameters at the defuzzification layer.

## 6 Conclusions

Fuzzy set theory plays an important role in dealing with uncertainty when making decisions in medical applications. The usage of fuzzy logic enabled us to use the uncertainty in the classifier design and consequently to increase the credibility of the system output. This research study presented a neural network implementation of the AFFIN and its application on the classification of stabilometric time series. The proposed network was trained and tested with the extracted features using a statistical ad-doc method for the identification and the characterisation of events in stabilometric time series. The simulation results reveal an almost perfect performance.

## References

1. Povinelli, R.: Time Series Data Mining: identifying temporal patterns for characterization and prediction of time series, PhD Thesis, Milwaukee (1999)
2. Romberg, M.H.: Manual of the Nervous Disease of Man, pp. 395–401, Sydenham Society, London (1853)



3. Baron, J.B., Bobot, J., Bessineton, J.C.: Statokinesimetric. *Presse Med.* 64, 36: 863 (1956)
4. Paulus, W.M., Straube, A., Brandt, T.: Visual stabilization of posture: physiological stimulus characteristics and clinical aspects. *Brain* 107, 1143–1163 (1984)
5. Gagey, P., Gentaz, R., Guillaumon, J., Bizzo, G., Bodot-Braeard, C., Debruille, B.C.: Normes 85, Association Française de Posturologie, Paris (1988)
6. Ronda, J.M., Galvañ, B., Monerris, E., Ballester, F.: Asociación entre Síntomas Clínicos y Resultados de la Posturografía Computerizada Dinámica. *Acta Otorrinolaringol Esp.* 53, 252–255 (2002)
7. Barona, R.: Interés clínico del sistema NedSVE/IBV en el diagnóstico y valoración de las alteraciones del equilibrio. *Biomechanics Magazine of the Institute of Biomechanics of Valencia (IBV)* (February 2003)
8. Rocchi, L., Chiari, L., Cappello, A.: Feature selection of stabilometric parameters based on principal component analysis. In: *Medical & Biological Engineering & Computing 2004*, vol. 42 (2004)
9. Demura, S., Kitabayashi, T.: Power spectrum characteristics of body sway time series and velocity time series of the center of foot pressure during a static upright posture in pre-school children. *Sport Sciences for Health* 3(1), 27–32 (2008)
10. Diener, H.C., Dichgans, J., Bacher, M., Gompf, B.: Quantification of postural sway in normals and patients with cerebellar diseases. *Electroenc. and Clin. Neurophysiol.* 57, 134–142 (1984)
11. Corradini, M.L., Fioretti, S., Leo, T., Piperno, R.: Early Recognition of Postural Disorders in Multiple Sclerosis Through Movement Analysis: A Modeling Study. *IEEE Transactions on Biomedical Engineering* 44(11) (November 1997)
12. Lara, J.A., Moreno, G., Perez, A., Valente, J.P., Lopez-Illescas, A.: Comparing Posturographic Time Series through Event Detection. In: *21st IEEE International Symposium on Computer-Based Medical Systems, CBMS*, pp. 293–295 (2008)
13. Peydro, M.F., Vivas, M.J., Garrido, J.D., Barona, R.: Procedimiento de rehabilitación del control postural mediante el sistema NedSVE/IBV. *Biomechanics Magazine of the Institute of Biomechanics of Valencia (IBV)*, Ed. (January 2006)
14. Jang, J.-S.R.: ANFIS: Adaptive-Network-based Fuzzy Inference Systems. *IEEE Trans. on Systems, Man and Cybernetics* 23, 665–685 (1993)
15. Jahankhani, P., Revett, K., Kodogiannis, V., Lygouras, J.: Classification Using Adaptive Fuzzy Inference Neural Network. In: *Proceedings of the Twelfth IASTED International Conference Artificial Intelligence and Soft Computing (ASC 2008)*, Palma de Mallorca, Spain, September 1-3 (2008)
16. Sonbol, A., Fadali, M.S.: A new approach for designing TSK fuzzy systems from input-output data. In: *Proc. of the American Control Conference*, vol. 2, pp. 989–994 (2002)

# An Approach to Brain Thinker Type Recognition Based on Facial Asymmetry

Piotr Milczarski, Leonid Kompanets, and Damian Kurach

University of Lodz, Pomorska str. 149, 90-236 Lodz, Poland  
{piotr.milczarski,lkompanets}@uni.lodz.pl, damiankurach.gmail.com

**Abstract.** In the paper, a hypothesis of determining brain hemisphere dominance from a frontal face image is investigated. Behind this hypothesis, a face is considered as somatic/psychological background. A biometrical method based on person facial asymmetry characteristics is used. The method includes finding a proper vertical face axis and applying new asymmetry measures. Experimental observations confirm that there exists a correlation between the left/right brain hemispheres mutual functioning and a human face asymmetry. Moreover, examples show that the indication of the facial asymmetry is consistent with information about the hemisphere dominance. The presented method has been tested on color images of individuals with previously examined brain types.

**Keywords:** biometrics; face asymmetry; brain hemisphere dominance; asymmetry measure; facial features; vertical face axis.

## 1 Introduction

A human face asymmetry is a personality characteristic and has been known as a factor for face recognition [6], [11], [10], face attractiveness determination [16], [2], and face expression recognition [11], etc. Currently, it is assumed that facial asymmetry is a result of a state of the brain hemispheres functioning [1]. This means that brain functionality is mapping directly on a facial appearance.

There are known 16 psychological types based on the theory of C. Jung [13], [5]. The type is a hypothetical concept that refers to behavior traits; it is revealed through subjective questionnaires done by Meyers-Briggs or D. Keirsey. In these models to describe personalities there were introduced 4 coordinates in the space of personality observation: first - extravert (E) or introvert (I), second - iNtuitive (N) or sensitive (S), third - thinking (T) or feeling (F), fourth - judging (J) or perceiving (P). It can be seen that the second coordinate refers to cognitive features (how person's sensors work), and the third one - to mental features (how person's mind works). The corresponding personality temperaments by D. Keirsey [5] are: SJ - organizer, SP - mediator, NT - seer, NF - a catalyst for group activity.

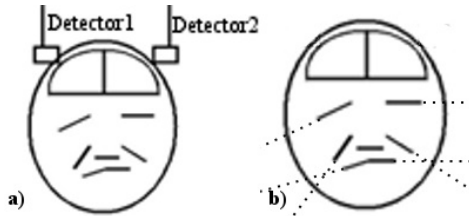
The authors put forward the hypothesis that any personality psychological types can be automatically identified on the basis of its frontal facial image using biological/psychological information or knowledge. There are some works [3], [4] that refer to and follow the idea presented in authors' previous papers,

e.g. [6]. Now, we are interested in developing a method to expand the usage of the facial asymmetry biometric and allow to recognize automatically the dominating hemisphere from a frontal facial image. This is the first step of our research in this direction.

The paper is organized as follows. In Section 2, we describe the existing, invasive method of the bilateral, left-brain and right-brain thinkers determination by Anuashvili [1]. The facial asymmetry measure algorithm is presented in Section 3. Our method of the hemisphere dominance determination from a frontal facial image and experimental results are described in Section 4. In Section 5, conclusions and future work are outlined.

## 2 Brain Hemisphere Dominance Determination by Anuashvili

According to the Anuashvili’s work [1] it is established that a human being psychological state is changing in two-dimensional  $L-I$  and  $S-D$  coordinates, where  $L$  means Logic (left-sided),  $I$ -Intuition (right-sided),  $S$ -Stability and  $D$ -Destability. The values  $L$  and  $I$  show the dominance of a given hemisphere, and are obtained by calculating  $\Delta A = A_l - A_r$ , defined as the difference between the signal amplitude of the left  $A_l$  and right  $A_r$  hemisphere waves (recorded by detectors shown in Fig.1a).



**Fig. 1.** a) Anushvili’s method of measuring hemispheres waves; b) illustration of the phase portrait based on the Anuashvili’s method [1]

In the coherent situation, the circular frequency of vibrations, respectively, in left,  $\omega_l$  and right,  $\omega_r$  hemispheres are equal ( $\omega_l = \omega_r$ ). the intensity of the coherent interference  $I_C$  between the hemispheres can be derived from

$$I_C = A_l e^{i(\omega_l t + \phi_l)} * A_r e^{i(\omega_r t + \phi_r)} = A_l A_r e^{i\Delta\phi} , \tag{1}$$

where  $\Delta\phi = \phi_l - \phi_r$  denotes the difference between initial oscillation phases of the left,  $\phi_l$  and the right,  $\phi_r$  hemispheres. The degree of coherence can be calculated by

$$C = \frac{\pi}{T} \int_{t_o+T}^{t_o} A_l A_r e^{i\Delta\phi} dt , \tag{2}$$

where  $T$  is the average time of the measurement, and  $C \in [-\pi, \pi]$ .

The values of *Stability* and *Destability* are calculated using equations (1) and (2). The personality degree of the harmony,  $H$ , is calculated by the formula

$$H = \sin\left(\frac{C}{2}\right) \left[ 1 + \left( \frac{\sin\left(\frac{\Delta A}{C}\pi\right)}{\frac{\Delta A}{C}\pi} \right)^2 \right] \quad (3)$$

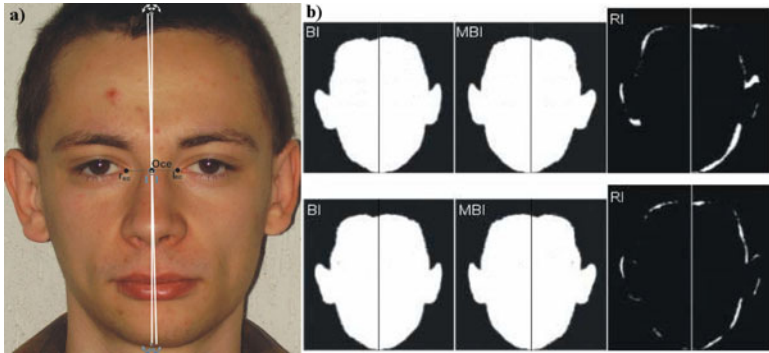
This approach needs direct expensive detectors of the electromagnetic brain waves as well as unique technologies during measuring parameters of the brain waves [1]. What is interesting a video-computer system has been designed for psycho-diagnostics and psycho-correction that uses the state of the phase portrait [14] (Fig.1b).

### 3 Algorithm of the Facial Asymmetry Measure

We have used the normalized *RGB* color space representation to detect face on an image and gradient intensity information to localize eyes position [9]. The idea of finding the possible iris edge by using information of region average pixel intensity is the original one. After edge candidates points estimation, the circle of iris diameter,  $Dr$ , and center,  $S$ , are extracted. This is a significant part of the method because the extracted size of the iris diameter is used to introduce the absolute normalization unit called by the authors [*Muld*] (if targeted to the diameter expressed in pixels) or [*Muld*<sup>2</sup>] (if targeted to the area, expressed in *pixel*<sup>2</sup>) [8]. This procedure gives possibility to map a personal face image to real dimension. The normalization unit has been based on a Muldashev [14] observation that after 4-5 years of our life, the diameter of non-transparent part of iris of any person is equal to  $10 \pm 0.56$  mm.

In the next step of the proposed method uniqueness of eyelid contours is achieved by means of approximation with three degree polynomials. The coordination points of upper and lower eyelids are detected based on pixel's illumination variation and unique mapping of inner and outer eyes corners. After the facial features extraction, our algorithm of the face asymmetry measure is applied. The algorithm includes automatic selection of facial asymmetry regions based on the proper vertical axis, and the facial size normalization using the *Muld* unit.

The problem of finding the facial proper vertical axis and values of asymmetry measures was previously described in [7], [8]. The facial axis is rotated around the anthropocentric point, *Oec* (center between inner eyes' corners), that minimizes the *Area Asymmetry* (*AAs*) measure of facial silhouette. The value asymmetry measure is calculated as the difference between the *Binary Image* (*BI*) and its *Mirrored Binary Image* (*MBI*). The binary image, *BI*, represents facial silhouette and is obtained from a frontal input image using the Otsu threshold algorithm [15] and morphological operations. The minimal values of *AAs* define the proper axis. Further, we present significant improvement of the method by introducing two new measures,  $AAs^+$ ,  $AAs^-$ , of appropriate facial regions. The values  $AAs^+$ ,  $AAs^-$  are calculated as the difference of the left



**Fig. 2.** Graphic illustration of determining the facial asymmetry: a) input image with two candidate axis ( $0^\circ$  and  $1.5^\circ$ ); b) BI binary images, MBI mirrored binary images, and result images, *RI*; for  $0^\circ$  axis (upper images) and for  $1.5^\circ$  axis (lower images)

and right parts (defined by the vertical axis) of the facial binary images (Fig.2 b). The values  $AA_s^+$  and  $AA_s^-$  represent the right-sided and left-sided facial asymmetry, respectively.

The *AA\_sMeasure* algorithm, applied in order to calculate the values of the measures  $AA_s$ ,  $AA_s^+$ ,  $AA_s^-$  in an adaptive way, can be presented as follows:

*Input data:* *Oec* - point coordinates,  $\phi[^\circ]$  - rotation angle, *BI* - silhouette's binary image, *Dr* - right iris diameter.

*Output data:*  $AA_s$ ,  $AA_s^+$  and  $AA_s^-$  values.

*Step 1.* Define the vertical axis,  $FA(\phi)$ , on a binary facial silhouette's image, *BI*, based on coordinates point, *Oec*, and the value of angle  $\phi[^\circ]$  (see *BI* images in Fig.2b)

*Step 2.* Create a vertically mirrored binary image, *MBI*, based on the information about *BI* image and the axis  $FA(\phi)$  (see *MBI* images in Fig.2b)

*Step 3.* Calculate  $AA_s^+$ ,  $AA_s^-$  values as the difference between the right and left side of *BI*'s and *MBI*'s images (see *RI* images in Fig.2b)

*Step 4.* Use the  $AA_s^+$  and  $AA_s^-$  values to determine the person's asymmetry type.

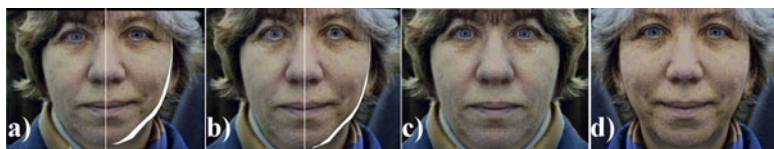
## 4 Results Concerning Brain Hemisphere Dominance and Facial Asymmetry

We take a color image of a person's face (for example, produced by a web-camera) as an input signal, and the program determines the facial symmetry type. We believe that the computed asymmetry types closely correspond to the brain thinker types i.e., bilateral, right-brain or left-brain thinker. If our assumption is true, the dominance of the right asymmetry means that the person on the image is right-brained thinker and the left asymmetry dominance determines

the left-brained one. When the asymmetry is balanced it means that the person is the bilateral one.

To objectively examine our method, we compare the obtained results of persons' asymmetry types with the psychological types described by Anuashvili's [1]. In the first step of the comparison we choose as the base axis this one calculated from the Anuashvili's composites, i.e. synthesized images from left-left and right-right parts of a face (see Figs. 3-5). Construction of the composite was described in [6]. In our opinion the Anuashvili's method [1] of determining axis is based on the *intuitive* technique of choosing the vertical axis ( i.e. *visually/manually*). In our method, originally described in [8], the vertical axis is the one determined by the automatic adaptation procedure, which is precise.

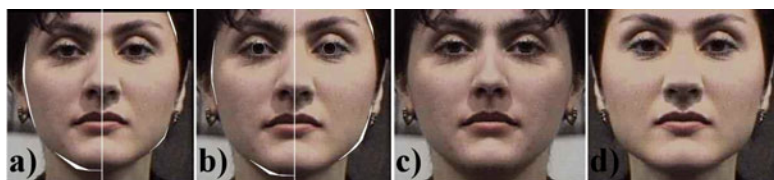
After the basis axis is determined, we rotate it around the anthropological point, *Oec*. The axis divides the face into two parts. For each part we build the mirrored images for the left and right sides. Then, we create two precise facial composites by connecting each part with its mirrored image part.



**Fig. 3.** Person with dominance of the left hemisphere: a) image with axis  $0^\circ$  and  $AAs=1.04 Muld^2$ , b) image with axis  $-1^\circ$  and  $AAs=0.5 Muld^2$ , c) right-sided, d) left-sided composite



**Fig. 4.** Person with dominance of the right hemisphere: a) image with axis  $0^\circ$  and  $AAs=3.79 Muld^2$ , b) image with axis  $+1^\circ$  and  $AAs=2.49 Muld^2$ , c) right-sided, d) left-sided composite



**Fig. 5.** Person with no dominance (bilateral): a) image with axis  $0^\circ$  and  $AAs=1.15 Muld^2$ , b) image with axis  $+1^\circ$  and  $AAs=0.87 Muld^2$ , c) right-sided, d) left-sided composite

Figures 3-5 present the input images of representative persons with previously examined brain type, for cases of left brain thinker, right brain thinker and bilateral thinker, respectively. Each figure contains images with marked (in white) regions of the asymmetry for two axes, the base one  $0^\circ$  (Fig. 3-5a) and the proper one, automatically obtained by the *AAsMeasure* algorithm (Fig.3-5b). The Fig.3-5c) and d) present the right and left composite obtained for the proper axis.

**Table 1.** Results of asymmetry measure for the left-brain thinker

| $\phi[^\circ]$ | <i>AAs</i> <sup>-</sup> |                              | <i>AAs</i> <sup>+</sup> |                              | <i>AAs</i>       |                              |
|----------------|-------------------------|------------------------------|-------------------------|------------------------------|------------------|------------------------------|
|                | [ <i>Pixel</i> ]        | [ <i>Muld</i> <sup>2</sup> ] | [ <i>Pixel</i> ]        | [ <i>Muld</i> <sup>2</sup> ] | [ <i>Pixel</i> ] | [ <i>Muld</i> <sup>2</sup> ] |
| -2             | 205                     | 0.31                         | 741                     | 1.14                         | 946              | 1.45                         |
| * -1           | <b>261</b>              | <b>0.40</b>                  | 69                      | 0.10                         | 330              | 0.50                         |
| 0              | 676                     | 1.04                         | 0                       | 0.00                         | 676              | 1.04                         |
| 1              | 1682                    | 2.59                         | 0                       | 0.00                         | 1682             | 2.59                         |
| 2              | 4787                    | 7.37                         | 80                      | 0.12                         | 4867             | 7.49                         |

**Table 2.** Results of asymmetry measure for the right-brain thinker

| $\phi[^\circ]$ | <i>AAs</i> <sup>-</sup> |                              | <i>AAs</i> <sup>+</sup> |                              | <i>AAs</i>       |                              |
|----------------|-------------------------|------------------------------|-------------------------|------------------------------|------------------|------------------------------|
|                | [ <i>Pixel</i> ]        | [ <i>Muld</i> <sup>2</sup> ] | [ <i>Pixel</i> ]        | [ <i>Muld</i> <sup>2</sup> ] | [ <i>Pixel</i> ] | [ <i>Muld</i> <sup>2</sup> ] |
| -2             | 381                     | 0.59                         | 4558                    | 7.02                         | 4939             | 7.61                         |
| -1             | 416                     | 0.64                         | 3090                    | 4.76                         | 3506             | 5.40                         |
| 0              | 554                     | 0.85                         | 1908                    | 2.94                         | 2462             | 3.79                         |
| * 1            | 757                     | 1.17                         | <b>860</b>              | <b>1.32</b>                  | 1617             | 2.49                         |
| 2              | 1498                    | 2.30                         | 232                     | 0.36                         | 1730             | 2.66                         |

**Table 3.** Results of asymmetry measure for the bilateral thinker

| $\phi[^\circ]$ | <i>AAs</i> <sup>-</sup> |                              | <i>AAs</i> <sup>+</sup> |                              | <i>AAs</i>       |                              |
|----------------|-------------------------|------------------------------|-------------------------|------------------------------|------------------|------------------------------|
|                | [ <i>Pixel</i> ]        | [ <i>Muld</i> <sup>2</sup> ] | [ <i>Pixel</i> ]        | [ <i>Muld</i> <sup>2</sup> ] | [ <i>Pixel</i> ] | [ <i>Muld</i> <sup>2</sup> ] |
| -2             | 202                     | 0.31                         | 1835                    | 2.83                         | 2037             | 3.14                         |
| -1             | 158                     | 0.24                         | 1197                    | 1.84                         | 1355             | 2.08                         |
| 0              | 579                     | 0.89                         | 170                     | 0.26                         | 749              | 1.15                         |
| * 1            | <b>348</b>              | <b>0.53</b>                  | 219                     | 0.34                         | 567              | 0.87                         |
| 2              | 690                     | 1.06                         | 82                      | 0.12                         | 772              | 1.18                         |

Tables 1-3 correspond to Figures 3-5, respectively and show the results of the area asymmetry measures *AAs* for 5 axes. The base axis is rotated left and right according to angles  $-2^\circ$ ,  $-1^\circ$ ,  $+1^\circ$ ,  $+2^\circ$  (the + sign means clockwise axes rotation) around the anthropological point. The values of *AAs*, *AAs*<sup>+</sup> and *AAs*<sup>-</sup> are given in [*Pixel*] and [*Muld*<sup>2</sup>] units. We show the results for determining the vertical axis with the accuracy  $1^0$  that is enough to illustrate the idea of the proposed asymmetry measures. The proper vertical axis is indicated by the lowest value of the asymmetry measure *Aas*; symbol (\*), in Tables 1-3, denotes the case of minimal *AAs* (see the \*-lines). With the gray colour we show the

results for the  $0^\circ$  axis. With bold letters we highline the bigger values of the sided asymmetry.

Our adaptive method of the proper axis determination balances the difference between the  $AA_s^+$  and  $AA_s^-$  values of asymmetry (see rows indicated by (\*) in Tables 1-3). The minimal value of the asymmetry  $AA_s$  for the left-brain thinker (see Fig.3 and Table 1) is equal to  $0.5 Muld^2$ . It is a small value which means that the person is close to the bilateral one. The value  $AA_s^-$  equals  $0.4 Muld^2$  and is bigger in comparison with  $AA_s^+$ . The bigger left side asymmetry indicates this person as the left-brain thinker type. For the person with right hemisphere dominance (see Fig.4 and Table 2) the minimal value of facial asymmetry measure equals  $2.49 Muld^2$ . The difference between  $AA_s^-$  and  $AA_s^+$  is rather small but with majority of  $AA_+$  value. This indicates that the person is classified as right-brained. The results obtained for the person with no dominance of a hemisphere (see Fig.5 and Table 3) show that  $AA_s$  equals  $0.87 Muld^2$  while  $AA_s^+$  equals  $0.34 Muld^2$  and  $AA_s^-$  equals  $0.53 Muld^2$ . The small difference between the sided asymmetry measures and the asymmetry region locations classify this person into bilateral thinker. When we consider the values obtained by  $0^\circ$  axis (the base one), those differences between the left and right asymmetry become even more visible.

## 5 Conclusions

The innovatory results of the computational solutions of the person's brain hemisphere dominance determination from a frontal facial image have been considered. All of the method's procedures are automatic, and include biometrics solutions such as face detection, eye localization, iris extraction, inner eye corners finding and a facial silhouettes determination. For the proposed method the measures ( $AA_s$ ,  $AA_s^+$  and  $AA_s^-$ ) of the facial silhouette asymmetry are introduced.

The asymmetry measure  $AA_s$  is used in the special adaptation procedure for precise determination of the proper facial vertical axis. The proper axis as well as the values  $AA_s^+$  and  $AA_s^-$  are the sufficient determinants of the facial asymmetry type. Moreover, those precise values are crucial to get information about correlation between the facial asymmetry type and the brain hemisphere domination. What is important, they are invariant to facial expression and rotation.

The results confirm that the facial asymmetry is important and informative. In addition, it confirms that types of the facial asymmetry correspond directly to the thinker types (the bilateral, left-brain and right-brain thinker types are determined by the model of the coherent hemisphere functioning [1]). Furthermore, in our method we can objectively compute (from a frontal facial image) a precise appropriate person's characteristics based on the introduced asymmetry measures. Those characteristics can be used as biometric in person's identification/authentication as well as other applications concerning psychological types determination.

In authors' opinion, further research in the field of biometrics/psychological information, may result in a pioneer chance to combine computer methods with



non-traditional computing psychological methods [5], [13]. For example the information about the brain hemisphere dominance can be directly used in the development of new directions in e-learning.

In order to work out a live-realistic decision about person's psychological type, it is worth to design fuzzy logic system-adviser with appropriate experts "if-then" rules (for example, like [12]).

## References

1. Anuashvili, A.N.: Fundamentals of Objective Psychology, 5th edn., Intern. Institute of Control, Psychology and Psychotherapy, Warsaw-Moscow (2008)
2. Chen, A.C., German, C., Zaidel, D.W.: Brain Asymmetry and Facial Attractiveness. *Neuropsychologia* 35(4), 471–476 (1997)
3. Kamenskaya, E., Kukharev, G.: Recognition of psychological characteristics from face. *Methods of Applied Informatics. Pol. Acad. Sci.* 1, 59–73 (2008)
4. Kamenskaya, E., Kukharev, G.: Some aspects of automated psychological characteristics recognition from the facial image. *Methods of Applied Informatics. Pol. Acad. Sci.* 2, 29–37 (2008)
5. Keirsej, D., Bates, M.: Please Understand Me: Character and Temperament Types. Prometheus Nemesis, Del Mar, CA (1984)
6. Kompanets, L.: Biometrics of Asymmetrical Face. In: Zhang, D., Jain, A.K. (eds.) ICBA 2004. LNCS, vol. 3072, pp. 67–73. Springer, Heidelberg (2004)
7. Kompanets, L.: Facial Composites, Ophthalmic Geometry Pattern and Based on Stated Phenomena the Test of Person/Personality Virtuality/Aliveness. In: 7th Intern. Conf. on Intelligent Systems Design and Applications, pp. 831–836. IEEE Computer Society Press, Los Alamitos (2007)
8. Kompanets, L., Kurach, D.: On Facial Frontal Vertical Axes Projections and Area Facial Asymmetry Measure. *Intern. J. of Computing, Multimedia and Intelligent Techniques* 3(1), 61–88 (2007)
9. Kurach, D., Milczarski, P.: Test of Aliveness/Virtuality Based on Ophthalmogeometry and Facial Asymmetry Characteristics. *Polish Journal of Environmental Studies* 17(4C), 497–502 (2008)
10. Liu, Y., Schmidt, K., Cohn, J., Mitra, S.: Facial asymmetry quantification for expression invariant human identification. *Computer Vision and Image Understanding* 91, 138–159 (2003)
11. Liu, Y., Weaver, R.L., Schmidt, K., Serban, N., Cohn, J.: Facial Asymmetry: A New Biometric. The Robotic Institute of Carnegie Melon Univ. (2001)
12. Martin, M.A., Mendel, J.M.: Flirtation, a Very Fuzzy Prospect: a Flirtation Advisor (1995)
13. Myers-Briggs, I., Myers, P.: Gifts Differing: Understanding Personality Type. Davies-Black Publishing (1995)
14. Muldashev, E.R.: Whom Did We Descend From? OLMA-Press, Moscow (2002)
15. Otsu, N.: A threshold selection method from gray-level histograms. *IEEE Trans. Systems, Man and Cybernetics* 9, 62–66 (1979)
16. Reis, V.A., Zaidel, D.W.: Functional Asymmetry in the Human Face: Perception of Health in the Left and Right Sides of the Face. *Laterality* 6(3), 225–231 (2001)

# Application of C&RT, CHAID, C4.5 and WizWhy Algorithms for Stroke Type Diagnosis

Igor S. Naftulin and Olga Yu. Rebrova

Research Center of Neurology, Moscow, Russia  
olga.rebrova@neurology.ru

**Abstract.** Four algorithms of data mining (C&RT, CHAID, C4.5 and WizWhy) were applied to produce rules for classification of three types of stroke on 298 cases used for learning and testing. The C&RT, CHAID algorithms did not give acceptable results of the classification. The system See5 was able to give low classification error in the mode of constructing a decision tree with decisions amplification in combination with fuzzy thresholds. Unfortunately, the rule sets obtained on the training samples, in test mode showed unsatisfactory results. WizWhy system showed acceptable accuracy, but practical use of generated rules is rather complicated.

## 1 Introduction

Problem of stroke diagnosis is important because of

- Stroke is the 3rd leading cause of mortality from noninfectious diseases
- Mortality from stroke in Russia is one of the highest in the world
- The incidence of stroke in young population increases
- Mortality in stroke is about 35-40%
- 62% of stroke sufferers are handicapped

There are three types of stroke: ischemic, about 75% of cases, hemorrhagic - 20% of cases, subarachnoid hemorrhage - 5% of cases. Treatments of the different types of stroke differ greatly, and therefore exact differential diagnosis of the stroke type is important.

Lack of access to brain imaging in Russia and many other countries [1, 2], and not very high agreement of interpretation of its results [3-7] encouraged to explore different ways to support medical decision-making in the differential diagnosis of stroke types. Earlier [8] we developed neural network algorithms with high diagnostic accuracy - 98%. However, we are interested in other methods for constructing computer diagnostics algorithms, first of all methods of data mining [9-13], in particular, those algorithms for constructing decision trees which are able to process data with missing values.

Decision trees is the way of representing the rules in a hierarchical coherent structure, where each object corresponds to a single node. Results of the classification decision trees can be transformed into sets of rules which refer to the logical design of the form “If-Then”.

Nowadays there is a great number of algorithms implementing decision trees - CART, C4.5, NewId, ITrule, CHAID, CN2, etc. The most popular are the following:

- a) CART (Classification and Regression Tree) - algorithm for constructing of binary decision tree - the dichotomic classification model. Each node of the tree has only two branches. This algorithm is designed for problems of classification and regression. The choice of classes is based on probabilistic approach using the so-called "index of Gini" (after the Italian economist C. Gini) evaluating the "distance" between classes' distributions.
- b) CHAID (Chi-Square Automatic Interaction Detector). In this algorithm the construction of classification trees produces multiple branching, where each parent node gives branches to more than two child nodes. As a criterion of classification chi-square test is used. Data analysis can be performed for both continuous and categorical independent variables.
- c) C4.5 (improved algorithm ID3 - "Iterative Dichotomizer") uses theoretic information approach (the evaluation of entropy) to split the original array of data into classes, as well as the procedure of the truncation of the resulting tree branches and construction of classification rules in the "If-Then" pattern to create a tree of solutions. This algorithm does not work with a continuous dependent variable, so it could be used only for the classification.

There is a great number of software products generating decision trees algorithms. We used the following software tools:

- a) Data-Mining module of STATISTICA v. 6.1 software (StatSoft, Inc., USA, statsoft.com);
- b) Demo version of See5 software (RuleQuest Research, Australia, rulequest.com), limitation of demo version is that it processes no more than 400 cases;
- c) Demo version of WizWhy 4.06 software (WizSoft, Inc., USA, wizsoft.com), limitation of demo version is that it processes no more than 1000 cases.

Full set of data forms the table of 164 variables and 298 cases. Decision variable has three values, and predictors are 154 categorical and 9 quantitative variables. One hundred and fifty predictor variables contain missing values. As a result of different procedures of reducing the variables' space dimension (expert evaluation, statistical analysis, genetic algorithm) data set of 30 independent variables was created. From this truncated data set random samples of 80% of cases for learning and 20% of cases for testing were generated (239 and 59 cases, respectively).

## 2 Applications of C&RT and CHAID Algorithms

Of the numerous methods of data mining available in the STATISTICA package (generalized regression and cluster analysis, multivariate adaptive regression splines, extensible simple trees, association rules, etc.) it was possible to use only those that allow to process data with missing values:

- interactive process of classification and regression ITrees C&RT;
- interactive procedure ITrees CHAID;
- generalized classification and regression trees;
- generalized CHAID models.

The first two procedures have mode of building a complete “Grow Tree” and truncated “Grow Tree & Prune” decision tree. The results are presented in Table 1. Unfortunately, none of the methods used gave acceptable results for any class even for learning sequence. In some experiments cases of class 3 were not detected at all. Some cases are not recognized, so terminal nodes include less number of cases.

**Table 1.** Classification of learning cases (n = 239) by sets of rules based on C&RT and CHAID algorithms

| Observed classes                      | Predicted N | Predicted class 1 | Predicted class 2 | Predicted class 3 | Not recognized | Correct classification % |
|---------------------------------------|-------------|-------------------|-------------------|-------------------|----------------|--------------------------|
| <b>C&amp;RT.Grow Tree</b>             |             |                   |                   |                   |                |                          |
| Class 1                               | 168         | 153               | 9                 | 2                 | 4              | 91                       |
| Class 2                               | 61          | 13                | 32                | 7                 | 9              | 53                       |
| Class 3                               | 10          | 2                 | 0                 | 8                 | 0              | 80                       |
| <b>C&amp;RT.Grow Tree &amp; Prune</b> |             |                   |                   |                   |                |                          |
| Class 1                               | 168         | 161               | 5                 | 0                 | 2              | 96                       |
| Class 2                               | 61          | 23                | 31                | 0                 | 7              | 51                       |
| Class 3                               | 10          | 2                 | 8                 | 0                 | 0              | 0                        |
| <b>CHAID.Grow Tree</b>                |             |                   |                   |                   |                |                          |
| Class 1                               | 168         | 160               | 7                 | 0                 | 1              | 95                       |
| Class 2                               | 61          | 30                | 31                | 0                 | 0              | 51                       |
| Class 3                               | 10          | 10                | 0                 | 0                 | 0              | 0                        |

STATISTICA v. 6.1 allows to provide an analysis in the tabular and in graphical form (Fig.1). Figure shows branching and terminal nodes (called leaves) are depicted as rectangles, which contain a number of nodes (ID), the number of observations assigned to this node (N), the number of observations from the initial classes of the dependent variable in the form of bar histogram (the number is in the center of the rectangle).

### 3 Application of Algorithm C4.5

See5 software uses algorithm C4.5 which yields a decision tree and sets of logical rules such as “If-Then”. A procedure of so called “adaptive boosting” creates a sequence of trees, each of them takes into account the errors of previous decisions. It is assumed that it may reduce the resultant classification error. Number of trees may vary. Another possibility to improve the accuracy of classification is

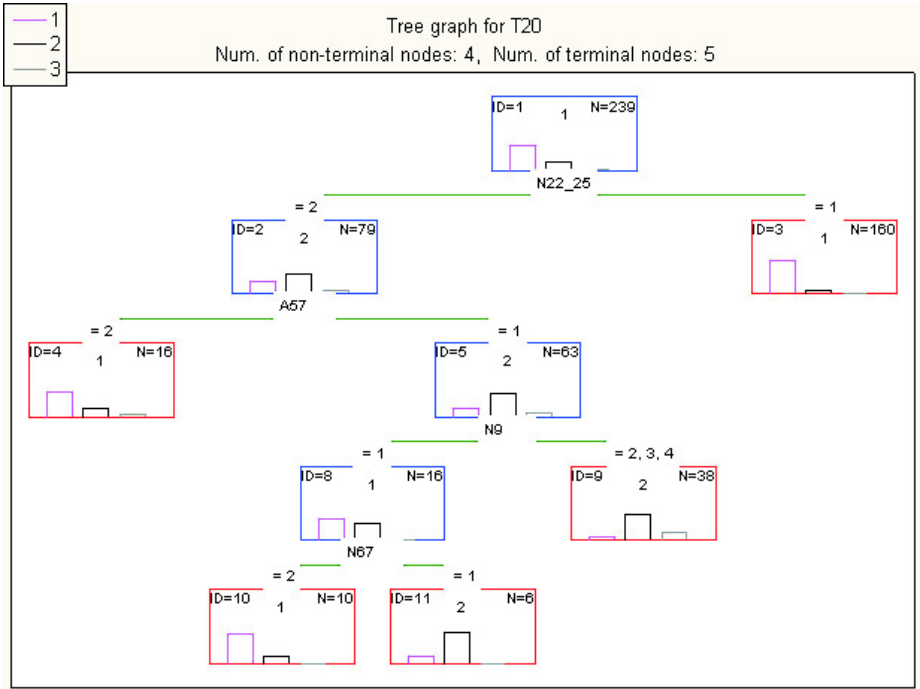


Fig. 1. Example of classification tree in Statistica v. 6.1 software

Table 2. Classification of cases by the set of rules constructed using the algorithm C4.5

| Observed classes   |         | Predicted | Predicted | Predicted | Correct classification, % |
|--------------------|---------|-----------|-----------|-----------|---------------------------|
|                    |         | class 1   | class 2   | class3    |                           |
| Learning (n = 239) | Class 1 | 168       | 0         | 0         | 100                       |
|                    | Class 2 | 7         | 54        | 0         | 89                        |
|                    | Class 3 | 0         | 0         | 10        | 100                       |
| Testing (n=59)     | Class 1 | 43        | 1         | 0         | 98                        |
|                    | Class 2 | 7         | 5         | 1         | 38                        |
|                    | Class 3 | 0         | 1         | 1         | 50                        |

the use of fuzzy thresholds to select branches of decision tree (the lower, central and upper limits). These values are set by the program.

Using interface one can set the parameters of so-called “construct classifier”. At the first stage, the system offers the default values of two parameters: the rate of truncated branches (25%) and the number of cases on the branches (2).

The disadvantage of this system is a time-consuming procedure of data preparation of two text files, one of which is the list of variables and their descriptions,

and another - the values of variables. Missing values are indicated by the “?” sign.

We carried out calculations both in the mode of obtaining a set of rules, and in the mode of constructing decision trees. We used the following parameters: the number of trees - 10, pruning confidence level - 50%, the number of cases on the branches - 2. The results are presented in Table 2. The level of errors in training was 2.9%, in testing 16.9%.

Calculations showed that the only effective mode was the combination of amplification of decision and fuzzy rules. Despite that the training was successful with zero errors for classes 1 and 3, the classification of test cases gave unacceptable results.

## 4 Application of WizWhy v. 4.06 Demo

The essence of algorithms of this commercial system is not disclosed, however taking into account that it is one of the most successful products of Data Mining, we decided to apply it in our investigations. This system has a great set of tools for the construction and subsequent analysis of decision rules, including “If-Then”, “If-And-Only-If” and “Unexpected”. The system creates a full set of rules for each value of dependent variable. Table 3 shows the result of only one version of calculations for the entire sample (100% of cases).

The classification results are good enough, but the interpretation of such quantity of rules is practically impossible, also the demo version does not generate the code of produced models, which limits their use in practice.

**Table 3.** The results of applying the algorithm WizWhy v. 4.06 Demo

| Observed classes | Number of rules | Correct classification,<br>% |
|------------------|-----------------|------------------------------|
| 1                | 8904            | 90.6                         |
| 2                | 7303            | 89.2                         |
| 3                | 2399            | 93.3                         |

## 5 Conclusion

The computational procedures of STATISTICA software did not give acceptable results of the classification. The system See5 was able to give low classification error in the mode of constructing a decision tree with decisions amplification in combination with fuzzy thresholds. The worse results were in the mode of set of rules constructions with the same values of parameters. Unfortunately, the rule sets obtained on the training samples gave unsatisfactory results. System WizWhy showed acceptable accuracy, but practical use of the results is rather complicated.

## References

1. Kotova, E.Y.: Clinical and epidemiological characteristics, the leading risk factors, the features of the of stroke in Ulyanovsk, Russia (according to the Register of stroke). Abstract of PhD thesis, Moscow, p. 25 (2009) (in Russian)
2. Gusev, E.I., et al.: Epidemiology of stroke in Russia. *J. Neurol. Psychiatry, Suppl.* 103(8), 4–9 (2003) (in Russian)
3. Wardlaw, J.M., Keir, S.L., Dennis, M.S.: The impact of delays in computed tomography of the brain on the accuracy of diagnosis and subsequent management in patients with minor stroke. *J. Neurol. Neurosurg. Psychiatry* 74(1), 77–81 (2003)
4. Saur, D., Kucinski, T., Grzyska, U., et al.: Sensitivity and interrater agreement of CT and diffusion-weighted MR imaging in hyperacute stroke. *Am J. Neuroradiol.* 24(5), 878–885 (2003)
5. Kalafut, M.A., Schriger, D.L., Saver, J.L., Starkman, S.: Detection of early CT signs of 1 / 3 middle cerebral artery infarctions: interrater reliability and sensitivity of CT interpretation by physicians involved in acute stroke care. *Stroke* 31(7), 1667–1671 (2000)
6. Dippel, D.W., Du Ry van Beest Holle, M., van Kooten, F., Koudstaal, P.J.: The validity and reliability of signs of early infarction on CT in acute ischaemic stroke. *Neuroradiology* 42(9), 629–633 (2000)
7. Grotta, J.C., Chiu, D., Lu, M., Patel, S., et al.: Agreement and variability in the interpretation of early CT changes in stroke patients qualifying for intravenous rtPA therapy. *Stroke* 30(8), 1528–1533 (1999)
8. Yu, R.O.: Application of data mining to solve the problem of medical diagnostics. *News of Artificial Intelligence* (3), 76–80 (2004) (in Russian)
9. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: *Classification and Regression Trees*. Cole Advanced Books & Software. Wadsworth & Brooks, Monterey (1984)
10. Kass, G.V.: An exploratory technique for investigating large quantities of categorical data. *Applied Statistics* 29, 119–127 (1980)
11. Friedman, J.H.: *Stochastic Gradient Boosting*. Stanford University (1999)
12. Hastie, T., Tibshirani, R., Friedman, J.H.: *The Elements of Statistical Learning: Data mining, Inference and Prediction*. Springer, New York (2001)
13. Hyafil, L., Rivest, R.L.: Constructing Optimal Binary Decision Trees is NP - complete. *Information Processing Letters* 5(1), 15–17 (1976)

# Discovering Potential Precursors of Mammography Abnormalities Based on Textual Features, Frequencies, and Sequences

Robert M. Patton and Thomas E. Potok

Oak Ridge National Laboratory  
{pattonrm,potokte}@ornl.gov

**Abstract.** Diagnosing breast cancer from mammography reports is heavily dependant on the time sequences of the patient visits. In the work described, we take a longitudinal view of the text of a patient's mammogram reports to explore the existence of certain phrase patterns that indicate future abnormalities may exist for the patient. Our approach uses various text analysis techniques combined with Haar wavelets for the discovery and analysis of such precursor phrase patterns. We believe the results show significant promise for the early detection of breast cancer and other breast abnormalities.

## 1 Introduction

Most research involving mammography is performed on the image data, usually without regard to the corresponding textual reports. These reports are unstructured text, written by a human subject-matter expert, about the images of a patient. A set of reports for an individual patient forms a sequence of observations based on each patient visit. We believe that frequency analysis of phrase patterns can be analyzed to enhance the accuracy of detecting and forecasting breast anomalies. Our work focuses on a longitudinal view of the patients with respect to the mammogram reports. We believe a frequency analysis of the complete patient record will provide precursors to breast anomalies that currently cannot be detected. Consequently, this work seeks to explore the following questions:

- Do phrase patterns exist that act as precursors to future abnormalities in a patient?
- If so, how far in advance of the abnormality do they occur?

Answers to such questions may provide enhanced, automated detection as well as more efficient and effective care of patients. This paper describes initial work being performed to answer such questions. Section 2 discusses the background of the radiology reports being addressed by this work as well as the natural language processing that is needed. Section 3 discusses the analysis approach, while section 4 discusses results. Section 5 discusses conclusions.



## 2 Background

In this initial study, reports from 12,809 patients over a 5-year period are analyzed. There are 61,064 actual reports in this set, which include a number of reports that simply state that the patient canceled their appointment. For the work discussed here, we are particularly interested in studying the patients with multiple reports over time. Some of these patients have reports that predate the study and also have diagnostic screenings, indicating a potential health problem. Abnormal reports tend to have a richer, broader, and more variable vocabulary than normal reports. In addition, normal reports tend to have a higher number of “negation” phrases. These are phrases that begin with the word “no” such as the phrase “no findings suggestive of malignancy” [3]. Another challenge in analyzing the natural language of these reports is that there are multiple ways of conveying the same meaning. Phrases such as “no strongly suspicious masses” and “no new suspicious mass lesions” both mean that nothing cancerous was observed. To account for this variability in the language, we used the natural language processing technique known as skip bigrams, or s-grams. S-grams are word pairs in their respective sentence order that allow for arbitrary gaps between the words [6]. For example, the s-gram for the previous phrase examples is the words *no* and *suspicious*. As discussed in [4], s-grams are an effective technique in determining normal and abnormal reports. The next step is to analyze patient records to determine if abnormal report occurrences can be forecasted.

## 3 Approach

The objective of this work is to identify whether certain phrase patterns exist within patient reports that act as precursors to future abnormalities. To explore this, we analyzed the patient records that contain a higher number of reports. This narrowed the patient data set to 667 patients who had between 12 and 16 reports each. Of these patients, the ones of most interest for this work are those with discernable patterns in their medical reports. For example, a patient with cancer may have had an early report that mentions something unusual or suspicious, followed by years of normal reports, before the cancer appeared. The early report may be a precursor for the cancer. To identify these patients, each report in each patient record is analyzed to count the number of normal and abnormal s-grams as described in [4]. This provided a temporal sequence of normal and abnormal s-gram counts for each patient record. The following table shows an example patient record where s-grams were counted for each report.

In the example shown in Table 1, there is some abnormality that is mentioned early in the record (May 24, 1984) and then the record contains multiple abnormal s-grams toward the end of the record (beginning on Dec 7, 1991). The normal and abnormal s-gram counts form a temporal sequence for each patient. Our goal is to be able to compare patients based on these sequences. To find patients with similar patterns in the set of 667, we use a discrete wavelet transform (DWT) of the temporal sequence of abnormal s-gram counts [1][5]. A wavelet

**Table 1.** Example record of normal and abnormal s-gram counts for patient A

| Mammogram Date | Normal S-grams | Abnormal S-grams |
|----------------|----------------|------------------|
| May 20, 1981   | 1              | 0                |
| May 24, 1984   | 3              | 1                |
| June 3, 1985   | 3              | 0                |
| March 9, 1988  | 1              | 0                |
| July 12, 1989  | 4              | 0                |
| Dec 5, 1990    | 3              | 0                |
| Dec 7, 1991    | 1              | 4                |
| March 11, 1992 | 0              | 4                |
| March 11, 1992 | 0              | 4                |
| March 22, 1992 | 0              | 1                |
| March 22, 1992 | 0              | 1                |
| March 23, 1992 | 0              | 0                |
| Nov 9, 1992    | 0              | 0                |

transform is a mathematical function that is used to split a function into separate scale components, thus providing a multi-resolution analysis. The wavelet transform is analogous to a prism that breaks light into its various spectral colors. They are widely used in time-series analysis, as well as in other domains such as image processing. A critical feature of the DWT is that it will not only identify the frequencies that constitute a temporal sequence, but also the location in time in which those frequencies occur. It is this feature of the DWT that is exploited in this work, as our objective is to find phrase patterns that occur prior to other phrase patterns. In addition, a DWT provides the ability to find similar temporal patterns, allowing for the flexibility of matching patterns despite amplitude and time shifts. Previous work has shown wavelets to be effective in performing similarity searches of time series [2]. However, the work described here utilizes a rule-based approach to finding similar temporal patterns using DWT that does not rely on the use of thresholds. This enables a wider range of temporal patterns to be found that contain the basic temporal characteristics of interest. Each patient record consisted of 16 or fewer reports. For records with less than 16 reports, the temporal sequences were padded with zeros until there were 16 elements. Next, for each patient record, the temporal sequence of abnormal s-gram counts were transformed using a Haar wavelet [1]. For example, the transform for patient A (of Table 1) is shown in the following table. After each of the 667 patient records is transformed via a Haar wavelet, the next step is to begin looking for the patterns of interest, early abnormality and late anomaly. First, resolution 1 of each patient is examined. Specifically, the first coefficient of resolution 1 should be less than 0 while the second coefficient of resolution 1 should be greater than 0. This particular pattern identifies those patients with an increasing amplitude change in their s-gram counts toward the end of the

**Table 2.** Haar wavelet transform of abnormal s-gram sequence for patient A

| 1st coefficient 0.9375 |                        |
|------------------------|------------------------|
| Band 0                 | 0.1875                 |
| Band 1                 | -0.875 0.75            |
| Band 2                 | 0.25 -2 1 0            |
| Band 3                 | -0.5 0 0 0 1.5 0.5 0 0 |

records (rather than at the beginning of their records), which suggests that diagnostic screening was performed near the end of the patient's record. Second, if the pattern for resolution 1 exists, then resolution 2 of each patient is examined. Specifically, either the first or second coefficient (or both) of resolution 2 should be less than 0 while the third and fourth coefficients should both be greater than 0. This particular pattern identifies those patients who have a short duration of abnormal s-gram counts early in the records, which suggests that some unusual feature about the patient was mentioned early in their record. For higher resolution, resolution 3 could be used instead of resolution 2. In that case, the first four coefficients would be checked for negative values, while the last four coefficients would need to be positive. Patient records that match these patterns in the Haar DWT are then selected. This reduced the data set to 123 patient records, which is approximately 1% of the original data set. For these selected patient records, all s-grams were extracted from the first report in which the abnormal s-gram count was at least 1 but less than or equal to the normal s-gram count. This represents a normal report where some potential abnormality was mentioned. Next, the time elapsed was computed between this first report and the next report where the abnormal s-gram count was higher than the normal s-gram count. This second report represents an abnormality that was detected and a diagnostic screening was requested. From the example data shown of patient *A* in Table 1, the first report would be the one dated May 24, 1984 and the next report would be the one dated December 7, 1991. All s-grams from the report dated May 24, 1984 are extracted and considered as potential precursor patterns. Finally, the frequency of each extracted s-gram was computed along with the corresponding average elapsed time. The results of this approach applied to the 123 selected patients are shown in the following tables and figures.

## 4 Results

Table 3 shows the top three precursor s-grams that were observed. In reviewing this table, there is no single definitive precursor s-gram. However, the top three s-grams have approximately a fifty percent occurrence as a precursor. This means that, of the 123 selected patients, if one of those s-grams were mentioned in the patient's record, then there is a fifty percent chance that the patient will have a diagnostic screening (i.e., an abnormality will be seen that requires additional testing) at some point in the future. While this percentage is equivalent to random selection, in comparison to the other s-grams found, these s-grams show promise as potential precursors and demonstrate a capability far beyond the current state of the practice, which is dependent entirely on manual analysis. Table 4 shows the average elapsed time in units of days for each of the s-grams shown in Table 3. What is very encouraging in these results is that the first and third s-grams provide approximately a three to five year lead-time. This provides a very early warning indication of a future abnormality. The drawback, however, is that the skewness and kurtosis values for these s-grams indicate significant variability in this window. The reason for this is that these terms are

**Table 3.** Top three precursor s-grams

| S-gram            | Occurrences as Precursor | Occurrences in Selected Patients | % Occurrence as Precursor |
|-------------------|--------------------------|----------------------------------|---------------------------|
| lymph & node      | 39                       | 71                               | 54.93                     |
| cm & density      | 12                       | 24                               | 50.00                     |
| nodular & density | 51                       | 104                              | 49.04                     |

general and vague in their meaning, but still provide some level of indication that the radiologist sees a feature of concern. In contrast, the second s-gram (*cm & density*) provides a much more specific window with an average of just over one year with very high positive skewness and kurtosis values. The reason for this is that this s-gram represents phrase patterns that are very specific about a particular feature that was observed in the patient. An example phrase that this s-gram would represent is “2.5 cm area of asymmetric density”. Such specificity by the radiologist suggests that the radiologist is very focused on this feature and is likely to be concerned enough to request additional diagnostic screenings. Consequently, the average time elapsed for this s-gram is much shorter and has less variability. The data in Table 5 shows the usage frequency of the s-grams shown in Table 3. In document text analysis, terms and phrases that are commonly used in a document set are not considered useful in characterizing the content of a particular document. However, if a term or phrase is not commonly used in a document set and a particular document has a high frequency of that term or phrase, then it is considered significant to that document. In a similar manner, the frequency of s-grams in Table 3 were computed over all of the patients (12,809 patients) and over the patients that were selected for analysis (123 patients). These frequencies, as well as the corresponding percent increase, are shown in Table 5. As can be seen, most of the s-grams have percent increases well over 100%. This is encouraging as it shows that these s-grams are highly related to patients with abnormalities. If the percent increases had been much below 100%, then this would indicate that these s-grams are very common, and consequently, the value as a precursor would be diminished. However, the percent increases and corresponding percent occurrence in selected patients shown in the table suggest that these s-grams have high potential as precursors.

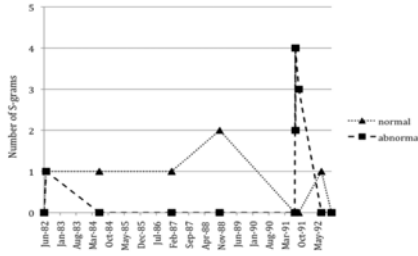
The figures show various patient records that were found using the approach described here. Figure 1 shows the normal and abnormal s-gram counts of a patient record found by this approach where “*lymph & node*” was a precursor s-gram. In one of the first reports in this record, a radiologist made particular note of specific lymph nodes in this patient. This patient was ultimately diagnosed

**Table 4.** Precursor lead-time

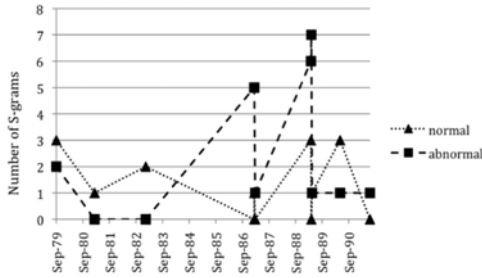
| S-gram            | Average Time Elapsed (years) | Std Dev (years) | Skewness / Kurtosis |
|-------------------|------------------------------|-----------------|---------------------|
| lymph & node      | 4.2                          | 2.9             | 0.01 / -1.38        |
| cm & density      | 1.1                          | 2.2             | 2.63 / 6.91         |
| nodular & density | 2.9                          | 2.9             | 0.68 / -0.64        |

**Table 5.** Comparison of s-gram usage frequency

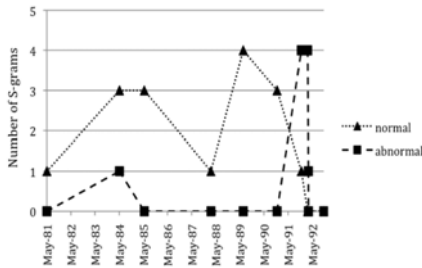
| S-gram            | % Occurrence in All Patients | % Occurrence in Selected Patients | % Increase in Occurrence |
|-------------------|------------------------------|-----------------------------------|--------------------------|
| lymph & node      | 25.17                        | 57.72                             | 129.34                   |
| cm & density      | 5.50                         | 19.51                             | 254.51                   |
| nodular & density | 31.39                        | 84.55                             | 169.35                   |



**Fig. 1.** Example patient record with “lymph” & “node” as a precursor s-gram



**Fig. 2.** Example patient record with “cm” & “density” as a precursor s-gram



**Fig. 3.** Example patient record with “nodular” & “density” as a precursor s-gram

with grade 1 infiltrating ductal carcinoma (i.e., breast cancer) with tubular differentiation. Figure 2 shows the normal and abnormal s-gram counts of another patient record found by this approach where “*cm & density*” was a precursor s-gram. In the first report of this record, the radiologist states “There is a less

than 1 cm area of focal increased density seen only on the left craniocaudal view in the lateral aspect of the left breast.” This patient was ultimately diagnosed with “mild fibrocystic disease with radial scar and focal florid sclerosing adenosis” in the right breast. Figure 3 shows the normal and abnormal s-gram counts of another patient record found by this approach where “*nodular & density*” was a precursor s-gram. In one of the first reports, the radiologist states “There is prominent nodular density posteriorly and inferiorly in both breasts on the mediolateral oblique views, left more than right.” This patient is ultimately diagnosed with a simple cyst. In that report, the radiologist states “Ultrasound directed to the inferocentral left breast 6 o’clock position demonstrates a 1-cm round, simple cyst.” In each of these examples, it should be noted that the precursor s-gram does not necessarily provide specific information concerning the abnormality that is ultimately diagnosed. In the first two examples, the s-grams are not related to the ultimate diagnosis. In the third example, the precursor s-gram is related, but it cannot be conclusively determined that it is, in fact, the exact same abnormality that is ultimately diagnosed. However, what the precursor s-gram does provide is an early warning indication that the radiologist noted some feature about the patient that seemed unusual, or was noteworthy. The approach described here seeks to leverage that information, even if it does not ultimately relate to the final diagnosis.

## 5 Conclusions

The initial objective of this work was to answer the following questions:

- Do certain phrase patterns exist that act as precursors to future abnormalities in a patient?
- If so, how far in advance of the abnormality do they occur?

As can be seen in the results, phrase patterns do exist that act as precursors. In addition, these precursors also hold the potential of providing lead times measured in years. This is potentially very significant, although additional work is needed to investigate this possibility. In this work, there are several other positive outcomes. First, in the approach developed, abnormal reports are identified based on s-grams related to diagnostic screenings, not based on specific types of abnormalities. Consequently, the precursor s-grams provide a general warning indication. Any form of early warning detection will provide various levels of specificity. This preliminary work provides the initial level of warning. Second, the results show that the precursor s-grams are used much more frequently in patients with abnormalities in comparison to the entire set of patients. This is significant in that it provides confidence that these precursor s-grams are, in fact, related to abnormalities.

## Acknowledgements

Our thanks to Robert M. Nishikawa, Ph.D., Department of Radiology, University of Chicago for providing the large dataset of unstructured mammography reports, from which the test subset was chosen.

Research sponsored by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory (ORNL), managed by UT-Battelle, LLC for the U. S. Department of Energy under Contract No. De-AC05-00OR22725.

This manuscript has been authored by UT-Battelle, LLC, under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

## References

1. Burrus, C.S., Gopinath, R.A., Guo, H.: Introduction to Wavelets and Wavelet Transforms, A Primer. Prentice Hall, Englewood Cliffs (1997)
2. Chan, F.K.-P., Fu, A.W.-C., Yu, C.: Haar wavelets for efficient similarity search of time-series: with and without time warping. *IEEE Trans. on Knowledge and Data Engineering* 15(3) (May-June 2003)
3. Patton, R.M., Beckerman, B.G., Potok, T.E.: Analysis of mammography reports using maximum variation sampling. In: Proceedings of the 4th GECCO Workshop on Medical Applications of Genetic and Evolutionary Computation (MedGEC), Atlanta, USA, July 2008. ACM Press, New York (2008)
4. Patton, R.M., Beckerman, B.G., Treadwell, J.N., Potok, T.E.: A Genetic Algorithm for Learning Significant Phrase Patterns in Radiology Reports. In: Proceedings of the 5th GECCO Workshop on Medical Applications of Genetic and Evolutionary Computation (MedGEC), Montreal, Canada, July 2009, ACM Press, New York (2009)
5. Percival, D.B., Walden, A.T.: Wavelet methods for time series analysis. Cambridge University Press, Cambridge (2000)
6. Pirkola, A., Keskustalo, H., Leppanen, E., Kansala, A., Jarvelin, K.: Targeted s-gram matching: a novel n-gram matching technique for cross- and monolingual word form variants. *Information Research* 7(2) (2002), <http://InformationR.net/ir/7-2/paper126.html>

# An Expert System for Human Personality Characteristics Recognition

Danuta Rutkowska

Department of Computer Engineering, Częstochowa University of Technology  
Armii Krajowej 36, 42-200 Częstochowa, Poland

<http://kik.pcz.pl>

Academy of Management (SWSPiZ), Institute of Information Technology  
Sienkiewicza 9, 90-113 Łódź, Poland

<http://www.swspiz.pl/>

**Abstract.** In this paper, a hybrid expert system that can recognize some personality characteristics, based on human face pictures, is proposed. The expert system includes a neural network and a fuzzy system for pattern recognition and classification. The hybrid intelligent expert system can produce a short psychological portrait - expressed in a natural language - of a person whose face is presented in the picture.

## 1 Introduction

Possibility of human personality recognition based on face pictures seems to be controversial. Some people believe that it is possible, while many others are rather sceptical. According to the author's knowledge as well as her intuition, the idea of the expert system proposed in this paper is worth realizing.

Such a system may be helpful for recognizing some selected psychological features of persons, in special applications. For example, the expert system can discover important personality characteristics of students or people applying for a job. In these situations, concerning students or job candidates, it may be very advantageous to know their abilities - in order to choose a suitable education way or find a proper job position.

There are many face reading practitioners (the information can be found on the Internet). Those people claim that it is possible to read features of someone's face and understand his or her personality (e.g. [21]).

Face reading is a general term that includes expression reading, physiognomy, and personology. Expression reading means interpretation of a person's feelings and thoughts by their facial expression (induced by their facial muscles). This is related to emotion recognition (see e.g. [24]); we can observe and analyze facial features for emotional expressions. Physiognomy is the art and science of judging or determining a person's facial characteristics and structure (see e.g. [1]). A significant book was published in 1800's by Lavater [15]. Personology is a form of physiognomy (see e.g. [12]). As a result of synergistic culmination of years of research and analysis by scientists and philosophers - through a blending



of genetics, physiology, anatomy, and neurology, we know that each individual has a unique personality profile.

Face reading is an ancient psychological system of understanding a person's character from his (or her) facial features. The roots of face reading start from ancient writings about the relationship between physical appearance, temperament and behavior. Later, great Greek philosophers, with Aristotle as the main, developed a system of judging character base on physiognomic thought. Face reading is an ancient, yet surprisingly accurate and effective way to uncover anyone's personality, potential and abilities [6]. In the early 1900's, all the previously available information on physiognomy was included in the book [4] that recently has been reproduced. Now there are many books concerning face reading (see e.g. [8], [17], [22], [23], [27]) and Internet web-sites.

The ancient Chinese art of face reading is over 2.500 years old, and it was documented as long ago as the time of Confucius. By looking at person's face and the features on it, the reader gets a lot of information: the character of the person, the potential, disposition, creativity, and whether the person is fortunate or not so fortunate. In face reading, capability for something or personality traits are, indeed, shown in the features and other facial markings. More details can be found on the Internet, e.g. [3]; see also [5], [10], [14].

There are also computer programs that face reading practitioners may employ. The Digital Physiognomy software [7] uses a sophisticated neural network to identify correlations between facial features and psychological characteristics using photo identification techniques. The algorithm was tested and the analysis indicated that the correlation was significant. The program determines a person's psychological characteristics and presents a detailed character analysis of that person in a graphic format or can give a written report. Only facial features that may be interpreted by physiognomy were used. The eyes, eyebrows, foreheads, cheekbones, chins, noses, mouths and ears are selected to assemble a face. The program does not forecast the future, but discovers how others really see a person. The Digital Physiognomy software tries to match the person's face to his or her character. The program determines a person's psychological characteristics based on: temperament, intellect, optimism-pessimism, conformism-adventurism, egoism-altruism, philanthropy-hostility, laziness, honesty, sexuality, lucky, humor. The main conclusion from this program is that physiognomy not so much allows others to determine someone's real personality and possible behavior, as it allows the person to understand how others see him or her.

Another approach to face reading, different from the physiognomy outlined above, was introduced by Anuashvili [2]. He analyzed oscillatory processes in human brains registered as EEG signals, and relations between activities of brain hemispheres and the psychological type of a person. He considered the so-called phase portrait of a human face that - in his opinion - reflects states of two brain hemispheres; especially dominance or non-dominance of the left or right hemisphere. For details, see also e.g. [19].

## 2 The Hybrid Expert System

In this section, an idea of the expert system for human personality characteristics recognition is proposed. The system should be viewed as an intelligent hybrid system created by use of a classic expert system (like well-known in artificial intelligence) combined with computational intelligence methods such as neural networks and fuzzy systems as well as evolutionary algorithms (see e.g. [25]). The system is intended to be applied in both face recognition and face reading. The former concerns the tasks of a person identification and classification, based on pictures presented to the system, taking into account special characteristics, e.g. face asymmetry. The latter is related to reading personality traits from the pictures, using the physiognomy knowledge as well as other approaches (similar to those described in [2] and [19]).

### 2.1 System Architecture

A classic expert system architecture consists of a knowledge base, an inference engine, an explanation facility, and a user interface (see e.g. [11]). The hybrid expert system, proposed in this paper, in addition, includes a neural network (e.g. [32]), and a fuzzy system [30].

At the input to the system, pictures of human faces are presented. Based on the pictures, the inference engine - using the knowledge base - produces short personal characteristics, expressed in a natural language, provided at the output of the expert system.

From the explanation facility, users can get information confirming the output result with regard to the rules, included in the knowledge base, associated with the description produced at the output.

The fuzzy system is implemented in the expert system in a synergic way. This means that the knowledge base includes fuzzy IF-THEN rules, and the inference engine uses fuzzy logic to produce the output decision [23]). The explanation facility also refers to the fuzzy rules to give users information concerning the result produced by the system.

The neural network is applied separately from the rule base and fuzzy logic inference. The network has learning ability to acquire knowledge from examples of the face pictures and personality types assigned to the presented faces. The knowledge is hidden in the neural network's connection weights. Thus, the neural network, after the learning process, can be employed to use its knowledge in order to identify and classify a person. Apart from typical classification tasks, based on a face shown in the picture fed to the input of the expert system, the neural network may decide to which personality type (according to the psychologic categorization [20]) the person belongs.

In addition to the neural network application described above, another neural network can also be used in order to recognize facts concerning values of face attributes, such as size of particular parts of the face, e.g. nose is big or small, lips are narrow or wide. This is needed in order to perform inference based on the physiognomy knowledge.

## 2.2 Fuzzy Knowledge Base

It is assumed that the knowledge is acquired from experts. For example, the knowledge presented in [8], as well as in many other similar sources related to face reading and physiognomy (also available on the Internet) may be used. This kind of knowledge can be easily expressed in the form of fuzzy IF-THEN rules.

It is worth mentioning that the knowledge concerning human faces ought to be gathered using the fuzzy description. As Prof. Zadeh - who introduced fuzzy sets [28] and fuzzy logic [30] - often emphasizes, it is not possible to determine precise borders between particular parts of a human face, such as the nose, cheek, etc. Thus, the fuzzy borders must be considered.

Therefore, the fuzzy IF-THEN rules describe the expert knowledge implemented in the rule base of the expert system proposed in this paper. The rules can be formulated as follow: IF nose is big and ... THEN ... - where "nose" is a linguistic variable with "big" as its value that is represented by a fuzzy set defined by an expert.

The rule base may contain such rules that each of them corresponds to a particular part or special feature of a human face, e.g. nose, eyes, lips, ears, forehead, as well as some special lines or points visible in the face. Antecedent parts of the rules (IF parts) can contain more than one linguistic variable representing particular parts of the human face. Of course, more than one rule may have the same consequent part (i.e. THEN part of the IF-THEN rules) including the rule conclusion.

Fuzzy sets that correspond to the linguistic variables in the rule base are characterized by their membership functions which can be defined in the way presented in Fig.1. To make the illustration clear, only one membership function - for the eye - is shown. We can easily construct similar membership functions for other parts of the face, as well as for the face shape itself. Values of the membership functions equal 0 for pixels that do not belong to the area of the face shape, and equal 1 for pixels that definitely belong to corresponding regions of the face shape and particular parts of the face, i.e. eyes, nose, mouth, cheeks, etc.. Thus, we divide the picture of a face into fuzzy parts, with fuzzy boundaries between them. This means that pixels laying on the boundaries may belong to more than one fuzzy set with some degrees of membership (greater than 0 and less than 1).

## 2.3 Fuzzy Logic Inference and Results

The inference procedure is based on the information about the degree of how much the facts, such as size of the nose, eyes, etc. match antecedents of the rules in the rule base. Thus, it should be calculated how the facts suit to the corresponding fuzzy sets in the rule antecedents, according to the membership functions of the fuzzy sets. In this way, the level of rule activation is determined for every rule in the rule base, and the fuzzy logic is employed to produce the result of system inference (see e.g. [25]).

As mentioned in the Introduction, each individual (human being) has a unique personality profile. Therefore, results of face reading should be different for every

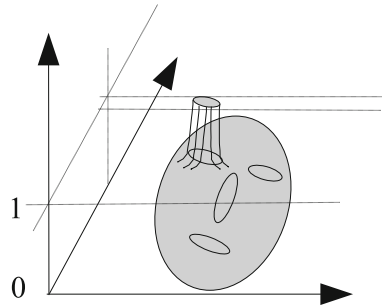


Fig. 1. Fuzzy sets that represent particular parts of a human face

person. However, personality traits are usually associated with basic categories of personality types known in psychology (see e.g. [9], [20]). The intelligent expert system, proposed in this paper, can be employed to recognize human personality traits in both cases. This means that typical classification tasks, which can be realized by classic expert systems, may be performed with application to personality traits recognition based on classic (non-fuzzy) IF-THEN rules. In addition, due to the fuzzy rule base, face reading can produce results that may be interpreted in a fuzzy way. For example, the system inform that a person belongs to more than one category, with certain degrees of the membership. The use of fuzzy IF-THEN rules, and fuzzy logic inference, allows to obtain different personality profile descriptions for different people.

Both classic and fuzzy IF-THEN rules may be exactly of the same (linguistic) form but the difference is significant. It concerns semantic interpretation of the rules. As a matter of fact, there is no meaning of the linguistic terms in the classic rules that the system can understand. Therefore, different inference methods are applied in classic expert systems (see e.g. [11]). The linguistic terms in fuzzy IF-THEN rules are linguistic variables with fuzzy sets (or fuzzy numbers) as their values; for details see [29] as well as [25]. Thus, we can say that the intelligent hybrid expert system "understands" the meaning of fuzzy IF-THEN rules included to its rule base because the semantic interpretation is encoded in the membership functions of fuzzy sets in the rules. Since the membership functions has been defined the meaning of the linguistic values (fuzzy sets) are well known for the system. Hence, results of the fuzzy inference can easily be explained using precisely computed values of rule activation levels. It should be emphasized that in fuzzy systems all rules in the rule base are activated in parallel while in classic expert systems rules are executed sequentially. Therefore, fuzzy systems - when employed to classification tasks - can produce results saying that an object belongs to more than only one class, and informing about degrees of the membership.

The psychological portraits of persons, whose pictures are presented at the input of the hybrid intelligent expert system, should be generated as short descriptions in a natural language. Results of the fuzzy classification ought to be

included. To prepare such a summarization of personality traits - in an intelligent way - a special algorithm, based on fuzzy approach and computing with words [13], [25], [31], should be elaborated and applied.

The system will be able to solve various tasks concerning face reading, and produce conclusions depending on the knowledge implemented in the knowledge base (rule base). In addition, the system will possess an ability to learn (acquire new knowledge from examples).

### 3 Conclusions and Final Remarks

As mentioned in the introduction, the expert system proposed in this paper may be useful for students and job candidates to recognize their abilities in order to choose the best way of education and most suitable job positions. On the other hand, the expert system may serve as a good tool that supports people who have to decide concerning an educational program to offer for individual students or to select best candidates for a specific job. In both cases, they should take into account personal characteristics of the people. Thus, the expert system may be implemented in e-learning platforms, in order to adjust a special way of education for distance learning students, based on their face pictures from web cameras. Such a system can also analyze pictures attached to job candidates' CVs in order to get some information about those people - from the "face reading" - that may be very helpful along with the interview.

The idea and methodology proposed in this paper may also be applied in the problem opposite to the personal traits recognition. Namely, the knowledge concerning the psychologic characteristics visible on persons' faces, and formulated as a fuzzy rule base, can be used in order to create avatars with desired personal features. This idea seems to be especially useful in application to design virtual assistants for various companies, as well as specific characters in computer games.

In both kinds of applications, i.e. the expert system for personality traits recognition, and the system created to design a specific virtual characters, it is assumed that some personal features are visible on human faces. Although many people are sceptical concerning this assumption, the publications mentioned in the Introduction confirm successful results in human personality recognition based on face pictures. Thus, it is worth scientifically developing this subject. Therefore, the idea of the hybrid intelligent system is presented and future research directions are proposed in this paper.

The research is concerned in application of artificial intelligence methods, such as neural networks and fuzzy systems. Evolutionary algorithms [18] can also be applied, e.g. to create avatars possessing best features according to a specific criterion. In such a case, the criterion should be appropriately formulated to play the role of a fitness function that is employed to evaluate individuals generated through the evolution procedure. During this optimization method, the best individual is produced as the result that represents the avatar with the best features encoded in his face.

This paper presents a general idea of the hybrid expert system for human personality characteristics recognition. As mentioned in the Introduction, there are some quite successful applications concerning face reading. However, the concept of the intelligent system proposed in order to recognize personality traits should be viewed as interesting subject that needs further scientific and experimental research.

The main idea of the hybrid expert system, proposed in this paper, concerns a wide aspect of various approaches combined in one intelligent system. At the first step, such a system may be employed in identification tasks where special facial characteristics are significant to be recognized. Among others, a degree of face asymmetry seems to be very important (see also e.g. [16]). Therefore, some approaches to determine face symmetry axis, construct composites built of left-left and right-right parts of a human face, and measure the degree of asymmetry (e.g. [19]) may be especially useful.

Then, such approaches should be developed in order to be applied in personality traits recognition - that is considered at the next step of the hybrid expert system proposed in this paper. The expert system may employ the face asymmetry recognition approach along with other face reading and artificial intelligence methods. The author's expectation is that such a system will be able to recognize a person from a picture of his/her face, taking into account special facial characteristics, including the degree of face asymmetry.

Results presented in [26] seems to confirm that there is a relation between activities of brain hemispheres and face asymmetry. The following statement says: in view of the contralateral control of two hemifaces (below eyes by two hemispheres of a person's brain, the two sides of the face undergo muscular development creating facial asymmetry (of course, other factors, such a gender, may also affect facial asymmetry)). Therefore, the intelligent expert system, proposed in this paper, may be extended for abilities to discover human personality characteristics based on facial asymmetry. It should be reminded that such a relation has also been analyzed by Anuashvili [2].

Moreover, the intelligent expert system should generate a description of a person (in a natural language) concerning both special facial characteristics and psychological traits.

## References

1. A Human Face Physiognomy Navigator. DataFace: Psychology, Appearance and Behavior of the Human Face. The Nature of Physiognomy, <http://www.face-and-emotion.com/dataface/physiognomy/>
2. Anuashvili, A.N.: Fundamentals of Objective Psychology. In: International Institute of Control, Psychology and Psychotherapy, Warsaw, Moscow (2005 - 4th edn., 2008 - 5th edn.) (in Russian)
3. Authentic Chinese Art of Face Reading Physiognomy, <http://www.facereading.cx/>
4. Blackford, K.M.H., Newcombe, A. (eds.): Character Reading at Sight. Nabu Press (1922); (2010 - reproduction)
5. Brown, S.G.: Face Reading: Secrets of the Chinese Masters. Sterling Publ. (2008)

6. Cordingley, B.: In Your Face: What Facial Features Reveal About the People You Know and Love. New Horizon Publ. (2001)
7. Digital Physiognomy Software, <http://www.uniphiz.com/physiognomy.htm>
8. Fulfer, M.: Amazing Face Reading (1994, 1996); Polish Edition, RM, Warsaw (2006)
9. Gray, P.: Psychology, 5th edn. Worth Publ. Boston (2007)
10. Haner, J.: The Wisdom of Your Face: Change Your Life with Chinese Face Reading. Hay House (2008)
11. Jackson, P.: Introduction to Expert Systems, 2nd edn. Wiley, Chichester (1990)
12. Jones, E.V.: Personology Research and Development Center, Inc., <http://www.personology.com/>
13. Kacprzyk, J., Zadrozny, S.: Computing with Words for Text Categorization. Springer, Heidelberg (2007)
14. Kanto, E., Kanto, I.: Your Face Tells All: Learn the Wisdom of the Chinese Art of Face Reading. Atophill Publ. (2005)
15. Lavater, J.C.: Physiognomy. London (MDCCCXXVI)
16. Liu, Y.: Understanding the role of facial asymmetry in human face identification. *Statistics and Computing* 17, 57–60 (2007)
17. Mar, T.T.: Face Reading: The Chinese Art of Physiognomy. Dodd, Mead. New York (1974)
18. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. Springer, Heidelberg (1992)
19. Milczarski, P., Kompanets, L., Kurach, D.: An approach to brain thinker type recognition based on facial asymmetry. In: Rutkowski, L., et al. (eds.) ICAISC 2010, Part I. LNCS (LNAI), vol. 6113, pp. 643–650. Springer, Heidelberg (2010)
20. Myers-Briggs, I., Myers, P.: Gifts Differing: Understanding Personality Type. Davies-Black Publ. (1995)
21. Plumb, K.W.: BrainChanger Institute, <http://www.brainchanger.com/>
22. Roberts, B.: Face Reading: How to Know Anyone at a Glance. USA (2009), <http://facereading1.com/>
23. Rosetree, R.: The Power of Face Reading, Canada (2001)
24. Russel, J.A.: Is the universal recognition of emotion from facial expression? A review of the cross-cultural studies. *Psychological Bulletin*. 115(1), 102–141 (1994)
25. Rutkowska, D.: Neuro-Fuzzy Architectures and Hybrid Learning. Springer, Heidelberg (2002)
26. Smith, W.M.: Hemispheric and facial asymmetry: Faces of Academe. *Journal of Cognitive Neuroscience* 10(6), 663–667 (1998)
27. Young, L.: The Naked Face: The Essential Guide to Reading Faces. St. Martin's Press (1994)
28. Zadeh, L.A.: Fuzzy sets. *Information and Control* 8, 338–353 (1965)
29. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning. *Information Science*. Part I, 8, 199–249. Part II, 8, 301–357. Part III, 9, 43–80 (1975)
30. Zadeh, L.A.: The role of fuzzy logic in the management of uncertainty in expert systems. *Fuzzy Sets and Systems* 11, 199–227 (1983)
31. Zadeh, L.A.: From computing with numbers to computing with words - from manipulation of measurements to manipulation of perceptions. *IEEE Trans. on Circuits and Systems - I: Fundamental Theory and Applications*. 45(1), 105–119 (1999)
32. Zurada, J.M.: Introduction to Artificial Neural Systems. West Publishing Company (1992)

# Author Index

- Aizenberg, Igor II-3  
Akita, Masatoshi II-355, II-451  
Alba, Enrique II-428  
Alpaydin, Ethem I-430  
Ashibe, Maiko II-411  
Augustyniak, Piotr I-581  
Ayouni, Sarra I-267
- Babenyshev, Sergey II-337  
Baczyński, Michał I-3  
Bandholtz, Sebastian II-132  
Barcellos de Paula, Luciano II-461  
Barloy, Yann II-363  
Barszcz, Tomasz II-11  
Bartczuk, Łukasz I-224, I-275 II-445  
Bartosiewicz, Pavel II-659  
Batet, Montserrat I-281  
Bednarska, Urszula II-622  
Beydoun, Ghassan II-614  
Bezdek, James C. I-363  
Bielecka, Marzena I-11  
Bielecki, Andrzej I-589, II-11  
Bielskis, Antanas Andrius I-605  
Biesiada, Jacek I-289, I-388  
Bilski, Jarosław II-19  
Blachnik, Marcin I-289, I-388,  
I-414, II-371  
Bouvry, Pascal II-428  
Bożejko, Wojciech II-379, II-387, II-395  
Braga, Rodrigo A.M. II-239  
Buche, Cédric I-299  
Bukowiec, Adam I-289  
Burczynski, Tadeusz II-500  
Butkiewicz, Bohdan S. I-19
- Cable, Baptiste II-363  
Cader, Andrzej II-468  
Calzada, Alberto I-202  
Cano, Isaac II-403  
Chomski, Jarosław I-185  
Cichocki, Andrzej I-563  
Cierniak, Robert I-505  
Ciskowski, Piotr I-307  
Cislariu, Mihaela I-27  
Colombo, Armando W. II-313
- Costa Jr., Sergio Oliveira I-35  
Cpałka, Krzysztof I-43, II-645  
Cunha, Frederico M. II-239  
Czajkowski, Marcin II-157  
Czapiński, Michał II-379  
Czoków, Maja II-420  
Czubenko, Michał II-516
- Dahal, Keshav I-216  
Dang, Thi-Hai-Ha II-247  
De Loor, Pierre I-299  
Denisov, Vitalij I-605  
Derbel, Imen I-97  
Derlatka, Marcin I-597  
Dorronsoro, Bernabé II-428  
Dralus, Grzegorz II-26  
Drozda, Stanisław I-74  
Drungilas, Darius I-605  
Duch, Włodzisław I-347, I-388, I-445  
Dudek, Damian I-315  
Dudek, Grzegorz II-437  
Du, Juan I-49, I-58  
Dymova, Ludmila I-66  
Dzemydienė, Dalė I-605  
Dziwiński, Piotr I-224, I-275, II-445
- Er, Meng Joo I-43, I-49, I-58, II-645
- Frischmuth, Kurt I-120
- Gabryel, Marcin I-74, II-143  
Genoe, Ray I-80  
Gibert, Karina I-281  
Gierlak, Piotr II-256  
Giesler, Björn I-547  
Gjorgjevikj, Dejan I-437  
Gongora, Mario A. II-492  
Gorawski, Marcin I-323  
Gordan, Mihaela I-27  
Gorzalczany, Marian B. I-88  
Gras, Robin I-487  
Grąbczewski, Krzysztof I-331, I-380  
Grochowski, Marek I-347  
Gromisz, Marcin I-339  
Grötzinger, Carsten II-132



- Grudziński, Karol I-347  
 Grzanek, Konrad II-468  
 Grzymala-Busse, Jerzy W. I-355  
  
 Haase, Sven I-479  
 Hachani, Narjes I-97  
 Hammer, Barbara I-479  
 Hasiewicz, Zygmunt II-34  
 Havens, Timothy C. I-363  
 Hawarah, Lamis I-372  
 Hendzel, Zenon II-264  
 Hidalgo, J. Ignacio II-582  
 Hirose, Akira II-42  
 Hiroyasu, Tomoyuki II-173, II-355,  
 II-411, II-451  
 Homenda, Władysław II-476  
 Hong, Long I-571  
 Hoppenot, Philippe II-247  
 Hrebień, Maciej II-484  
 Hussain, Alamgir I-216  
 Hutzler, Guillaume II-247  
  
 Ignor, Tomasz II-698  
 Ilnatouski, Mikhail I-597  
 Irvine, David II-492  
 Iwanowicz, Piotr II-165  
  
 Jacomino, Mireille I-372  
 Jahankhani, Pari I-635  
 Jankowski, Norbert I-331, I-380  
 Jarosz, Paweł II-500  
  
 Kacalak, Wojciech II-508  
 Kachel, Adam I-388  
 Kacprzyk, Janusz I-105, I-232  
 Kaku, Fumiya II-451  
 Kapuscinski, Tomasz II-272  
 Kaya, Heysem I-397  
 Kechadi, Tahar I-80  
 Kecman, Vojislav I-613  
 Keller, James M. I-363  
 Kida, Naoto II-451  
 Kikec, Mirna I-613  
 Kitowski, Jacek II-532  
 Klepaczko, Artur II-149  
 Klęsk, Przemysław I-405  
 Knoll, Alois I-547  
 Kobyliński, Łukasz I-515  
 Kodogiannis, Vassilis I-635  
 Kompanets, Leonid I-643  
  
 Korbicz, Józef II-484  
 Korczak, Oskar I-160  
 Kordos, Mirosław I-289, I-414  
 Korkosz, Mariusz I-589  
 Korytkowski, Marcin I-74, I-114, I-621  
 Kosiński, Witold I-120  
 Kosiorowski, Przemysław II-689  
 Kotulski, Leszek II-280  
 Kowalczyk, Zdzisław II-516  
 Krętowska, Małgorzata II-524  
 Krętowski, Marek II-157  
 Król-Korczak, Jadwiga I-11  
 Krüger, Lars I-128  
 Krzyżak, Adam I-422  
 Kubanek, Mariusz I-523  
 Kuczyński, Karol I-627  
 Kudelski, Michał II-289  
 Kühne, Ronald II-132  
 Kunene, Niki I-495  
 Kurach, Damian I-643  
 Kurşun, Olcay I-397  
 Kurek, Jerzy E. II-321  
 Kursun, Olcay I-430  
 Kusiak, Jan II-80  
 Kuta, Marcin II-532  
 Kwedło, Wojciech II-165  
  
 Lafuente, Anna María Gil II-461  
 Lara, Juan A. I-635  
 Laskowski, Łukasz II-47  
 Laurent, Anne I-267  
 Ławryńczuk, Maciej II-297, II-305  
 Lech, Piotr II-329  
 Leitão, Paulo II-313  
 Lewandowski, Roman II-651  
 Liu, Jun I-202  
 Lorette, Sophie II-363  
 Lucińska, Małgorzata II-540  
 Lu, Jie I-194  
  
 Mączka, Krystian II-371  
 Madzarov, Gjorgji I-437  
 Magaj, Janusz I-185  
 Majewski, Maciej II-508  
 Mańdziuk, Jacek II-667  
 Manyakov, Nikolay V. II-548  
 Marciniak, Jakub II-556  
 Marepally, Shantanu R. I-355  
 Martínez, Luis I-202  
 Marusak, Piotr M. I-136

- Marvuglia, Antonino I-224  
Maszczyk, Tomasz I-445  
Materka, Andrzej II-149  
Matsumoto, Toshiko II-566  
Matuszak, Michał II-574  
Mazurkiewicz, Jacek II-675  
Mendes, J. Marco II-313  
Miki, Mitsunori II-173, II-355,  
II-411, II-451  
Mikrut, Zbigniew I-531  
Milczarski, Piotr I-643  
Millán-Ruiz, David II-582  
Miller, Bartosz II-590  
Min, Ji-Hee I-240  
Mitsubishi, Takashi I-144  
Mourelle, Luiza de Macedo I-35  
Możaryn, Jakub II-321  
Myszkorowski, Krzysztof I-152  
Mzyk, Grzegorz II-34
- Naftulin, Igor S. I-651  
Nalepa, Grzegorz J. II-598  
Nawarycz, Tadeusz II-197  
Nazarko, Piotr II-56  
Nedjah, Nadia I-35  
Neruda, Roman II-124  
Niewiadomski, Adam I-160  
Nigro, Jean-Marc II-363  
Nishida, Takeshi II-451  
Nishimoto, Tatsuo II-451  
Nowicki, Robert K. I-168
- Oba, Mitsuharu II-566  
Obuchowicz, Andrzej II-181  
Okada, Noriko II-173  
Okarma, Krzysztof I-539, II-329  
Olchowy, Marcin I-175  
Onoyama, Takashi II-566  
Osinski, Jędrzej II-606  
Oszust, Mariusz II-189  
Othman, Siti Hajar II-614  
Ounelli, Habib I-97
- Pacheco, Jorge II-582  
Pacut, Andrzej II-289  
Paradowski, Mariusz I-555  
Patton, Robert M. I-657  
Pérez, Aurora I-635  
Piech, Henryk II-622  
Pieczyński, Andrzej II-630
- Piegat, Andrzej I-175  
Piekiewski, Filip II-64  
Pietrzykowski, Zbigniew I-185  
Pires, Matheus Giovanni II-72  
Ploix, Stéphane I-372  
Pluciennik-Psota, Ewa I-323  
Poelmans, Jonas II-548  
Pokropinska, Agata I-74  
Poncelet, P. I-267  
Potok, Thomas E. I-657  
Prętki, Przemysław II-181  
Prodan, Lucian II-205  
Przybył, Andrzej II-645  
Przybyszewski, Krzysztof II-638  
Purba, Julwan H. I-194  
Pytel, Krzysztof II-197
- Rafajłowicz, Ewaryst I-422, I-453  
Rauch, Łukasz II-80  
Rebrova, Olga Yu. I-651  
Reis, Luis P. II-239  
Restivo, Francisco II-313  
Robak, Silva II-630  
Rodríguez, Rosa M. I-202  
Rojek, Izabela II-88  
Ruan, Da I-194, I-202  
Rudziński, Filip I-88  
Ruican, Cristian II-205  
Rusiecki, Andrzej II-96  
Rutkowska, Danuta I-665  
Rutkowski, Leszek I-43, I-49, I-58,  
I-621, II-143, II-645  
Rybakov, Vladimir II-337
- Salehi, Elham I-487  
Sędziwy, Adam II-280  
Scherer, Rafał I-74, I-114, I-210, I-621  
Schleif, Frank-Michael I-479  
Schmidt, Adam II-651  
Schönherr, Kristin I-547  
Schreiber, Tomasz II-420, II-574  
Şeker, Hüseyin I-397  
Sevastjanov, Pavel I-66, II-659  
Shidama, Yasunari I-144  
Siczek, Maciej I-627  
Silva, Ivan Nunes da II-72  
Skrzypczyk, Krzysztof II-345  
Skubalska-Rafajłowicz, Ewa I-462  
Śliwiński, Przemysław II-34  
Słowik, Adam II-213

- Śluzek, Andrzej I-555  
 Smolağ, Jacek II-19  
 Sowan, Bilal I-216  
 Starczewski, Janusz T. I-168, I-224,  
 I-275, II-445  
 Stegierski, Rafał I-627  
 Strzempa, Dawid I-414  
 Suchorzewski, Marcin II-221  
 Suszyński, Waldemar I-627  
 Suyanto, II-229  
 Szarek, Arkadiusz I-621  
 Szmidt, Eulalia I-232  
 Sztangret, Lukasz II-80  
 Szupiluk, Ryszard I-471  
 Szuster, Marcin II-256, II-264  
  
 Tadeusiewicz, Ryszard II-104  
 Tanaka, Shingo II-451  
 Tanisawa, Junichi II-451  
 Tatjewski, Piotr II-305  
 Tkacz, Kamil II-659  
 Tomishima, Chitose II-411  
 Torra, Vicenç I-240, II-403  
  
 Uchroński, Mariusz II-387, II-395  
 Udrescu, Mihai II-205  
  
 Valente, Juan P. I-635  
 Valls, Aida I-281  
 Van Hulle, Marc M. II-548  
 Vélez, José L. II-582  
 Verstraete, Jörg I-248  
 Vidnerová, Petra II-124  
 Villmann, Thomas I-479  
 Vladutiu, Mircea II-205  
 Vlaicu, Aurel I-27  
 Vogels, Rufin II-548  
  
 Wałędzik, Karol II-667  
 Walczak, Krzysztof I-515  
 Walkowiak, Tomasz II-675  
 Wąs, Jarosław II-683  
 Waszczyszyn, Zenon II-590  
 Wiak, Sławomir II-689  
 Wichard, Jörg D. II-132  
 Wieczorek, Tadeusz II-371  
 Wietrzych, Jerzy I-453  
 Wilbik, Anna I-105  
 Wilczyńska-Sztyma, Dorota I-120  
 Wiliński, Antoni I-405  
 Wodecki, Mieczysław II-379, II-395  
 Wojciechowski, Wadim I-589  
 Wójcik, Mateusz II-11  
 Wojewnik, Piotr I-471  
 Wolejsza, Piotr I-185  
 Wysocki, Marian II-189  
  
 Yahia, Sadok Ben I-267  
 Yang, Duanduan I-555  
 Yang, Qin I-487  
 Yoshimi, Masato II-173, II-355,  
 II-411, II-451  
  
 Zabkowski, Tomasz I-471  
 Zadrozny, Sławomir I-339  
 Zajdel, Roman I-256  
 Zaton, Marek I-307  
 Zdunek, Rafał I-563  
 Zdunek, Rafał II-698  
 Zhang, Guangquan I-194  
 Zhou, Ning-Ning I-571  
 Zieliński, Bartosz I-589  
 Ziemiański, Leonard II-56, II-590  
 Zurada, Jacek M. II-508  
 Zurada, Jozef I-495  
 Żylski, Wiesław II-256