Springer Optimization and Its Applications 202

Boris Goldengorin
Sergei Kuznetsov *Editors*

Data Analysis and Optimization

In Honor of Boris Mirkin's 80th Birthday

Springer

Springer Optimization and Its Applications

Volume 202

Aims and Scope

Optimization has continued to expand in all directions at an astonishing rate. New algorithmic and theoretical techniques are continually developing and the diffusion into other disciplines is proceeding at a rapid pace, with a spot light on machine learning, artificial intelligence, and quantum computing. Our knowledge of all aspects of the field has grown even more profound. At the same time, one of the most striking trends in optimization is the constantly increasing emphasis on the interdisciplinary nature of the field. Optimization has been a basic tool in areas not limited to applied mathematics, engineering, medicine, economics, computer science, operations research, and other sciences.

The series **Springer Optimization and Its Applications (SOIA)** aims to publish state-of-the-art expository works (monographs, contributed volumes, textbooks, handbooks) that focus on theory, methods, and applications of optimization. Topics covered include, but are not limited to, nonlinear optimization, combinatorial optimization, continuous optimization, stochastic optimization, Bayesian optimization, optimal control, discrete optimization, multi-objective optimization, and more. New to the series portfolio include Works at the intersection of optimization and machine learning, artificial intelligence, and quantum computing.

Volumes from this series are indexed by Web of Science, zbMATH, Mathematical Reviews, and SCOPUS.

Boris Goldengorin • Sergei Kuznetsov
Editors

Data Analysis and Optimization

In Honor of Boris Mirkin's 80th Birthday

Springer

Editors

Boris Goldengorin
Department of Mathematics
New Uzbekistan University
Tashkent, Uzbekistan

Center for Applied Optimization
University of Florida
Gainesville, FL, USA

Sergei Kuznetsov
Faculty of Computer Science
National Research University Higher
School of Economics
Moscow, Russia

ISSN 1931-6828 ISSN 1931-6836 (electronic)
Springer Optimization and Its Applications
ISBN 978-3-031-31653-1 ISBN 978-3-031-31654-8 (eBook)
https://doi.org/10.1007/978-3-031-31654-8

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Paper in this product is recyclable.

Preface

This book presents the state of the art in the emerging field of data science which includes models for layered security, protection of large gathering sites, cancer diagnostics, self-driving cars and other applications with catastrophic consequences of wrong decisions. The manipulability of aggregation procedures for the case of large numbers of voters is analyzed from a theoretical point of view and justified by computational experiments involving at least an order of magnitude larger number of voters. Many tree-type structures are considered: from phylogenetic trees representing the main patterns of vertical descent through consensus trees and super- trees widely used in evolutionary studies to combine phylogenetic information contained in individual gene trees. The statistical part of this book studies an impact of data mining and modeling on predictability assessment of time series. New notions of mean values based on ideas of multicriteria optimization are compared to their conventional definitions leading to fresh algorithmic approaches. To summarize, the book presents methods for automated analysis of patterns and models for data of different nature with applications ranging from scientific discovery to business intelligence and analytics. The style of the written chapters allows to recommend this book for senior undergraduate and graduate data mining courses providing a broad yet in-depth review integrating novel concepts from machine learning and statistics. The main parts of the book include exploratory data analysis, pattern mining, clustering, and classification supported by real life case studies.

Students and professionals specializing in computer and management science, data mining for high-dimensional data, complex graphs, and networks will benefit from many cutting-edge ideas and practically motivated case studies.

From a technical point of view, each paper has received at least two referee reports, and almost all papers are presented at the International conference "Data Analysis, Optimization and Their Applications" on the occasion of Boris Mirkin's 80th birthday, January, 30–31, 2023. Dolgoprudny, Moscow Region, Moscow Institute of Physics and Technology.

With the purpose to keep the atmosphere of that conference, we have included the Book of Abstracts and its schedule. We would like to thank all speakers and contributors to the conference and volume dedicated to Boris Mirkin's 80th birthday. Without the crucial support on organizational issues, the conference and the book could not have become a reality. Here is the list of our supporters and reviewers:

Fuad Aleskerov, NRU HSE, Russia
Daniel Berend, Ben Gurion University, Israel
Jean Diatta, University of la Réunion, France
Trevor Fenner, University of London, UK
Sergei Frenkel, FRC Computer Science and Control, Russia
Alexander Gasnikov, MIPT, Russia
Boris Goldengorin, University of Florida, USA
Eugene Koonin, NIH/NLM/NCBI, USA
Sergei Kuznetsov, NRU HSE, Russia
Boris Kovalerchuk, Central Washington University, USA
Leonid Litinskii, ISA RAS, Russia
Vladimir Makarenkov, University of Quebec in Montreal, Canada
Magomed Malsagov, ISA RAS, Russia
Boris Mirkin, NRU HSE, Russia and UL London, UK
Susana Nascimento, New University of Lisbon, Portugal
Alexandre Orlov, Bauman Moscow State Technical University, Russia
Panos Pardalos, University of Florida, USA
Natalia Pavlova, MIPT, Russia
Andrei Raigorodskii, MIPT, Russia
Alexey Remizov, MIPT, Russia
Fred Roberts, Rutgers University, USA
Yuri Sidel'nikov, MAI, Russia
Konstantin Vorontsov, Lomonosov MSU, Russia

Tashkent, Uzbekistan Boris Goldengorin
Gainesville, FL, USA
Moscow, Russia Sergei Kuznetsov

Program and Abstract Book

International Conference “Data Analysis, Optimization and Their Applications” on the Occasion of Boris Mirkin’s 80th Birthday

(January 30–31, 2023. Dolgoprudny, Moscow Region, Moscow Institute of Physics and Technology)

https://mipt.ru/education/chairs/dm/conferences/data-analysis-optimization-and-their-applications-2023.php

The conference is organized in honor of professor Boris Mirkin’s 80th birthday to celebrate his contributions in the fields of Data Science that he introduced or extensively explored: Anomalous Cluster, Bi-cluster, Categorical Factor Analysis, Chain Order Partition, Complementary Partition Criterion, Core-Shell Cluster, Data Recovery Approach, Distance Between Partitions, Federation Consensus Rule, Generalization in Taxonomy, Interval Order, Linear Stratification, Mapping Between Evolutionary Trees, Minkowski Weighted Feature Clustering, Parsimonious Gene History Reconstruction, Single Cluster Clustering, Structured Partition, Taxonomic Rank of Results, Tri-clustering.

Organization Committee

Fuad Aleskerov, NRU HSE, Russia
Trevor Fenner, University of London, UK
Alexander Gasnikov, MIPT, Russia
Fred Roberts, Rutgers University, USA
Boris Goldengorin, University of Groningen, The Netherlands
Sergei Kuznetsov, NRU HSE,Russia, Co-Chair
Boris Kovalerchuk, Central Washington University, USA
Vladimir Makarenkov, University of Quebec in Montreal, Canada
Panos Pardalos, University of Florida, USA

Andrei Raigorodskii, MIPT, Russia, Chair
Konstantin Vorontsov, Lomonosov MSU, Russia

Invited Speakers

Fuad Aleskerov, NRU HSE, Russia
Aleksandr Beznosikov, MIPT, Russia
Tendai Chikake, MIPT, Russia
Fred Roberts, Rutgers University, USA
Boris Goldengorin, University of Groningen, The Netherlands
Alexander Karpov, NRU HSE, Russia
Yakov Karandashev, RUDN, Russia
Eugene Koonin, National Center for Biotechnology Information, USA
Boris Kovalerchuk, Central Washington University, USA
Sergei Kuznetsov, NRU HSE,Russia
Alexander Lepskiy, NRU HSE, Russia
Guy Leshem. Ashkelon Academic College, Israel
Vladimir Makarenkov, University of Quebec in Montreal, Canada
Irina Maximova, RUDN, Russia
Boris Mirkin, NRU HSE, Russia and UL London, UK
Susana Nascimento, New University of Lisbon, Portugal
E. Dov Neimand, Stevens Institute of Technology, USA
Maria Pilgun, Russian State Social University, Russia
Maria Poptsova, NRU HSE, Russia
Alexander Rubchinsky, NRU HSE, Russia
Alexey Samosyuk, MIPT, Russia
Soroosh Shalileh, NRU HSE, Russia
Zina Taran, Delta State University, USA
Yuliya A. Veselova, NRU HSE, Russia
Konstantin Vorontsov, Lomonosov MSU, Russia

Schedule

Monday, 30 January 2023

Time	**Speaker, Table of Contents**
10:30–10:55	Conference Opening
11:00–11:25	Sergei Kuznetsov
11:30–11:55	Boris Mirkin
12:00–12:25	Soroosh Shalileh
	Coffee break
13:00–13:25	Guy Leshem
13:30–13:55	Yakov Karandashev
14:00–14:25	Maria Pilgun
	Lunch
15:00–15:25	Konstantin Vorontsov
15:30–15:55	Maria Poptsova
16:0016:25	Fred Roberts
	Coffee break
17:00–17:25	Eugene Koonin
17:30–17:55	Boris Goldengorin
18:00–18:25	Vladimir Makarenkov
	19:00 Conference Dinner, https://theorybar.ru/

Tuesday, 31 January 2023

Time	**Speaker**
11:00–11:25	Fuad Aleskerov
11:30–11:55	Alexander Lepskiy
12:00–12:25	Irina Maximova
	Coffee break
13:00–13:25	Yuliya A Veselova
13:30–13:55	Tendai Chikake
14:00–14:25	Alexey Samosyuk
	Lunch
15:00–15:25	Alexander Rubchinsky
15:30–15:55	Susana Nascimento
16:00–16:25	Dmitry Frolov
	Coffee break
17:00–17:25	Aleksandr Beznosikov
17:30–17:55	E Dov Neimand
18:00–18:25	Boris Kovalerchuk
18:30–18:55	Alexander Karpov

Biclustering, n-Clustering and Formal Concept Analysis

Authors	**Sergei Kuznetsov**
Abstract	The term bi-clustering was coined by Boris Mirkin to denote grouping objects based not on a similarity or distance notion, but on commonality of shared attributes, so that two types of entities – objects and attributes – are grouped at the same type. Formal Concept Analysis, based on Galois connections and lattice of closed sets, proposes a model of a strict bi-cluster – called (formal) concept – where a subset of objects shares all attributes from a subset of binary attributes. The order on concepts is closely related to implicational dependencies, both of exact (like functional dependencies) and approximate (liked association rules) nature. Several well-known types of bi-clusters can be efficiently reduced to concepts, thus taking advantage of machinery developed for processing concepts. We consider some useful relaxations of concepts and their generalization to multidimensional and unstructured data.
Affiliation	National Research University Higher School of Economics, Moscow, Russian Federation
Contact	skuznetsov@hse.ru
Keywords	Bi-cluster, n-cluster, formal concept, lattice of closed sets, dependency, association rule

Clustering as Empirical Classification

Authors	**Boris Mirkin**
Abstract	Clustering can be considered within various frameworks: machine learning, data science, systems analysis. We consider it part of the science of classification established by Aristotle a while ago. Unfortunately, the science of classification is not well developed yet, although one may distinguish among its goals and structures. Classification goals include: domain structuring, relating different aspects of phenomena, and knowledge representation. Classification structures included those faceted, ranking, partition, typology, and hierarchy. We analyze the current state of clustering research with respect to classification structures and goals.
Affiliation	NRU HSE, Moscow, RU and UL London, UK,
Contact	bmirkin@hse.ru
Keywords	Clustering, Classification, Clustering goals

Classification Using Marginalized Maximum Likelihood Estimation and Black-Box Variational Inference

Authors	**Soroosh Shalileh**
Abstract	Based upon variational inference (VI), a new set of classification algorithms has recently emerged. This set of algorithms aims (A) to increase generalization power; (B) to decrease computational complexity. However, the complex math and implementation considerations have led to the emergence of black-box variational inference methods (BBVI). Relying on these principles, we assume the existence of a set of latent variables during the generation of data points. We subsequently marginalize the conventional maximum likelihood objective function w.r.t this set of latent variables and then apply black-box variational inference to estimate the model's parameters. We evaluate the performance of the proposed method by comparing the results obtained from the application of our method to realworld and synthetic data sets with those obtained using basic and state-ofart classification algorithms. We proceed and scrutinize the impact: (1) the existence of non-informative features at various dimensionalities, (2) the imbalanced data representation, (3) non-linear data sets, and (4) different data set size on the performance of algorithms under consideration. The results obtained prove to be encouraging and effective.
Affiliation	Center for Language and Brain, NRU HSE University, Moscow, Russian Federation
Contact	sr.shalileh@gmail.com
Keywords	Automatic differentiation, Classification with Black Box 25 Variational Inference, Variational Inference, Classification

Purifying Data by Machine Learning with Certainty Levels

Authors	**Shlomi Dolev, Guy Leshem**
Abstract	For autonomic computing, self-managing systems, and decision-making under uncertainty and faults, in many cases we are using machine learning models and combine them to solve any problem. This models uses a dataset, or a set of data items, and data item is a vector of feature values and a classification. In many cases, these data sets include outlier and/or misleading data items that were created by input device malfunctions or were maliciously inserted to lead the machine learning to wrong conclusions. A reliable machine learning model must be able to handle a corrupted data set; otherwise, a malfunctioning input device that corrupts a portion of the data set or malicious adversary may lead to inaccurate classifications. Therefore, the challenge is to find an effective method to evaluate and increase the certainty level of the learning process as much as possible. This work introduces the use of a certainty level measure to obtain better classification capability in the presence of corrupted or malicious data items. Assuming we know the data distribution, e.g., is a normal distribution (which is a reasonable assumption in a large amount of data items) and/or a known upper bound on the given number of corrupted data items, our techniques define a certainty level for classifications. Another approach that will be presented in this work suggests enhancing the random forest techniques (the original model was developed by Leo Breiman) to cope with corrupted data items by augmenting the certainty level for the classification obtained in each leaf in the forest. This method is of independent interest that of significantly improving the classification of the random forest machine learning technique in less severe settings.
Affiliation	Dolev Shlomi, Ben-Gurion University of the Negev, Israel; Department of Computer Science Ashkelon Academic College, Israel
Contact	dolev@cs.bgu.ac.il, gleshem2525@gmail.com
Keywords	Data corruption, PAC learning, Machine learning, Certainty level

Anomaly Detection with Neural Network Using a Generator

Authors	**A.S. Markov, E.Yu. Kotlyarov, N.P. Anosova, V.A. Popov, Yakov Karandashev, D.E. Apushkinskaya**
Abstract	This paper concerns with the problem of detecting anomalies on X-ray images taken by full-body scanners (FBS). Our previous work describes the sequence of image preprocessing methods used to convert the original images, which are produced with FBS, to images with visually distinguishable anomalies. In this paper, we focus on development of the proposed methods, including the addition of preprocessing methods and the creation of generator which can produce synthetic anomalies. Examples of processed images are given. The results of using a neural network for anomaly detection are shown.
Affiliation	Peoples' Friendship University of Russia (RUDN University), Moscow, Russian Federation
Contact	to.asmarkov@gmail.com,tyztot@gmail.com,anosova-np@rudn.ru, popov-va@rudn.ru,karandashev@niisi.ras.ru,apushkinskaya@gmail.com
Keywords	Full-body scanner, X-ray image, anomaly detection, image histogram equalization, generator, neural network, U-Net

Data and Text Interpretation in Social Media: Urban Planning Conflicts

Authors	**Maria Pilgun, Nailia Gabdrakhmanova**
Abstract	The relevance of this study is determined by the need to develop technologies for effective urban systems management and resolution of urban planning conflicts. The paper presents an algorithm for analyzing urban planning conflicts on the example of data and text interpretation in social media. The material for the study was data from social networks, microblogging, blogs, instant messaging, forums, reviews, video hosting services, thematic portals, online media, print media, and TV related to the construction of the Big circle metro line (Southern section) in Moscow (Russian Federation). Data collection: 1 October 2020–10 June 2021. Number of tokens: 62 657 289. To analyze the content of social media, a multi-modal approach was used. The paper presents the results of research on the development of methods and approaches for constructing mathematical and neural network models for analyzing the social media users' perceptions based on the user generated content and on digital footprints of users. Artificial neural networks, differential equations, and mathematical statistics were involved in building the models. Differential equations of dynamic systems were based on observations enabled by machine learning. In combination with mathematical and neural network model, the developed approaches made it possible to draw a conclusion about the tense situation, identify complaints of residents to constructors and city authorities, and propose recommendations to resolve and prevent conflicts.
Affiliation	Russian State Social University, Moscow, Russian Federation, Peoples' Friendship University of Russia, Moscow, Russia, Russian Federation
Contact	pilgunm@yandex.ru,gabd-nelli@yandex.ru
Keywords	Time series, neural networks, stochastic process, differential equation, urban environment, social tension, social media

Rethinking Probabilistic Topic Modeling from the Point of View of Classical Non-Bayesian Regularization

Authors	**Konstantin Vorontsov**
Abstract	Probabilistic topic modeling with hundreds of its models and applications has been an efficient text analysis technique for almost twenty years. This research area has evolved mostly within the frame of the Bayesian learning theory. For a long time, the possibility of learning topic models with a simpler conventional (non-Bayesian) regularization remained underestimated and rarely used. The framework of additive regularization for topic modeling (ARTM) fills this gap. It dramatically simplifies the model inference and opens up new possibilities for combining topic models by just adding their regularizers. This makes the ARTM a tool for synthesizing models with desired properties and gives rise to developing the fast online algorithms in the BigARTM open-source environment equipped with a modular extensible library of regularizers. In this paper, a general iterative process is proposed that maximizes a smooth function on unit simplices. This process can be used as inference mechanism for a wide variety of topic models. This approach is believed to be useful not only for rethinking probabilistic topic modeling, but also for building the neural topic models increasingly popular in recent years.
Affiliation	Federal Research Center “Computer Science and Control” of RAS, M.V.Lomonosov Moscow State University (MSU), Moscow, Russian Federation
Contact	voron@mlsa-iai.ru
Keywords	Probabilistic Topic Modeling, Additive Regularization of Topic Models, EM-algorithm, BigARTM, Multimodal Topic Modeling, Hierarchical Topic Modeling, Hypergraph Topic Modeling, Sequential Topic Modeling, Topical Embedding, Transactional Data, Recommender Systems, Latent Dirichlet Allocation, Bayesian Learning.

Generating Genomic Maps of Z-DNA with the Transformer Algorithm

Authors	**Dmitry Umerenkov, Vladimir Kokh, Alan Herbert, Maria Poptsova**
Abstract	Z-DNA and Z-RNA were shown to play an important role in various processes of genome functioning acting as flipons that launch or suppress genetic programs. Genome-wide experimental detection of Z-DNA remains a challenge due to dynamic nature of its formation. Recently we developed a deep learning approach DeepZ, based on CNN and RNN architectures, that predicts Z-DNA regions using additional information from omics data collected from different cell types. Here we took advantage of the transformer algorithm that trains attention maps to improve classifier performance. We started with pretrained DNABERT models and fine-tuned their performance by training with experimental Z-DNA regions from mouse and human genome wide studies. The resulting DNABERT-Z outperformed DeepZ. We demonstrated that DNABERT-Z finetuned on human data sets also generalizes to predict Z-DNA sites in mouse genome.
Affiliation	Sber Artificial Intelligence Lab, Moscow, Russian Federation Laboratory of Bioinformatics, Faculty of Computer Science, HSE University, Moscow, Russian Federation, InsideOutBio, Charlestown, MA, USA
Contact	mPoptsova@hse.ru
Keywords	non-B DNA structures, machine learning, deep learning, transformer, Z-DNA

Graph-Theoretical Models of the Spread and Control of Disease and of Fighting Fires

Authors	**Fred S. Roberts**
Abstract	We will describe irreversible threshold processes on graphs that model the spread of disease and lead to insights about strategies for vaccination, quarantine, etc.; we will describe models of the control of fires that are mathematically analogous to the disease spread models. The analogy will lead to insights about both types of processes and a variety of challenging graph-theoretical problems.
Affiliation	DIMACS, Rutgers University Piscataway, NJ USA
Contact	froberts@dimacs.rutgers.edu
Keywords	Graph-theoretical models, Disease spread models, Vaccination models, Threshold processes, The firefighter problem

The Last Universal Cellular Ancestor: What Have We Learned After 20 Years of Effort?

Authors	**Eugene Koonin**
Abstract	In 2003, Boris Mirkin and colleagues published a seminal on the Last Universal Cellular Ancestor (LUCA). In this work, a modified maximum parsimony approach was developed and applied to reconstruct the genome of the LUCA from a mapping of genes that are conserved across different ranges of extant organism on the universal phylogenetic tree. The size and composition of the reconstructed gene sets of the LUCA critically depended on the key parameter, namely, the gain penalty, or the ratio of the rates of gene gain, via emergence of new genes and horizontal gene transfer, to the rate of gene loss. As it can be expected, the size of the reconstructed gene sets grew with increasing gain penalty (g) such that the number of genes mapped to the LUCA varied from less than 600 for $g=1$ (gene gains considered as frequent as gene gains) to about 1700 for $g=10$ (gains considered 10 times less frequent than losses). During the 20 years elapsed since the publication of this work, several attempts to reconstruct the gene set of the LUCA using more sophisticated algorithms and many more genomes of prokaryotes have been undertaken, but the results of Mirkin and colleagues do not appear to have been superseded. At the time of the original publication, the high rate of horizontal gene transfer in bacteria and archaea had been just discovered, and a low gain penalty, leading to a simple LUCA, appeared most plausible. Subsequently, however a confluence of biological considerations was increasingly pointing towards a LUCA that was comparable in complexity to modern bacteria and archaea. The reconstructions obtained by Mirkin and colleagues in 2003 with the largest gain penalties might have been surprisingly accurate, at least in terms of the number of genes in the LUCA genome.
Affiliation	National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Bethesda, MD, USA
Contact	koonin@ncbi.nlm.nih.gov
Keywords	Last Universal Cellular Ancestor, Phylogenetic Trees, Maximum Parsimony, Gene Gain, Gene Loss

Pseudo-Boolean Polynomials, Dilworth Theorem and Data Aggregation

Authors	**Boris Goldengorin**
Abstract	In this talk, we use a pseudo-Boolean representation of the given matrix (two-dimensional table) with the purpose to aggregate (compress) its columns preserving their monotone behavior [2]. A natural application of Dilworth theorem (1950) [1] to the simplified polynomial returns the mi-nimum number of aggregated columns. Further, matrix compactification can be done by truncation columns depending on the number of rows fairly representing the original data [3]. We show that truncated pseudo-Boolean polynomials are invariants to represent clusters of equivalent data (matrices). The presented approach might be applied to 3- and multi-dimensional data sets. Promising preliminary experiments with the 4-dimensional Iris Flower dataset, 30-dimensional Wisconsin diagnostic breast cancer (WDBC), edge/blob detection, contour analysis, image segmentation and optical character recognition (OCR) are discussed in the talk of Tendai Chikake [4]. **References** 1. Robert P. Dilworth. A Decomposition Theorem for Partially Ordered Sets, Annals of Mathematics, 1950, **51** (1): 161–166, doi:10.2307/1969503. 2. B. F. AlBdaiwi, D. Ghosh, B. Goldengorin. Data Aggregation for p-Median Problems. Journal of Combinatorial Optimization, 2011, **3**(21), 348-363. 3. B.Goldengorin, D.Krushinsky, P.M. Pardalos. Cell Formation in Industrial Engineering. Theory and Computational Experiments. Springer, 2013 (chapter 2). 4. B.Goldengorin, T. Chikake. Pseudo-Boolean polynomials (pBp) for dimensionality reduction and image processing. **Book of Abstracts**. International conference "Data Analysis, Optimization and their Applications" on the occasion of Boris Mirkin's 80th birthday. MIPT, 2023. Page 23
Affiliation	University of Florida, USA; University of Groningen, NL and MIPT, Russian Federation
Contact	goldengorin@gmail.com
Keywords	Pseudo-Boolean Polynomials, Monotonicity, Data Aggregation, Dilworth Theorem

Inferring Multiple Consensus Trees and Super-Trees Using Clustering: A Review

Authors	**Vladimir Makarenkov, Gayane S Barseghyan, Nadia Tahiri**
Abstract	Phylogenetic trees (i.e. evolutionary trees, additive trees or X-trees) play a key role in the processes of modeling and representing species evolution. Genome evolution of a given group of species is usually modeled by a species phylogenetic tree that represents the main patterns of vertical descent. However, the evolution of each gene is unique. It can be represented by its own gene tree which can differ substantially from a general species tree representation. Consensus trees and supertrees have been widely used in evolutionary studies to combine phylogenetic information contained in individual gene trees. Nevertheless, if the available gene trees are quite different, then the resulting consensus tree or supertree can either include many unresolved subtrees corresponding to internal nodes of high degree or can simply be a star tree. This may happen if the available gene trees have been affected by different reticulate evolutionary events, such as horizontal gene transfer, hybridization or genetic recombination. In this case, the problem of inferring multiple alternative consensus trees or supertrees, using clustering, becomes relevant since it allows one to regroup in different cluster gene trees having similar evolutionary patterns (e.g. gene trees representing genes that have undergone the same horizontal gene transfer or recombination events). We critically review recent advances and methods in the field of phylogenetic tree clustering, discuss the methods' mathematical properties and describe the advantages and limitations of multiple consensus trees and supertree approaches. In the application section, we show how the discussed supertree clustering approach can be used to cluster aaRS evolutionary trees according to their evolutionary patterns.
Affiliation	Département d'Informatique, Université du Québec à Montréal, Case postale 8888, Succursale Centre-ville, Montreal, QC, H3C 3P8, Canada Département d'Informatique, Université de Sherbrooke, 2500 Boulevard de l'Université, Sherbrooke, Québec J1K 2R1, Canada
Contact	makarenkov.vladimir@uqam.ca
Keywords	Clustering, Cluster validity index, Consensus tree, k-means, k-medoids, Phylogenetic tree, Robinson and Foulds distance, Supertree

The Model of Tunnel Clustering and Its Applications

Authors	**Fuad Aleskerov, Darya Chubarova , Vyacheslav Yakuba**
Abstract	We propose a man-machine procedure for dynamic pattern analysis. This very model allows us to analyze the changes of clusters over time of objects of different nature based on ε-tube defined on the parameters of objects. We present several examples of a dynamic pattern-analysis; in particular, we consider the oil export/import operations among countries in 2005–2020.
Affiliation	HSE University, Moscow, Russian Federation, Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russian Federation, HSE University, Moscow, Russia, Institute of Control Sciences of Russian Academy of Sciences and HSE University, Moscow, Russian Federation
Contact	fa201204@gmail.com,dchubarova@hse.ru,rspp25@gmail.com
Keywords	Pattern-analysis, Clustering, Dynamic pattern-analysis, Oil export/import

About Some Clustering Algorithms in Evidence Theory

Authors	**Alexander Lepskiy**
Abstract	The Dempster–Shafer theory of evidence considers data that have a frequency-set nature (the so-called body of evidence). In recent years, there has been interest in clustering such objects to approximate them with simpler bodies of evidence, to analyze the inconsistency of information, reducing the computational complexity of processing algorithms, revealing the structure of the set of focal elements, etc. The article discusses some existing algorithms for clustering evidence bodies and suggests some new algorithms and approaches in such clustering.
Affiliation	National Research University Higher School of Economics (HSE), Moscow, Russian Federation
Contact	alepskiy@hse.ru
Keywords	Evidence Theory, Conflict Measure, Clustering Bodies of Evidence

Controllability of Triangular Systems with Phase Space Change

Authors	**Irina Maximova**
Abstract	In the present paper, the controllability of a composite system of the following structure is investigated: two phase spaces and two consecutive time intervals are given, in each space on the corresponding time interval the motion of the object is described by a nonlinear system. The phase spaces are changed with the help of some given mapping, and the docking of trajectories is also connected with it. The conditions of controllability of this system from the initial set of one space to the finite set of another space are obtained in the paper. An approach to finding trajectories for this motion is proposed.
Affiliation	S. M. Nikolsky Mathematical Institute, Peoples' friendship university of Russia (RUDN University), 6 Miklukho-Maklaya St, Moscow, 117198, Russian Federation
Contact	irismax@yandex.ru
Keywords	Controllability, full controllability, phase space change, triangular system.

Manipulation by Coalitions in Voting with Incomplete Information

Authors	**Yuliya A Veselova**
Abstract	We consider the problem of coalitional manipulation in collective decision making and a probabilistic approach for solving it. We assume that voters have some information about other voters' preferences from opinion polls held before voting. There are five different types of poll information functions. Coalition members are assumed to have identical preferences. We consider the probability that in a randomly chosen preference profile there exists a coalition which has an incentive to manipulate under a given type of poll information. We answer the following questions. How does coalitional manipulability differ from individual? How do different types of poll information affect coalitional manipulability? We answer these questions via both theoretical investigation and computational experiments.
Affiliation	National Research University Higher School of Economics (HSE University), Moscow, Russian Federation, Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russian Federation
Contact	yul-r@mail.ru
Keywords	Coalitions, manipulation, manipulability index, opinion poll, incomplete information

Pseudo-Boolean Polynomials (pBp) for Dimensionality Reduction and Image Processing

Authors	**Tendai Chikake, Boris Goldengorin**
Abstract	We introduce usage of a reduction property of penalty-based formulation of pseudo-Boolean polynomials as a mechanism for invariant dimensionality reduction in cluster analysis processes. In our experiments, we show that multidimensional data, like a 4-dimensional Iris Flower dataset, can be reduced to 2-dimensional space while a 30-dimensional Wisconsin Diagnostic Breast Cancer (WDBC) dataset can be reduced to 3-dimensional space, and through linear edge searches we can extract clusters in a linear and unbiased manner with competitive accuracies, reproducibility, and clear interpretation. We further showcase the exploitation of equivalence and polynomial degree for detecting gradient change in image data represented in greyscale matrices. These properties enable us to propose new methods of edge/blob detection, contour analysis, image segmentation and optical character recognition (OCR) using combinatorial techniques.
Affiliation	Department of Discrete Mathematics, Phystech School of Applied Mathematics and Informatics, Moscow Institute of Physics and Technology (MIPT) Dolgoprudny, Moscow Region, Russian Federation
Contact	tendaichikake@phystech.edu, goldengorin.bi@mipt.ru
Keywords	Pseudo-Boolean polynomials, equivalence, dimensionality reduction, cluster analysis, feature selection, outlier detection, interpretable clustering, edge/blob detection, contour analysis, image segmentation, OCR

Clicks, Random Graphs and Neural Networks

Authors	**Alexey Samosyuk, Shokorov Viacheslav**
Abstract	In this work, we present several computational experiments that demonstrate applications the Erdős-Rényi graph model and the Johnson–Lindenstrauss lemma to the evolution of neural networks during training and fine-tuning on the academia- and industry-level datasets (MS1M, WebFace42) and models (ShuffleNetv2, ResNet200). We show this on the 2 million classes FaceID problem, followed by the domain adaptation step.
Affiliation	Moscow Institute of Physics and Technology (MIPT) Dolgoprudny, Moscow Region, Russian Federation
Contact	alexeysamosyuk@gmail.com, slava.schokorov@gmail.com
Keywords	Neural Networks, Explainable, Random Graphs, Domain Adaptation

Algorithm of Trading on the Stock Market, Providing Satisfactory Results

Authors	**Alexander Rubchinsky, Kristina Baikova**
Abstract	The paper proposes a new trading algorithm for S&P-500 stock market, which provides positive results over a sufficiently long period of time. No assumptions are made about the behave or of this market (including probabilistic ones), and no predictions are made and used. The daily real stock price data are considered, and the gain (or loss) that would be obtained if the proposed stock choice algorithm for the next day was applied is calculated. The algorithm uses only the closing price data for the preceding days and includes a special stopping rule based on the income, accumulated since an initial day of a considered period till a current day. The suggested algorithm substantially uses the previously developed approach to construction a family of graph decompositions (see publications [1−3]).
Affiliation	National Research University HSE, Moscow, Russian Federation, National University of Science and Technology (MISIS), Moscow, Russian Federation
Contact	arubchinsky@yahoo.com, kristinkabaikova@gmail.com
Keywords	Stock market, graph decomposition, trading algorithm, cluster, stopping rule, secretary problem, fractal

Three-Stage Cluster Modeling for the Spatiotemporal Analysis of Coastal Upwelling

Authors	**Susana Nascimento, Alexandre Martins, Paulo Relvas, Joaquim F. Lu√≠s, Boris Mirkin**
Abstract	This work proposes a three-stage spatiotemporal clustering approach for the automatic recognition and analysis of coastal upwelling from Sea Surface Temperature (SST) grids derived from satellite images. The algorithm, core-shell clustering, models the upwelling as an evolving cluster whose core points are constant during a certain time window while the shell points move through an in-and-out binary sequence. The least squares minimization of clustering criterion allows to derive key parameters in an automated way. The algorithm is initialized with an extension of Seeded Region Growing offering self-tuning thresholding, the STSEC algorithm, that is able to precisely delineate the upwelling regions at each SST instant grid. Yet, the application of STSEC to the SST grids as temporal data puts the business of finding relatively stable “time windows”, here called “time ranges”, for obtaining the core clusters onto an automated footing. The approach successfully applies to SST image data for sixteen successive years of coastal upwelling of the Canary Current Upwelling System, covering two distinct regions: the Portuguese coast and the Morocco coast. The extracted time series of upwelling features presented consistent regularities among the upwelling seasons.
Affiliation	Dep. of Computer Science and NOVA Laboratory for Computer Science and Informatics (NOVA-LINCS) Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Lisbon, Portugal, Centre of Marine Sciences (CCMAR), Universidade do Algarve, Faro, Portugal, Instituto Dom Luís, Universidade de Lisboa, Lisboa Portugal, and Universidade do Algarve,Faro, Portugal Department of Data Analysis and AI, Higher School of Economics, Moscow, Russia, Department of Computer Science, Birkbeck University of London, London, UK)
Contact	snt@fct.unl.pt
Keywords	Spatiotemporal clustering, Time series segmentation, Data recovery clustering, SST images, Coastal upwelling

A Three-Step Method for Audience Extension in Internet Advertising using an Industrial Taxonomy

Authors **Dmitry Frolov, Zina Taran**

Abstract The paper addresses a very common problem in targeted digital advertising, insufficient audience size. Many approaches to audience extension frequently lead to much diminishing quality metrics, such as audience quality or conversion rates. This is the case, for example, for so-called lookalike techniques. We present a novel method for the efficient extension of target audiences. Our base is a popular taxonomy of user interests, the IAB contents taxonomy, combined with the representation of browsing behavior of millions of users by fuzzy sets of visited IAB taxonomy segments, that are leaves of the taxonomy tree. We use this idea in our method. The method consists of three steps: (1) computing membership values for the interest segments for a user by a classifier; (2) performing generalization of those sets and obtaining highranked segments, which is a core part of the method; (3) obtaining a set of advertising campaigns for a user. Our method involves an algorithm for optimally lifting individual fuzzy leaf sets into a higher rank taxonomy node, a so-called "head subject". The head subject must cover the input fuzzy leaf set in such a way that the number of errors is minimized. This algorithm was proposed as an intelligent information retrieval tool. It can be applied, however, to a very different task of targeted advertisement. To extend the audiences of a targeted advertisement, we find their head subjects off-line. Given a set of taxonomy segments corresponding to targeted audiences, we include a user as a target if her head subject covers any of those segments. This lifting-based step does increase the number of successful matches between user segments and campaign segments two- or three-fold without losing in the targeting quality, the click-through rate, CTR. This is in stark contrast to the conventional look-alike methods for increasing the audience numbers by reducing the admissibility thresholds, which leads to a large decrease in CTR, which was experimentally proven.

Affiliation Department of Data Analysis and Artificial Intelligence, HSE University, Moscow, Russian Federation, Division of Management, Marketing, and Business Administration, Delta State University, Cleveland, MS, USA

Contact dmitsf@gmail.com, ztaran@deltastate.edu

Keywords Targeted advertising, Audience extension, Taxonomy, Fuzzy leaf set, Optimal lifting

SARAH-Based Variance-Reduced Algorithm for Stochastic Finite-Sum Cocoercive Variational Inequalities

Authors	**Aleksandr Beznosikov, Alexander Gasnikov**
Abstract	Variational inequalities are a broad formalism that encompasses a vast number of applications. Motivated by applications in machine learning and beyond, stochastic methods are of great importance. In this paper, we consider the problem of stochastic finite-sum cocoercive variational inequalities. For this class of problems, we investigate the convergence of the method based on the SARAH variance reduction technique. We show that for strongly monotone problems it is possible to achieve linear convergence to a solution using this method. Experiments confirm the importance and practical applicability of our approach.
Affiliation	Moscow Institute of Physics and Technology (MIPT), Dolgoprudny, Moscow region, Russian Federation; HSE University, Moscow, Russian Federation, IITP RAS, Moscow, Russian Federation, Caucasus Mathematical Center, Adyghe State University, Maikop, Russian Federation
Contact	beznosikov.an@phystech.edu, gasnikov.av@mipt.ru
Keywords	Stochastic optimization, variational inequalities, finite sum problems

A Parallel Linear Active Set Method

Authors	**E. Dov Neimand, Şerban Sabău**
Abstract	Given a linear-inequality-constrained convex minimization problem in a Hilbert space, we develop a novel binary test that examines sets of constraints and passes only active-constraint sets. The test employs a blackbox, linear-equality-constrained convex minimization method but can often fast fail, without calling the black-box method, by considering information from previous applications of the test on subsets of the current constraint set. This fast fail, as a function of the number of dimensions, has quadratic complexity and can be completely multi-threaded down to near-constant complexity. Only when the test is unable to fast fail, does it use the blackbox method. In both cases, the test generates the optimal point over the subject inequalities. Iterative and largely parallel applications of the test over growing subsets of inequality constraints yields a minimization algorithm. We also include an adaptation of the algorithm for a non-convex polyhedron in Euclidean space. Outside of calling the black-box method, complexity is not a function of accuracy. The algorithm does not require the feasible space to have a non-empty interior, or even be nonempty. With ample threads, the multi-threaded complexity of the algorithm is constant as a function of the number of inequalities.
Affiliation	Department of Electrical and Computer Engineering, Stevens Institute of Technology
Contact	eneimand@stevens.edu, ssabau@stevens.edu
Keywords	Convex Optimization, Non-Convex Polyhedron, Hilbert Space, Strict Convexity, Parallel Optimization

Visual Explainable Machine Learning for High-Stake Decision-Making with Worst Case Estimates

Authors	**Charles Recaido, Boris Kovalerchuk**
Abstract	A major motivation for explaining and rigorous evaluating Machine Learning (ML) models is coming from high-stake decision-making tasks like cancer diagnostics, self-driving cars, and others with possible catastrophic consequences of wrong decisions. This paper shows that visual knowledge discovery (VKD) methods, based on the General Line Coordinates (GLC) recently developed, can significantly contribute to solving this problem. The concept of hyperblocks (n-D rectangles) as interpretable dataset units and GLC are combined to create visual selfservice machine learning models. Two variants of Dynamic Scaffold Coordinates (DSC) are proposed. It allows losslessly mapping high-dimensional datasets to a single two-dimensional Cartesian plane and building interactively an ML predictive model in this 2-D visualization space. Major benefits of DSC1 and DSC2 are their highly interpretable nature. They allow domain experts to control or establish new machine learning models through visual pattern discovery. It opens a visually appealing opportunity for domain experts, who are not ML experts, to build ML models as a self-service bringing the domain expertise to the model discovery, which increases model explainability and trust for the end user. DSC were used to find, visualize, and estimate the worst-case validation splits in several benchmark datasets, which is important for high-risk application. For large datasets, DSC is combined with dimensionality reduction techniques such as principal component analysis, singular value decomposition, and t-distributed stochastic neighbor embedding. A software package referred to as Dynamic Scaffold Coordinates Visualization System (DSCViz) was created to showcase the DSC1 and DSC2 systems.
Affiliation	Department of Computer Science, Central Washington University, USA
Contact	Charles.Recaido@cwu.edu, Boris.Kovalerchuk@cwu.edu
Keywords	Explainable machine learning, Visualization, Hyperblock, Multidimensional coordinate system, Self-service model.

Preferences over Mixed Manna

Authors	**Alexander Karpov**
Abstract	We define a new class of Condorcet domains, which are called GF-domains. GF-domains are unique Condorcet domains that are weakly minimally rich, semi-connected and contain a pair of mutually reverse preference orders. GF-domains are single-peaked on a circle that leads to a clear interpretation.
Affiliation	HSE University, Moscow, Russian Federation, Institute of Control Sciences Russian Academy of Sciences, Moscow, Russian Federation
Contact	akarpov@hse.ru
Keywords	Restricted domains, majority voting, single-peaked on a circle, circular city model

Contents

About the Editors

Boris Goldengorin is the author and inventor of data correcting and tolerance-based algorithms applied to many problems in operations research, supply chain management, quantitative logistics, industrial engineering, data and stock market analysis. Boris is the author of more than 100 articles published in leading international journals, including the *Journal of Algebraic Combinatorics*, *Discrete Optimization*, *Journal of Combinatorial Optimization*, *Journal of Global Optimization*, *Operations Research*, *Management Science*, *European Journal of Operational Research*, *Journal of Operational Research Society*, *Mathematical and Computer Modelling*, *Computers and Operations Research*, *Computers and Industrial Engineering*, *Expert Systems with Applications*, *Journal of Heuristics*, *Optimization Methods and Software*, *Computational Management Science*, and many others. Dr. Goldengorin has published four monographs, three textbooks, and an editor of five books on mathematical programming, game theory, combinatorial optimization, network analysis algorithms, graph theory, and big data analysis.

He is an associate editor of *Journal of Global Optimization*, *Journal of Combinatorial Optimization*, *SN Operations Research Forum*, and member of the Editorial Board of the *Journal of Computational and Applied Mathematics* of the Taras Shevchenko National University of Kyiv, Ukraine.

Sergei Kuznetsov graduated from the Faculty of Applied Mathematics and Control of the Moscow Institute for Physics and Technology in 1985. He obtained his Doctor of Science Degree in Theoretical Computer Science at the Computing Center of Russian Academy of Science. Since 2006, he is the Head of Department for Data Analysis and Artificial Intelligence, Head of the International Laboratory for Intelligent Systems and Structural Analysis, and Academic Supervisor of the Data Science master program at National Research University Higher School of Economics (Moscow). His research interests are in the algorithms of data mining, knowledge discovery, and formal concept analysis.

Optimal Layered Defense For Site Protection

Tsvetan Asamov, Emre Yamangil, Endre Boros, Paul B. Kantor, and Fred Roberts

We are pleased to dedicate this paper to our friend and colleague Boris Mirkin on the occasion of his 80th birthday.

1 Introduction

We study the problem of defending a target such as a stadium or a large gathering place with multiple access paths. In practice, the notion of "layered defense" is commonly used to describe the idea that we have an outer perimeter where we first seek to capture dangerous entities (vehicles, people, cargo), then perhaps a middle perimeter or perimeters where we do the same thing using different methods and perhaps information gathered from the outer perimeter, and then an inner perimeter where we again use different methods and information gathered from earlier perimeter defense. Thus, as vehicles approach a stadium, we might do license plate reading; in a middle layer or layers we use radiation detectors or behavioral detection of patrons after they have parked their cars; then in an inner layer we use metal detection through wanding or walkthrough magnetometers or where we inspect bags or pat-down patrons. We seek to make this idea of layered defense precise in an abstract, simplified way.

We speak abstractly of "sensors" at each layer of defense, but understand that our "sensors" could be physical sensors but also tests of different kinds such as behavioral observation. Our approach is based in an increasing literature that deals with inspection processes using a number of potential tests, for example at ports of entry. In the past few years numerous techniques for sensor optimization of port-of-entry inspection have been explored in the literature [1, 5–9, 12–14, 16, 17]. Several

T. Asamov · E. Yamangil · E. Boros
Rutgers Center for Operations Research, New Brunswick, NJ, USA

P. B. Kantor · F. Roberts (✉)
CCICADA Center, Rutgers University, New Brunswick, NJ, USA
e-mail: paul.kantor@rutgers.edu; froberts@dimacs.rutgers.edu

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_1

authors have reported numerical results that demonstrate significant improvement over straightforward inspection approaches [1, 6, 7, 12, 13, 16]. In line with existing practices, most researchers have assumed that the vast majority of the inspected items and people are perfectly legal and only a very small proportion of the incoming flow is harmful. Under such circumstances, the sensor operating cost (though not the capital cost) is usually only a small fraction of the overall cost of the inspection operation. The bulk of the total cost and time spent is attributed to a thorough inspection procedure that is performed on potential suspicious items and individuals. Such a situation is usually encountered in airport security checkpoints, border crossings, maritime port inspection stations, large sports stadiums, etc.

In mathematical terms, the problem of layered site security is quite different from optimizing a set of sensors searching for illegal cargo at a port-of-entry. Unlike much of the existing inspection optimization work which considers two distinct populations of inspected items, i.e. legal and illegal, in our model we only consider the latter. We work under the assumption that we can incorporate the processing cost of occasional encounters of legal traffic into the overall cost curve for detecting contraband.

2 Mathematical Model

To develop our ideas, we have formulated a model of a perimeter defense of the target with two layers of defense where we have a limited budget for surveillance and we need to decide how much to invest in each layer and where to invest it if there are several locations where we might do inspection in each layer. Defense at the outer layers might be less successful but could provide useful information to selectively refine and adapt strategies at inner layers. Arranging defense in layers so that decisions can be made sequentially might significantly reduce costs and increase chance of success. Monitoring at an outer layer could not only hinder an attacker but could provide information about the current state of threat that could be used to refine and adapt strategies at inner layers. There is a complex tradeoff between maximizing the cost-effectiveness of each layer and overall benefits from devoting some efforts at the outer layer to gathering as much information as possible to maximize effectiveness of the inner layer.

To give a stylized abstract version of what we have in mind, consider Fig. 1, where we show a target in the middle, threats arrive via two inner channels and each is reachable from two outer flows of vehicles, patrons, etc.

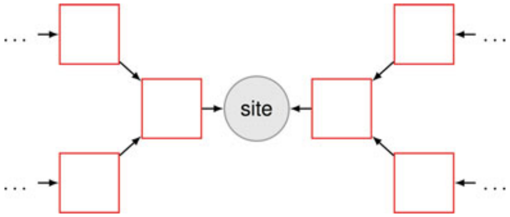


Fig. 1 An abstract model of layered defense showing a target in the middle, threats arriving via two inner channels, and each reachable from two outer flows of vehicles, patrons, etc.

We concentrate in this paper on the problem with two layers of defense, where each security layer has a number of sensors placed on possible paths of incoming illegal flow of vehicles and/or patrons. Inner layers are composed of sensors that are used to detect units that have managed to infiltrate outer layers undetected. Every interior sensor is connected to one or more sensors in the immediately preceding outer layer in the sense that it is responsible for backing up those sensors, i.e., the goal of the interior sensor is to discover traffic which has remained undetected by those outer sensors. In order for an illegal unit to penetrate the system, it would need to remain unnoticed at all layers of inspection. We denote the set of sensors in the internal layer of defense with I, and the set of sensors in the external layer with J. We assume that sensors in the set I share a limited total resource budget X and sensors in the set J share a limited total budget Y. More subtle models allow one to make decisions about how much budget to allocate between the inside and outside layers. Our objective is to develop optimization methods to determine the optimal allocation of resources to security sensors in such a manner that the expected detection rate of incoming threats is maximized. For modeling purposes we employ the following assumptions:

- There exists only one type of violation that we are protecting against.
- The expected number of contraband units on each incoming path is a known parameter. For the outermost perimeter sensor j, we denote the incoming contraband flow with F_j.
- For each sensor $i \in I$, located at the inside security layer, we know the function $D_i^x(x)$, which specifies the detection rate at the sensor for contraband items if the total amount of resources made available to the sensor is x, and similarly for each $j \in J$ we know the detection function $D_j^y(y)$. In this paper we assume that the detection functions are specified as concave increasing piecewise linear functions. Thus, we do not require the detection functions to be differentiable everywhere, which is an important property of our method. We also assume that the resources are normalized to take values between 0 and 1.
- All sensors at a given layer share a limited common resource. For example, an outside perimeter could be supported by a fixed number of infrared motion detectors or license plate readers, while an inside perimeter could consist of walkthrough magnetometer tests or security guards conducting wanding of patrons.

Our goal is to allocate the total outside resources among individual sensors and allocate the total inside resources among individual sensors in order to maximize the detected illegal flow. Thus we arrive at the following mathematical formulation:

$$\begin{aligned}
&\max_{\mathbf{x},\mathbf{y}} \sum_{i\in I} \left\{ \left(\sum_{j\in N(i)} F_j \cdot D_j^y(y_j) \right) + D_i^x(x_i) \left(\sum_{j\in N(i)} F_j(1 - D_j^y(y_j)) \right) \right\} \\
&\text{s.t.} \sum_{i\in I} x_i \le X \\
&\sum_{j\in J} y_j \le Y \\
&x_i \ge 0, \forall i \in I \\
&y_j \ge 0, \forall j \in J
\end{aligned} \tag{1}$$

where $N(i)$ denotes the set of outside sensors adjacent to inside sensor i. Here, the first sum over the outside neighbors j of i gives the flow that is captured at j and the second sum gives the flow that is not captured at j but is captured at i.

Now, let us examine the given objective function. Clearly, it contains mixed nonlinear product terms of detection probabilities. Moreover, since there are no pure quadratic terms, we know that in general the objective function is neither convex, nor concave. We illustrate this in Example 2.1.

Example 2.1 (Indefinite Objective Function) Suppose we have a single exterior sensor j preceding a single interior sensor i, as shown in Fig. 2. In this case, we would need to solve the following problem.

$$\begin{aligned}
&\max_{\mathbf{x_i},\mathbf{y_j}} \left\{ F_j \cdot D_j^y(y_j) + D_i^x(x_i) \cdot (F_j \cdot (1 - D_j^y(y_j))) \right\} \\
&\text{s.t. } 0 \le x_i \le X \\
&\quad 0 \le y_j \le Y
\end{aligned} \tag{2}$$

Let us for a moment consider what would happen if $D_i^x(x) = x$ and $D_j^y(y) = y$ as shown in Fig. 3. In that case, D_i^x and D_j^y are differentiable everywhere. Thus, if we denote the objective function in problem (2) as

$$f^{i,j}(x_i, y_j) = F_j \cdot D_j^y(y_j) + D_i^x(x_i) \cdot (F_j \cdot (1 - D_j^y(y_j)))$$

then we know

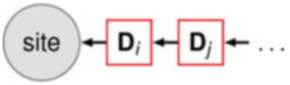


Fig. 2 A model of layered defense showing a target with a single exterior sensor preceding a single interior sensor

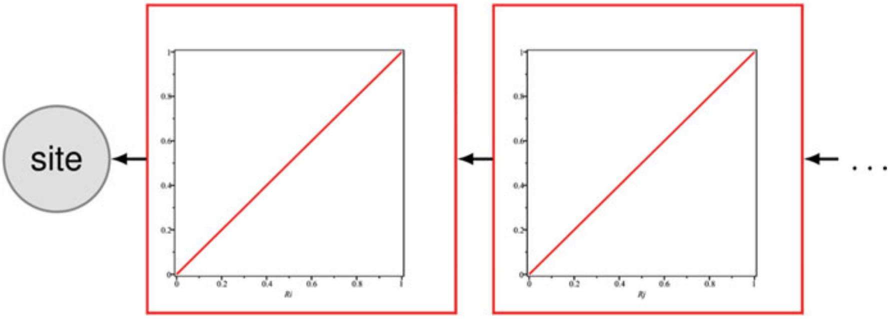


Fig. 3 Linear detection rates at both the exterior and interior sensors of Fig. 2

$$
\begin{aligned}
\nabla f^{i,j}(x_i, y_j) &= \begin{bmatrix} \frac{\partial f^{i,j}(x_i,y_j)}{\partial x_i} \\ \frac{\partial f^{i,j}(x_i,y_j)}{\partial y_j} \end{bmatrix} \\
&= \begin{bmatrix} F_j(1 - D_j^y(y_j)) \\ F_j(1 - D_i^x(x_i)) \end{bmatrix} \\
&= \begin{bmatrix} F_j(1 - y_j) \\ F_j(1 - x_i) \end{bmatrix} \\
&\geq \begin{bmatrix} 0 \\ 0 \end{bmatrix}
\end{aligned} \tag{3}
$$

Therefore the objective function (Fig. 4) is increasing everywhere in the feasible region. Thus, we know that we would get an optimal solution to problem (2) by setting $x_i = X$ and $y_j = Y$. However, upon further inspection we can notice that if we attempted to solve the problem as a convex optimization problem, we would run into difficulties. The Hessian of the objective function has the following form:

$$
\nabla^2 f^{i,j}(x_i, y_j) = \begin{bmatrix} 0 & -F_j \\ -F_j & 0 \end{bmatrix} \tag{4}
$$

And its two eigenvalues are $\lambda_1 = F_j$ and $\lambda_2 = -F_j$. Hence we know that the Hessian matrix associated with the quadratic terms is indefinite.

The indefiniteness of the Hessian presents a major obstacle to solving the problem with standard solvers for quadratic programming. In our study, we tried solving numerous instances using different methods implemented in the MATLAB optimization toolbox. While in some cases we were able to produce consistent output, none of the examined methods were able to overcome the indefiniteness of the Hessian matrix for all possible values of the input parameters. This created the need for the development of an alternative solution method for the problem.

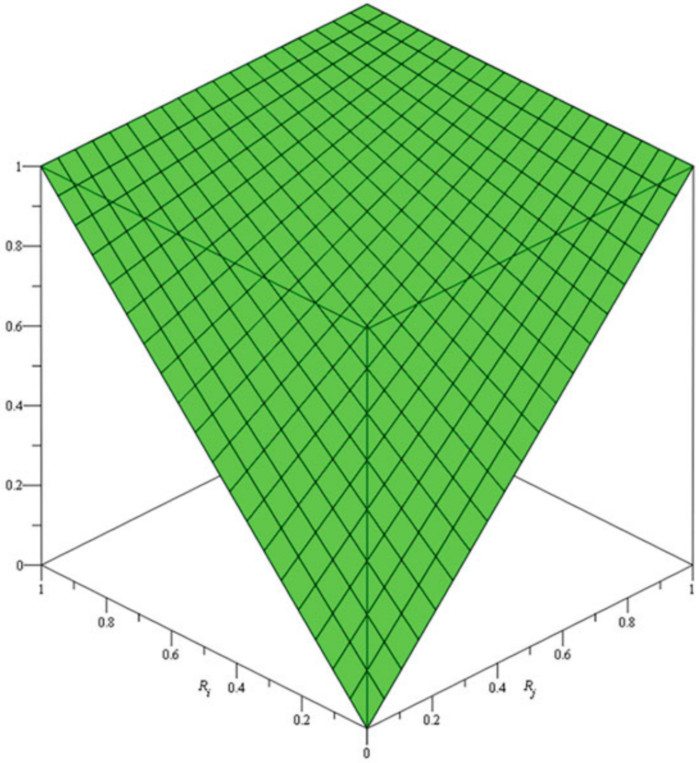


Fig. 4 A plot of the objective function of (2) for the case of $D_i^x(x_i) = x_i$, $D_j^y(y_j) = y_j$ and $F_j = 1$. In this case three of the corners of the feasible region are optimal solutions

3 Exhaustive Search Methods

A standard approach to such problems is a brute force approach that fixes a resource partition mesh and enumerates all possibilities. This exhaustive search approach would be to discretize the resource space for each sensor into a number of subintervals. Then we could examine every possible resource allocation scenario and among all feasible cases select the one that maximizes the objective function value. However, this method would be computationally intractable even for trivial cases. For example, suppose that we have four inside sensors, and each of them is related to exactly two outside sensors. Further, suppose we split the parameter search space of each sensor into one hundred discrete intervals. Then we would need to evaluate the objective function a total of $100^{4+8} = 10^{24}$ times, which is clearly unacceptable unless a very large cluster is used. Moreover, if we considered a slightly larger case of fifteen interior sensors, each supporting a couple of outside perimeter sensors, then the number of cases explodes to $100^{15+30} = 10^{90}$, which exceeds the current estimates for the number of atoms in the universe.

However, it is sufficient to discretize the parameter space for the interior sensors. Then, for each fixed set of values, we can find the optimal configuration of the exterior perimeter by solving a linear programming problem. If we take this approach, then the two above mentioned instances require, respectively, the solution of 10^8 and 10^{30} small linear programming problems. While this is a significant improvement, we are still subjected to the curse of dimensionality as the number of sensors in the interior perimeter increases. Fortunately, we can overcome this challenge.

4 Dynamic Programming Method

To illustrate the idea behind our method, we consider the following basic example. Suppose we would like to solve problem (2) for the case when D_i^x and D_j^y are general piecewise linear functions, as illustrated in Fig. 5.

In that case, the objective function $f^{i,j}(x_i, y_j)$ of problem (2) is still continuous. Further, $f^{i,j}(x_i, y_j)$ is differentiable everywhere except at points corresponding to corner points of the detection functions D_i^x and D_j^y. Moreover, at points where $f^{i,j}(x_i, y_j)$ is differentiable, its gradient has the form

$$\begin{aligned} \nabla f^{i,j}(x_i, y_j) &= \begin{bmatrix} \frac{\partial f(x_i, y_j)}{\partial x_i} \\ \frac{\partial f(x_i, y_j)}{\partial y_j} \end{bmatrix} \\ &= \begin{bmatrix} c_i F_j (1 - D_j^y(y_j)) \\ c_j F_j (1 - D_i^x(x_i)) \end{bmatrix} \\ &\geq \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{aligned} \tag{5}$$

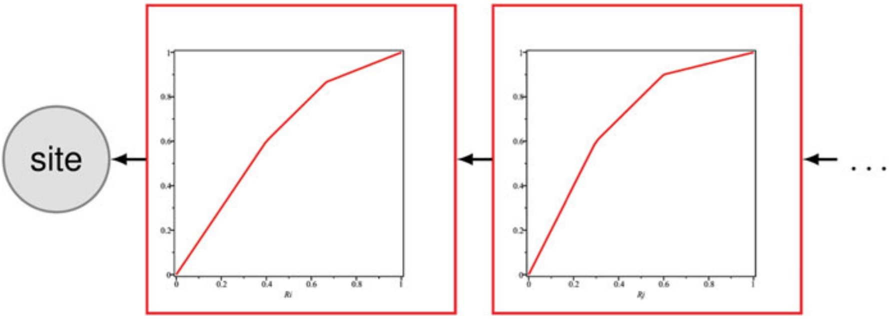


Fig. 5 Piecewise linear detection rates at both the exterior and interior sensors of Fig. 2

for some constants $c_i, c_j \geq 0$. Thus, the optimal solution is again obtained by setting $x_i = X$ and $y_j = Y$.

Our solution method involves four main steps.

Step 1

For a fixed $\epsilon > 0$, we create a partition $\mathcal{Y} = \{0, \epsilon, 2\epsilon, \ldots, Y\}$ of the interval $[0, Y]$, as well as a partition $\mathcal{X} = \{0, \epsilon, 2\epsilon, \ldots, X\}$ of the interval $[0, X]$.

Step 2

For every pair of sensors i, j such that $i \in I$ and $j \in N(i)$, we compute $T^{i,j}(X_i, Y_j)$ for each $(X_i, Y_j) \in \mathcal{X} \times \mathcal{Y}$, where we use $T^{i,j}(X_i, Y_j)$ to denote the maximum amount of detected illegal contraband when inner sensor i uses at most X_i inner resources, and outer sensor j uses at most Y_j outer resources. Formally, we need to compute the optimal value of the following optimization problem:

$$T^{i,j}(X_i, Y_j) = \left\{ \begin{array}{l} \max\limits_{\mathbf{x_i}, \mathbf{y_j}} \left\{ F_j \cdot D_j^y(y_j) + D_i^x(x_i) \cdot (F_j \cdot (1 - D_j^y(y_j))) \right\} \\ \text{s.t. } 0 \leq x_i \leq X_i \\ \qquad 0 \leq y_j \leq Y_j \end{array} \right\} \tag{6}$$

We can solve an instance of problem (6) in $O(1)$ time by setting $x_i = X_i$ and $y_j = Y_j$. Thus, we can compute $T^{i,j}(X_i, Y_j)$ for each $(X_i, Y_j) \in \mathcal{X} \times \mathcal{Y}$ in $O(|\mathcal{X}||\mathcal{Y}|)$. Hence, we can plot the objective function in (2) with arbitrary precision which would ultimately allow us to solve problem (1) with arbitrary precision (see Fig. 6).

Step 3

Now, suppose that instead of a single outside sensor j, we consider two outside sensors j_1 and j_2 that are both backed up by inner sensor i (see Fig. 7). We use $\{j_1, j_2\}$ to denote quantities that refer to the combined system with two outside sensors j_1 and j_2. For example, $T^{i,\{j_1,j_2\}}$ denotes a table of optimal detection values for the combined system of inside sensor i and two outside sensors j_1 and j_2. Further, we also use $X^{i,\{j_1,j_2\}}$ and $Y^{i,\{j_1,j_2\}}$ to denote the amount of inside and outside resources budgeted to the sensor system of inner sensor i, and outer sensors j_1 and j_2. Please recall that in Step 2, we computed the two tables of optimal values T^{i,j_1} and T^{i,j_2} (considering first the problem with only sensors i, j_1 and then the problem with only sensors i, j_2 (see Fig. 8)). Since they both share the same inner sensor, all we need to do in order to find the optimal detection value $T^{i,\{j_1,j_2\}}(X^{i,\{j_1,j_2\}}, Y^{i,\{j_1,j_2\}})$ for the combined system is to determine the optimal way to allocate $Y^{i,\{j_1,j_2\}}$ between sensor j_1 and j_2. Thus we have the problem of optimizing the following formulation:

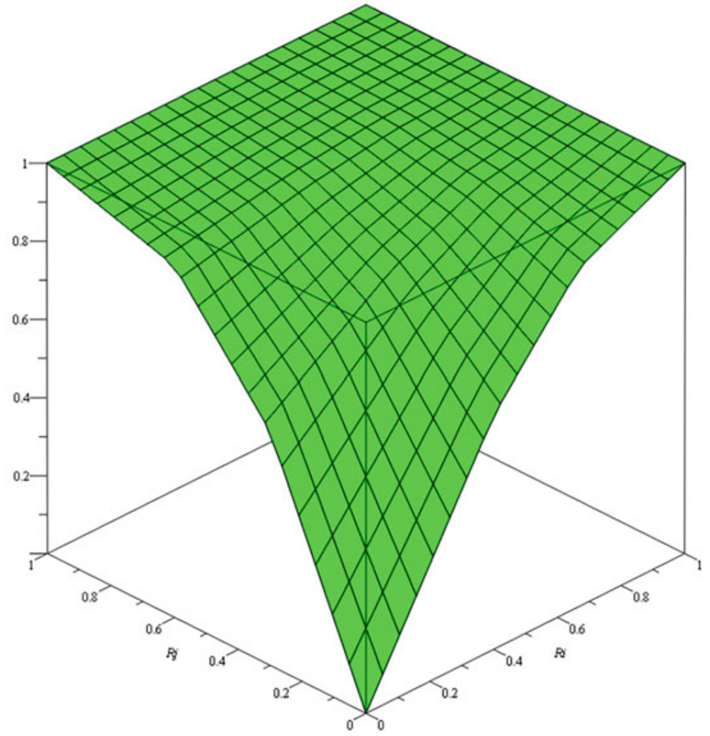


Fig. 6 A plot of the objective function of (2) for piecewise linear functions $D_i^x(x_i)$ and $D_j^y(y_j)$

$$T^{i,\{j_1,j_2\}}\left(X^{i,\{j_1,j_2\}}, Y^{i,\{j_1,j_2\}}\right) =$$

$$= \left\{ \begin{array}{l} \max\limits_{y^{i,j_1}, y^{i,j_2}} T^{i,j_1}\left(X^{i,\{j_1,j_2\}}, y^{i,j_1}\right) + T^{i,j_2}\left(X^{i,\{j_1,j_2\}}, y^{i,j_2}\right) \\ \text{s.t.} \\ y^{i,j_1} + y^{i,j_2} \leq Y^{i,\{j_1,j_2\}} \\ y^{i,j_1} \in \mathcal{Y} \\ y^{i,j_2} \in \mathcal{Y} \end{array} \right\} \tag{7}$$

Notice that even though we consider all different values of $y_j^{i,j_1}, y_j^{i,j_2} \in \mathcal{Y}$, problem (7) can be solved in time linear in the cardinality of $\mathcal{Y}$. This is accomplished by using two index variables initialized at the two ending points of the outside resource partition $\mathcal{Y}$.

If we have three outside sensors j_1, j_2, j_3 corresponding to inside sensor i, then we can find their solution matrix $T^{i,\{j_1,j_2,j_3\}}$ as follows. Once we have computed

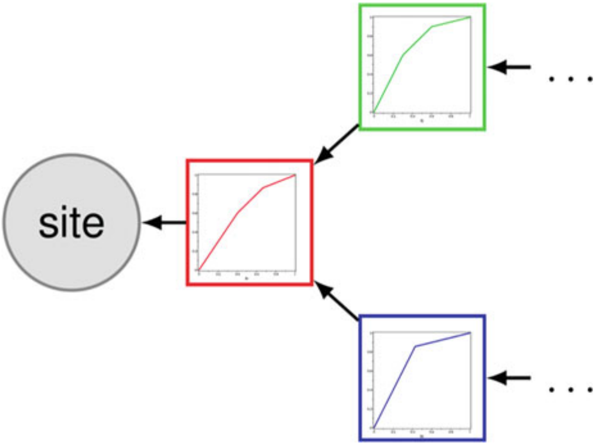


Fig. 7 A network with two outside sensors (green and blue) and one inside sensor backing them both up

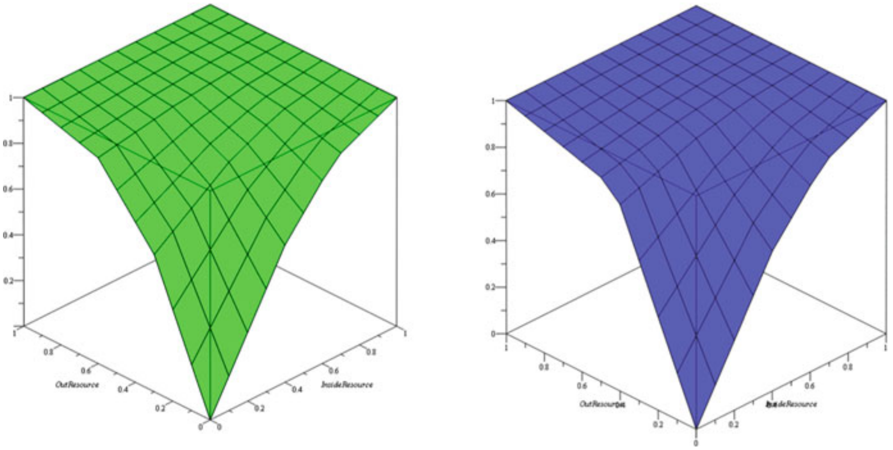


Fig. 8 Finding separate solutions for inner sensor i with each outside sensor j_1 and j_2

the matrix $T^{i,\{j_1,j_2\}}$ we use it as an input to Eq. (7), together with the matrix T^{i,j_3} to generate the solution matrix $T^{i,\{j_1,j_2,j_3\}}$. By recursion, we can solve a problem instance that involves one inner sensor i and any number of outside sensors. We denote with T^i the solution table of optimal values corresponding to inner sensor i together with all of its adjacent outside sensors $j \in N(i)$.

Step 4

Suppose we are given two matrices, T^{i_1} and T^{i_2}, that correspond respectively to two inner sensors i_1 and i_2 with their adjacent outside sensors. In order to determine the optimal detection value $T^{\{i_1,i_2\}}(X^{\{i_1,i_2\}}, Y^{\{i_1,i_2\}})$ for the combined system, we need to find the optimal way to allocate $X_i^{\{i_1,i_2\}}$ between T^{i_1} and T^{i_2}. Thus we consider the problem of optimizing the following,

$$T^{\{i_1,i_2\}}\left(X^{\{i_1,i_2\}}, Y^{\{i_1,i_2\}}\right) = \left\{\begin{array}{l} \max\limits_{x^{i_1},x^{i_2},y^{i_1},y^{i_2}} \quad T^{i_1}\left(x^{i_1}, y^{i_1}\right) + T^{i_2}\left(x^{i_2}, y^{i_2}\right) \\ \text{s.t.} \\ x^{i_1} + x^{i_2} \leq X^{\{i_1,i_2\}} \\ y^{i_1} + y^{i_2} \leq Y^{\{i_1,i_2\}} \\ x^{i_1}, x^{i_2} \in \mathcal{X} \\ y^{i_1}, y^{i_2} \in \mathcal{Y} \end{array}\right\} \tag{8}$$

We point out that problem (8) can be solved in $O(|\mathcal{X}||\mathcal{Y}|)$ time. Finally, once we have computed $T^{\{i_1,i_2\}}$, we can proceed by recursion to solve problems involving an arbitrary number of interior and exterior sensors.

5 Running Time

We consider the running times of all of the four steps.

Step 1: Creating the partitions takes $O(|\mathcal{X}| + |\mathcal{Y}|)$.
Step 2: For every pair i, j such that $i \in I,\ j \in N(i)$, we have to compute a matrix in $O(|\mathcal{X}||\mathcal{Y}|)$ time. Thus, step 2 takes $O(|\mathcal{X}||\mathcal{Y}||I||J|)$.
Step 3: For every $i \in I$ we perform step 3 $|N(i)|$ times, and every time we need to compute $|\mathcal{X}||\mathcal{Y}|$ number of entries, each taking $O(|\mathcal{Y}|)$. Thus the overall complexity of step 3 is $O(|\mathcal{X}||\mathcal{Y}|^2|I||J|)$.
Step 4: We need to execute this step $|I| - 1$ times. Each of $|\mathcal{X}||\mathcal{Y}|$ entries in the resulting matrix takes $O(|\mathcal{X}||\mathcal{Y}|)$ time to compute. Thus, the computational complexity of step 4 is $O((|\mathcal{X}||\mathcal{Y}|)^2|I|)$.

Since the four steps are performed sequentially, we know the overall running time of the dynamic programming method is $O(|\mathcal{X}||\mathcal{Y}|^2|I|(|J| + |\mathcal{X}|))$.

6 Convergence

So far, we have only considered discrete approximations of the optimal detection values. In this section we show that as the partition mesh $\epsilon \to 0$, the values of the discrete approximation tables T converge to the true continuous optimal detection values t.

Since, the discrete approximation is exact for the case of one inside and one outside sensor, we know

$$t^{i,j}(X_i, Y_j) = T^{i,j}(X_i, Y_j),\ \forall (X_i, Y_j) \in \mathcal{X} \times \mathcal{Y}$$

Consider the objective function $f^{i,j}(\cdot,\cdot)$ of the system consisting of inner sensor i and its external neighbor j. Since $f^{i,j}$ is continuous, as well as quadratic everywhere except for a set of measure zero, we know that $f^{i,j}$ is Lipschitz continuous and we denote its Lipschitz constant with $L^{i,j}$. If we choose $L \in \mathbb{R}$ such that

$$L = \max_{\substack{i \in I \\ j \in N(i)}} L^{i,j}$$

then L is a Lipschitz constant for all functions $f^{i,j}$.

Suppose $i \in I$ and $j_1, j_2 \in N(i)$, $j_1 \neq j_2$. We use $\xi^{i,\{j_1,j_2\}}$ to denote the error between the true optimal detection value $t^{i,\{j_1,j_2\}}$, and the discrete approximation $T^{i,\{j_1,j_2\}}$. We would use lower case x^{i,j_1} and x^{i,j_2} to denote the optimal way to split up the inside resources $X^{i,\{j_1,j_2\}}$ between t^{i,j_1} and t^{i,j_2}, while y^{i,j_1} and y^{i,j_2} denote the optimal way to split up the outside resources $Y^{i,\{j_1,j_2\}}$ between t^{i,j_1} and t^{i,j_2}.

On the other hand, we would use X^{i,j_1} and X^{i,j_2} to denote the optimal way to split up the inside resources $X^{i,\{j_1,j_2\}}$ between T^{i,j_1} and T^{i,j_2}, while Y^{i,j_1} and Y^{i,j_2} denote the optimal way to split up the outside resources $Y^{i,\{j_1,j_2\}}$ between T^{i,j_1} and T^{i,j_2}. Before we proceed, we need to introduce the following notation. We use $\lfloor x \rfloor$ to denote the point in $\mathcal{X}$ that is closest to x from below, and we use $\lfloor y \rfloor$ to denote the point in $\mathcal{Y}$ that is closest to y from below. Similarly, we use $\lceil x \rceil$ to denote the point in $\mathcal{X}$ that is closest to x from above, and we use $\lceil y \rceil$ to denote the point in $\mathcal{Y}$ that is closest to y from above. Then the discrete approximation error can be written as,

$$\begin{aligned}
&\xi^{i,\{j_1,j_2\}}\left(X^{i,\{j_1,j_2\}}, Y^{i,\{j_1,j_2\}}\right) \\
&= t^{i,\{j_1,j_2\}}\left(X^{i,\{j_1,j_2\}}, Y^{i,\{j_1,j_2\}}\right) - T^{i,\{j_1,j_2\}}\left(X^{i,\{j_1,j_2\}}, Y^{i,\{j_1,j_2\}}\right) \\
&= t^{i,j_1}\left(x^{i,j_1}, y^{i,j_1}\right) + t^{i,j_2}\left(x^{i,j_2}, y^{i,j_2}\right) \\
&\quad -T^{i,j_1}\left(X^{i,j_1}, Y^{i,j_1}\right) - T^{i,j_2}\left(X^{i,j_2}, Y^{i,j_2}\right) \\
&\leq t^{i,j_1}\left(x^{i,j_1}, y^{i,j_1}\right) + t^{i,j_2}\left(x^{i,j_2}, y^{i,j_2}\right) \\
&\quad -T^{i,j_1}\left(\lfloor x^{i,j_1} \rfloor, \lfloor y^{i,j_1} \rfloor\right) - T^{i,j_2}\left(\lfloor x^{i,j_2} \rfloor, \lfloor y^{i,j_2} \rfloor\right) \\
&= t^{i,j_1}\left(x^{i,j_1}, y^{i,j_1}\right) + t^{i,j_2}\left(x^{i,j_2}, y^{i,j_2}\right) \\
&\quad -t^{i,j_1}\left(\lfloor x^{i,j_1} \rfloor, \lfloor y^{i,j_1} \rfloor\right) - t^{i,j_2}\left(\lfloor x^{i,j_2} \rfloor, \lfloor y^{i,j_2} \rfloor\right) \\
&= \left\{t^{i,j_1}\left(x^{i,j_1}, y^{i,j_1}\right) - t^{i,j_1}\left(\lfloor x^{i,j_1} \rfloor, \lfloor y^{i,j_1} \rfloor\right)\right\} \\
&\quad + \left\{t^{i,j_2}\left(x^{i,j_2}, y^{i,j_2}\right) - t^{i,j_2}\left(\lfloor x^{i,j_2} \rfloor, \lfloor y^{i,j_2} \rfloor\right)\right\} \qquad (9)
\end{aligned}$$

Since we have $\left\{t^{i,j_k}\left(x^{i,j_k}, y^{i,j_k}\right) - t^{i,j_k}\left(\lfloor x^{i,j_k}\rfloor, \lfloor y^{i,j_k}\rfloor\right)\right\} \leq \sqrt{2}\epsilon L$ for both $k = 1, 2$ the bound

$$\xi^{i,\{j_1,j_2\}}\left(X^{i,\{j_1,j_2\}}, Y^{i,\{j_1,j_2\}}\right) \leq 2\sqrt{2}\epsilon L \tag{10}$$

follows. Now, we can also bound the error in the case of three outside sensors:

$$\begin{aligned}
&\xi^{i,\{j_1,j_2,j_3\}}\left(X^{i,\{j_1,j_2,j_3\}}, Y^{i,\{j_1,j_2,j_3\}}\right) = \\
&\quad = t^{i,\{j_1,j_2,j_3\}}\left(X^{i,\{j_1,j_2,j_3\}}, Y^{i,\{j_1,j_2,j_3\}}\right) \\
&\qquad\quad - T^{i,\{j_1,j_2,j_3\}}\left(X^{i,\{j_1,j_2,j_3\}}, Y^{i,\{j_1,j_2,j_3\}}\right) \\
&\quad = t^{i,\{j_1,j_2\}}\left(x^{i,\{j_1,j_2\}}, x^{i,\{j_1,j_2\}}\right) + t^{i,j_3}\left(x^{i,j_3}, y^{i,j_3}\right) \\
&\qquad\quad - T^{i,\{j_1,j_2\}}\left(X^{i,\{j_1,j_2\}}, Y^{i,\{j_1,j_2\}}\right) \\
&\qquad\quad - T^{i,j_3}\left(X^{i,j_3}, Y^{i,j_3}\right) \\
&\quad \leq t^{i,\{j_1,j_2\}}\left(x^{i,\{j_1,j_2\}}, y^{i,\{j_1,j_2\}}\right) + t^{i,j_3}\left(x^{i,j_3}, y^{i,j_3}\right) \\
&\qquad\quad - T^{i,\{j_1,j_2\}}\left(\lfloor x^{i,\{j_1,j_2\}}\rfloor, \lfloor y^{i,\{j_1,j_2\}}\rfloor\right) \\
&\qquad\quad - T^{i,j_3}\left(\lfloor x^{i,j_3}\rfloor, \lfloor y^{i,j_3}\rfloor\right) \\
&\quad \leq t^{i,\{j_1,j_2\}}\left(x^{i,\{j_1,j_2\}}, y^{i,\{j_1,j_2\}}\right) + t^{i,j_3}\left(x^{i,j_3}, y^{i,j_3}\right) \\
&\qquad\quad - t^{i,\{j_1,j_2\}}\left(\lfloor x^{i,\{j_1,j_2\}}\rfloor, \lfloor y^{i,\{j_1,j_2\}}\rfloor\right) \\
&\qquad\quad - t^{i,j_3}\left(\lfloor x^{i,j_3}\rfloor, \lfloor y^{i,j_3}\rfloor\right) + 2\sqrt{2}\epsilon L \\
&\quad = \left\{t^{i,j_1}\left(x^{i,j_1}, y^{i,j_1}\right) - t^{i,j_1}\left(\lfloor x^{i,j_1}\rfloor, \lfloor y^{i,j_1}\rfloor\right)\right\} \\
&\qquad\quad + \left\{t^{i,j_2}\left(x^{i,j_2}, y^{i,j_2}\right) - t^{i,j_2}\left(\lfloor x^{i,j_2}\rfloor, \lfloor y^{i,j_2}\rfloor\right)\right\} \\
&\qquad\quad + 2\sqrt{2}\epsilon L \\
&\quad \leq \sqrt{2}\epsilon L + \sqrt{2}\epsilon L + 2\sqrt{2}\epsilon L \\
&\quad = 4\sqrt{2}\epsilon L
\end{aligned} \tag{11}$$

Proceeding by induction, we know that

$$\xi^i \leq 2\sqrt{2}(|J|-1)\epsilon L, \forall i \in I$$

since an inside sensor can have at most $|J|$ adjacent outside sensors.

Now, suppose that $i_1, i_2 \in I, i_1 \neq i_2$. Then,

$$\begin{aligned}
&\xi^{\{i_1,i_2\}}\left(X^{\{i_1,i_2\}}, Y^{\{i_1,i_2\}}\right)\\
&\quad = t^{\{i_1,i_2\}}\left(X^{\{i_1,i_2\}}, Y^{\{i_1,i_2\}}\right) - T^{\{i_1,i_2\}}\left(X^{\{i_1,i_2\}}, Y^{\{i_1,i_2\}}\right)\\
&\quad = t^{i_1}\left(x^{i_1}, y^{i_1}\right) + t^{i_2}\left(x^{i_2}, y^{i_2}\right) - T^{i_1}\left(X^{i_1}, Y^{i_1}\right) - T^{i_2}\left(X^{i_2}, Y^{i_2}\right)\\
&\quad \leq t^{i_1}\left(x^{i_1}, y^{i_1}\right) + t^{i_2}\left(x^{i_2}, y^{i_2}\right) - T^{i_1}\left(\lfloor x^{i_1}\rfloor, \lfloor y^{i_1}\rfloor\right) - T^{i_2}\left(\lfloor x^{i_2}\rfloor, \lfloor y^{i_2}\rfloor\right)\\
&\quad \leq t^{i_1}\left(\lceil x^{i_1}\rceil, \lceil y^{i_1}\rceil\right) + t^{i_2}\left(\lceil x^{i_2}\rceil, \lceil y^{i_2}\rceil\right)\\
&\qquad\qquad - T^{i_1}\left(\lfloor x^{i_1}\rfloor, \lfloor y^{i_1}\rfloor\right) - T^{i_2}\left(\lfloor x^{i_2}\rfloor, \lfloor y^{i_2}\rfloor\right)\\
&\quad = \left\{t^{i_1}\left(\lfloor x^{i_1}\rfloor, \lfloor y^{i_1}\rfloor\right) - T^{i_1}\left(\lfloor x^{i_1}\rfloor, \lfloor y^{i_1}\rfloor\right)\right\}\\
&\qquad\qquad + \left\{t^{i_2}\left(\lfloor x^{i_2}\rfloor, \lfloor y^{i_2}\rfloor\right) - T^{i_2}\left(\lfloor x^{i_2}\rfloor, \lfloor y^{i_2}\rfloor\right)\right\}\\
&\qquad\qquad + \left\{t^{i_1}\left(\lceil x^{i_1}\rceil, \lceil y^{i_1}\rceil\right) - t^{i_1}\left(\lfloor x^{i_1}\rfloor, \lfloor y^{i_1}\rfloor\right)\right\}\\
&\qquad\qquad + \left\{t^{i_2}\left(\lceil x^{i_2}\rceil, \lceil y^{i_2}\rceil\right) - t^{i_2}\left(\lfloor x^{i_2}\rfloor, \lfloor y^{i_2}\rfloor\right)\right\}\\
&\quad \leq 2\sqrt{2}\left(|J|-1\right)\epsilon L + 2\sqrt{2}\left(|J|-1\right)\epsilon L + 2\sqrt{2}\epsilon L\\
&\quad \leq 4\sqrt{2}\left(|J|\right)\epsilon L
\end{aligned} \tag{12}$$

We can also bound the error in the case of three inside sensors and all of their adjacent outside sensors:

$$\begin{aligned}
&\xi^{\{i_1,i_2,i_3\}}(X^{\{i_1,i_2,i_3\}}, Y^{\{i_1,i_2,i_3\}})\\
&\quad = t^{\{i_1,i_2,i_3\}}\left(X^{\{i_1,i_2,i_3\}}, Y^{\{i_1,i_2,i_3\}}\right) - T^{\{i_1,i_2,i_3\}}\left(X^{\{i_1,i_2,i_3\}}, Y^{\{i_1,i_2,i_3\}}\right)\\
&\quad = t^{\{i_1,i_2\}}\left(x^{\{i_1,i_2\}}, y^{\{i_1,i_2\}}\right) + t^{i_3}\left(x^{i_3}, y^{i_3}\right)\\
&\qquad\qquad -T^{\{i_1,i_2\}}\left(X^{\{i_1,i_2\}}, Y^{\{i_1,i_2\}}\right) - T^{i_3}\left(X^{i_3}, Y^{i_3}\right)\\
&\quad \leq t^{\{i_1,i_2\}}\left(x^{\{i_1,i_2\}}, y^{\{i_1,i_2\}}\right) + t^{i_3}\left(x^{i_3}, y^{i_3}\right)\\
&\qquad\qquad -T^{\{i_1,i_2\}}\left(\lfloor x^{\{i_1,i_2\}}\rfloor, \lfloor y^{\{i_1,i_2\}}\rfloor\right) - T^{i_3}\left(\lfloor x^{i_3}\rfloor, \lfloor y^{i_3}\rfloor\right)
\end{aligned}$$

$$
\begin{aligned}
&\leq t^{\{i_1,i_2\}}\left(\lceil x^{\{i_1,i_2\}}\rceil, \lceil y^{\{i_1,i_2\}}\rceil\right) + t^{i_3}\left(\lceil x^{i_3}\rceil, \lceil y^{i_3}\rceil\right) \\
&\qquad\qquad -T^{\{i_1,i_2\}}\left(\lfloor x^{\{i_1,i_2\}}\rfloor, \lfloor y^{\{i_1,i_2\}}\rfloor\right) - T^{i_3}\left(\lfloor x^{i_3}\rfloor, \lfloor y^{i_3}\rfloor\right) \\
&= \left\{t^{\{i_1,i_2\}}\left(\lfloor x^{\{i_1,i_2\}}\rfloor, \lfloor y^{\{i_1,i_2\}}\rfloor\right) - T^{\{i_1,i_2\}}\left(\lfloor x^{\{i_1,i_2\}}\rfloor, \lfloor y^{\{i_1,i_2\}}\rfloor\right)\right\} \\
&\quad + \left\{t^{i_3}\left(\lfloor x^{i_3}\rfloor, \lfloor y^{i_3}\rfloor\right) - T^{i_3}(\lfloor x^{i_3}\rfloor, \lfloor y^{i_3}\rfloor)\right\} \\
&\quad + \left\{t^{\{i_1,i_2\}}\left(\lceil x^{\{i_1,i_2\}}\rceil, \lceil y^{\{i_1,i_2\}}\rceil\right) - t^{\{i_1,i_2\}}\left(\lfloor x^{\{i_1,i_2\}}\rfloor, \lfloor y^{\{i_1,i_2\}}\rfloor\right)\right\} \\
&\quad + \left\{t^{i_3}\left(\lceil x^{i_3}\rceil, \lceil y^{i_3}\rceil\right) - t^{i_3}(\lfloor x^{i_3}\rfloor, \lfloor y^{i_3}\rfloor)\right\} \\
&\leq 4\sqrt{2}|J|\epsilon L + 2\sqrt{2}(|J|-1)\epsilon L + 2\sqrt{2}\epsilon L \\
&\leq 6\sqrt{2}(|J|)\epsilon L
\end{aligned}
\tag{13}
$$

Proceeding by induction, we know that ξ^I the error of the discrete approximation for the entire set of internal sensors I and all of their adjacent outside sensors is bounded by

$$\xi^I \leq 2\sqrt{2}|I||J|\epsilon L$$

Therefore,

$$
\begin{aligned}
\lim_{\epsilon\to 0}\xi^I &= t^I(X^I, Y^I) - T^I(X^I, Y^I) \\
&\leq \lim_{\epsilon\to 0} 2\sqrt{2}|I||J|\epsilon \\
&= 0
\end{aligned}
\tag{14}
$$

Hence, as $\epsilon \to 0$ the discrete approximation $T^I(X^I, Y^I)$ converges to the true continuous optimal detection value $t^I(X^I, Y^I)$.

7 The Case of an Adaptive Adversary

So far, our model assumed a fixed flow of dangerous material on each pathway, and we have presented a method that would allow law enforcement officials to use current information on attacker behavior to maximize the amount of captured illegal or dangerous contraband. However, we can think of attackers as intelligent adversaries who would adjust their strategy once they observe the changes in site security. Therefore, the goal of a defensive strategy could be to make sure that no path leading into the site has a violation detection rate that is unreasonably low. For example, suppose we have an adaptive adversary who recognizes how much of a resource we use for sensors on each node and then chooses the path that minimizes the probability of detection. To defend against such an adversary we might seek to assign sensor resources so as to maximize the minimum detection rate on any path. Hence we face the following optimization challenge:

$$\begin{aligned}
&\max_{\mathbf{x},\mathbf{y}} \min_{\substack{i\in I\\ j\in N(i)}} \left\{D_j^y(y_j) + D_i^x(x_i)(1 - D_j^y(y_j))\right\}\\
&\text{s.t.} \sum_{i\in I} x_i \le X\\
&\quad\sum_{j\in J} y_j \le Y\\
&\quad x_i \ge 0, \forall i \in I\\
&\quad y_j \ge 0, \forall j \in J
\end{aligned} \tag{15}$$

In order to solve this problem we can use a similar approach to the one discussed in the previous section.

Steps 1 and 2
These are identical to their counterparts described in Sect. 4, with $F_j = 1$ for every outside sensor j.

Step 3
Once again, we denote by $T^{i,j}$ the resulting table of values generated at Step 2. More specifically, we denote with $T^{i,j}(X_i, Y_j)$ the optimal detection value that can be achieved by investing $(X_i, Y_j) \in \mathcal{X} \times \mathcal{Y}$ resources of respectively, inner and outer resources. Then we consider the case of two outside sensors j_1, j_2 that are backed up by inner sensor i. We can merge T^{i,j_1} and T^{i,j_2} into a single solution according to:

$$T^{i,\{j_1,j_2\}}\left(X^{i,\{j_1,j_2\}}, Y^{i,\{j_1,j_2\}}\right) =$$
$$= \left\{\begin{array}{l}
\max\limits_{y^{i,j_1},y^{i,j_2}} \min\left\{T^{i,j_1}\left(X^{i,\{j_1,j_2\}}, y^{i,j_1}\right), T^{i,j_2}\left(X^{i,\{j_1,j_2\}}, y^{i,j_2}\right)\right\}\\
\text{s.t.}\\
y^{i,j_1} + y^{i,j_2} \le Y^{i,\{j_1,j_2\}}\\
y^{i,j_1} \in \mathcal{Y}\\
y^{i,j_2} \in \mathcal{Y}
\end{array}\right\} \tag{16}$$

where $T^{i,\{j_1,j_2\}}\left(X^{i,\{j_1,j_2\}}, Y^{i,\{j_1,j_2\}}\right)$ is the optimal detection value for the combined system.

Again, if we proceed by induction, we can generate an optimal value table for a problem instance that involves one inner sensor i, and an arbitrary number of outside sensors. We denote such a table by T^i.

Step 4
Consider two matrices T^{i_1} and T^{i_2} that correspond to respectively inner sensors i_1 and i_2 with all of their adjacent outside sensors. We could again merge the two solutions into a single global solution according to the following rule:

$$T^{\{i_1,i_2\}}\big(X^{\{i_1,i_2\}}, Y^{\{i_1,i_2\}}\big) = \left\{ \begin{array}{l} \max\limits_{x^{i_1},x^{i_2},y^{i_1},y^{i_2}} \min\left\{T^{i_1}\big(x^{i_1}, y^{i_1}\big), T^{i_2}\big(x^{i_2}, y^{i_2}\big)\right\} \\ \text{s.t.} \\ x^{i_1} + x^{i_2} \le X^{\{i_1,i_2\}} \\ y^{i_1} + y^{i_2} \le Y^{\{i_1,i_2\}} \\ x^{i_1}, x^{i_2} \in \mathcal{X} \\ y^{i_1}, y^{i_2} \in \mathcal{Y} \end{array} \right\} \tag{17}$$

where $T^{\{i_1,i_2\}}\big(X^{\{i_1,i_2\}}, Y^{\{i_1,i_2\}}\big)$ denotes the optimal value for the combined system that employs internal sensors i_1 and i_2, and all of their outside neighbors.

Once again, if we have a third branch consisting of inside sensor i_3 and its outside neighbors, then we can use $T^{\{i_1,i_2\}}$ and T^{i_3} as inputs to Eq. (17) and find the table of optimal values $T^{\{i_1,i_2,i_3\}}$ for the combined system consisting of inside sensors i_1, i_2, i_3, and all of their adjacent outside sensors. Proceeding by induction, we know that even if we have an adaptive adversary, we can solve problems involving an arbitrary number of interior and exterior sensors, as well as sensor detection curves specified by concave increasing piecewise linear functions.

8 Computational Results

In this section we present computational results for the methods developed above. The experiments were performed on an AMD Phenom X4 9550 workstation with 6GB of DDR2 RAM. We consider two different system configurations, and for each of them we provide plots of the objective function value for both the original and adaptive adversary models.

Example 8.1 In this example we consider an inner layer consisting of four sensors, one with three adjacent outside sensors (indices 1, 2, 3), a second with two adjacent outside sensors (indices 4, 5), a third with two adjacent outside sensors (indices 6, 7), and a fourth with two adjacent outside sensors (indices 8, 9). For each inside sensor $i \in I$, we specify $D_i^x(x) = \min\{0.2x, 0.4 + 0.1x\}$. Further, for the first outside

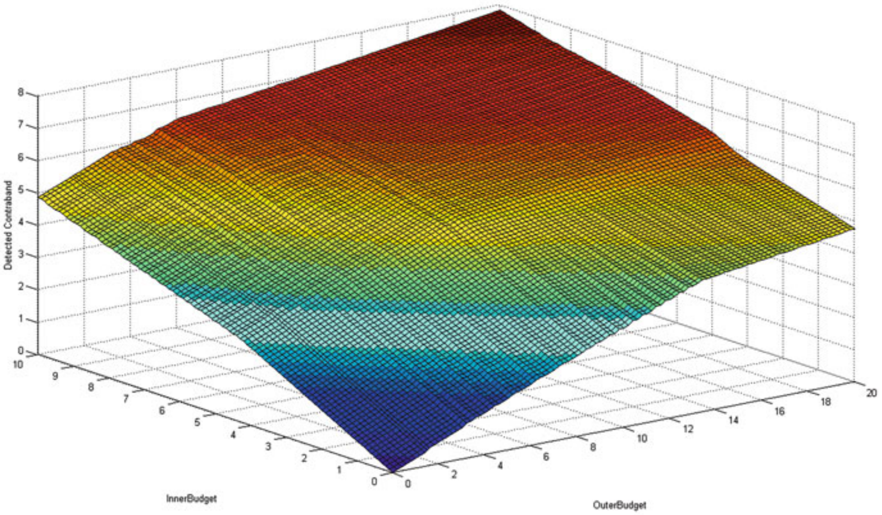


Fig. 9 Solution maximizing the expected amount of captured contraband for a range of interior and exterior budgets for Example 8.1

sensor we use $D^y_{j_1}(y) = \min\{0.3y, 0.3+0.1y, 0.5+0.05y\}$, and for outside sensors of index $j = 2, 3, \ldots, 9$, we use $D^y_j(y) = \min\{0.3y, 0.3 + 0.1y\}$. In addition, all outside sensors have exactly 1 unit of incoming flow. Figure 9 gives the solution maximizing the expected amount of captured contraband for a range of interior and exterior budgets, i.e., the solution to the first problem. The solution matrix includes 10,302 distinct points and the computation took 117 seconds.

We can also calculate the adaptive adversary solution maximizing the minimum probability of capturing contraband along all paths for a range of interior and exterior budgets. The solution shown in Fig. 10 includes 40,401 distinct points and the computation took 3102 seconds (52 minutes).

Example 8.2 In this example, we modify Example 8.1 so that all the outside sensors have exactly 1 unit of incoming flow except for outside sensors 1 and 9 which have 10 units of incoming flow. Figure 11 shows the optimal objective values of the maximized amount of captured contraband for a range of interior and exterior budgets. The solution table includes 10,302 distinct points, and the computation took 119 seconds.

In this example we only changed the flow values of Example 8.1. For this reason, we do not need to compute a new adaptive adversary solution, as it would be identical to the one for Example 8.1. Naturally, this illustrates the robustness of the adaptive adversary formulation compared to its original counterpart.

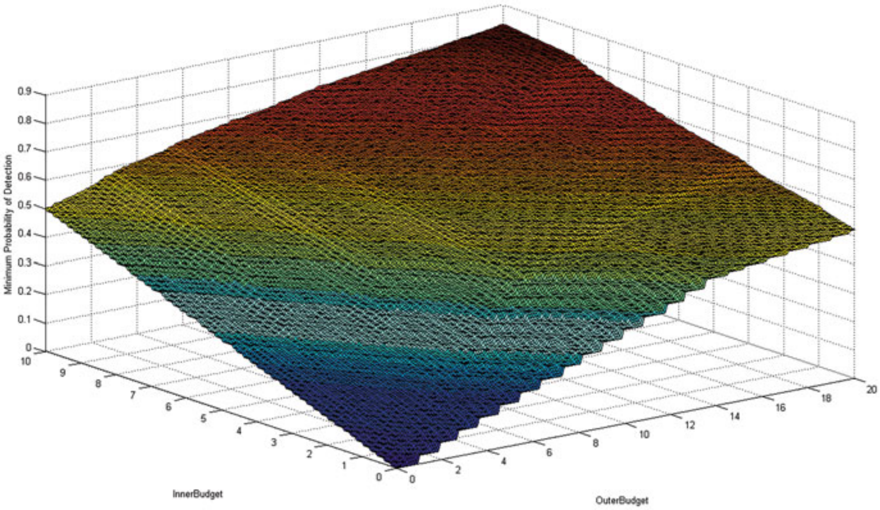


Fig. 10 Adaptive adversary solution maximizing the minimum probability of capturing contraband along all paths for a range of interior and exterior budgets for Example 8.1

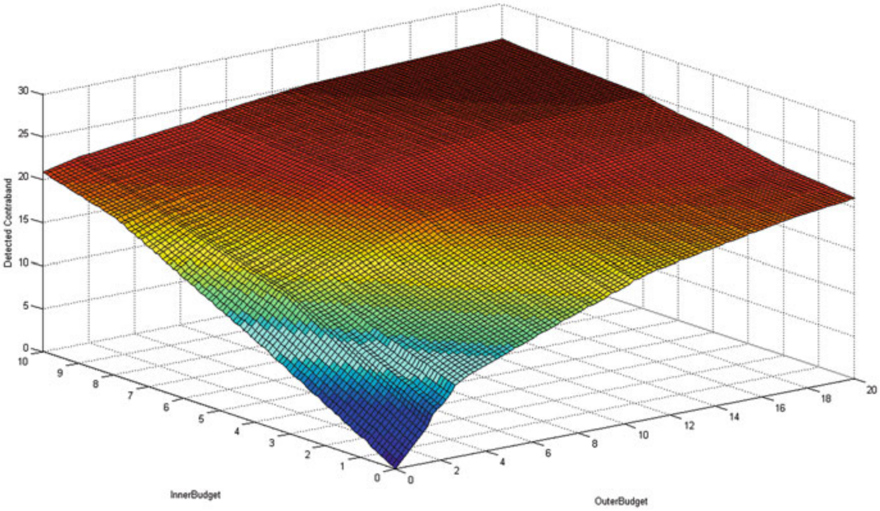


Fig. 11 Solution maximizing the expected amount of captured contraband for a range of interior and exterior budgets for Example 8.2

9 Closing Remarks

We have considered the problem of determining the optimal resource allocation for layered security. A computational method for the maximization of captured contraband and an adaptive adversary approach for the maximization of the worst case probability of detection have been developed. Both methods are computationally tractable and can be applied to non-trivial practical problems.

We have a great deal more that we can do in the future. One thing is to consider both legal and illegal flow, which we also refer to as respectively good and bad units. Hence, in addition to detecting bad units we could consider false positive decisions for each sensor and adopt a risk-averse optimization approach [3]. Another possible direction would be attempting to write the problem as a large game and use approximation methods similar to the ones developed by Grigoriadis and Khachian [10]. Alternatively, we could look into interdiction on planar graphs methods similar to the ones developed by Zenklusen [18, 19].

Still another approach is to follow the applications of Stackelberg games that have been used in pioneering defensive approaches at the nation's airports, ports, and in applications by the Federal Air Marshals Service, US Coast Guard, etc. (see [11, 15]). In a Stackelberg game between an attacker and a defender, the defender (security) acts first. The attacker can observe the defender's strategy and choose the most beneficial point of attack. The challenge is to introduce some randomness in the defender's strategy to increase the uncertainty on the part of the attacker. Bayesian Stackelberg games do exactly that. Layered defense makes this into a new kind of Stackelberg game to analyze, one with two rounds, one involving the outer layer and one involving the inner layer based on results at the outer layer. We can look both at nonrandomized and randomized strategies for the defender.

There are many other directions in which this work could go. Even with our current model, we have not yet developed practical methods to handle more than two layers of defense. There are also many variations on our model that could be quite interesting. For example, we could consider a fixed resource limit that the defender could allocate between inner and outer layers. Then, we could allow adaptive redistribution of resources across layers and across time (see [2, 4]).

Disclaimer The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.

Acknowledgments This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Numbers 22STESE000010101 from DHS Office of University Programs via Northeastern University, DHS 2009- ST-061-CCI002 from DHS Office of University Programs to Rutgers University, and HSHQDC-13-X-00069 from DNDO. PBK acknowledges support from the NSF Grants Numbered 1247696 and 2041428, and the Department of Industrial and Systems Engineering, University of Wisconsin, Madison.

References

1. Anand, S., Madigan, D., Mammone, R., Pathak, S., Roberts, F.: Experimental analysis of sequential decision making algorithms for port of entry inspection procedures. In: Intelligence and Security Informatics, pp. 319–330. Springer (2006)

2. Asamov, T., Powell, W.B.: Regularized decomposition of high-dimensional multistage stochastic programs with Markov uncertainty. SIAM J. Optim. **28**(1), 575–595 (2018)
3. Asamov, T., Ruszczyński, A.: Time-consistent approximations of risk-averse multistage stochastic optimization problems. Math. Program. **153**(2), 459–493 (2015)
4. Asamov, T., Salas, D.F., Powell, W.B.: SDDP vs. ADP: the effect of dimensionality in multistage stochastic optimization for grid level energy. storage. Preprint. arXiv:1605.01521 (2016)
5. Boros, E., Elsayed, E.A., Kantor, P.D., Roberts, F., Xie, M.: Optimization problems for port-of-entry detection systems. In: Intelligence and Security Informatics, pp. 319–335. Springer (2008)
6. Boros, E., Fedzhora, L., Kantor, P.B., Saeger, K., Stroud, P.: A large-scale linear programming model for finding optimal container inspection strategies. Naval Res. Logist. **56**(5), 404–420 (2009)
7. Boros, E., Goldberg, N., Kantor, P.B., Word, J.: Optimal sequential inspection policies. Ann. Oper. Res. **187**(1), 89–119 (2011)
8. Carpenter, T., Cheng, J., Roberts, F., Xie, M.: Sensor management problems of nuclear detection. In: Safety and Risk Modeling and Its Applications, pp. 299–323. Springer (2011)
9. Elsayed, E.A., Young, C.M., Xie, M., Zhang, H., Zhu, Y.: Port-of-entry inspection: sensor deployment policy optimization. IEEE Trans. Autom. Sci. Eng. **6**(2), 265–276 (2009)
10. Grigoriadis, M.D., Khachiyan, L.G., Porkolab, L., Villavicencio, J.: Approximate max-min resource sharing for structured concave optimization. SIAM J. Optim. **11**(4), 1081–1091 (2001)
11. Jain, M., Tsai, J., Pita, J., Kiekintveld, C., Rathi, S., Tambe, M., Ordóñez, F.: Software assistants for randomized patrol planning for the LAX airport police and the federal air marshal service. Interfaces **40**(4), 267–290 (2010)
12. Madigan, D., Mittal, S., Roberts, F.: Sequential decision making algorithms for port of entry inspection: Overcoming computational challenges. In: Intelligence and Security Informatics, 2007 IEEE, pp. 1–7. IEEE (2007)
13. Madigan, D., Mittal, S., Roberts, F.: Efficient sequential decision-making algorithms for container inspection operations. Naval Res. Logist. (NRL) **58**(7), 637–654 (2011)
14. Pita, J., Jain, M., Marecki, J., Ordóñez, F., Portway, C., Tambe, M., Western, C., Paruchuri, P., Kraus, S.: Deployed ARMOR protection: the application of a game theoretic model for security at the Los Angeles International Airport. In: Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems: Industrial Track, pp. 125–132. International Foundation for Autonomous Agents and Multiagent Systems (2008)
15. Sandler, T., Arce, D.G.: Terrorism: a game-theoretic approach. In: Handbook of Defense Economics, vol. 2, pp. 775–813 (2007)
16. Stroud, P.D., Saeger, K.J.: Enumeration of increasing boolean expressions and alternative digraph implementations for diagnostic applications. In: Proceedings, vol. 4, pp. 328–333 (2003)
17. Young, C.M., Li, M., Zhu, Y., Xie, M., Elsayed, E.A., Asamov, T.: Multiobjective optimization of a port-of-entry inspection policy. IEEE Trans. Autom. Sci. Eng. **7**(2), 392–400 (2010)
18. Zenklusen, R.: Extensions to network flow interdiction on planar graphs. Preprint. arXiv:0801.1737 (2008)
19. Zenklusen, R.: Network flow interdiction on planar graphs. Discrete Appl. Math. **158**(13), 1441–1455 (2010)

Developing and Running a Set of Competitive College/University Blended Courses

Alexander S. Belenky

1 Introduction

Distance learning is one of the rapidly growing directions in higher education, and, according to numerous reports, only in the U.S., it currently embraces millions of students [1]. Higher education specialists continue to discuss various philosophical and pedagogical problems of distance learning, including those related to the quality of education that it may provide (compared to a traditional off-line higher education) [2]. At the same time, the distance learning economics becomes one of the areas which starts to concern administrations of all the colleges and universities at which the courses available via the Internet are or are planned to be in use [3].

One of the topics that is actively discussed in scientific publications in the framework of the economics of distance learning deals with specifics of the so-called blended courses [4–7]. A group of such courses is formed by those incorporating recorded fragments from particular courses available on the Internet into lectures and seminars that are taught by college/university professors in classrooms offline.

The available data on studies analyzing the effectiveness of various forms of distance learning in general suggest that from the viewpoint of the quality of education, blended courses can be not less effective than recorded ones [8, 9]. Moreover, the so-called "peer effect"—which is inevitably present in all the blended courses—apparently, positively affects this quality [10, 11]. However, running a blended course on a particular subject may require the college/university administration to provide more professors and teaching assistants than it would provide for an offline

A. S. Belenky (✉)
Department of Mathematics, Faculty of Economic Sciences and International Centre of Decision Choice and Analysis, The National Research University Higher School of Economics, Moscow, Russia
e-mail: abelenky@hse.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_2

course on the same subject. Yet, in most cases, such an expansion of the teaching personnel can be done only within financial boundaries of the college/university budget.

Meeting desirable quality requirements within certain financial restrictions is a typical decision-making problem that is solved in industrial, transportation, agricultural, business, financial, and other systems (See, for instance [12–18].) These systems have long been using decision support tools letting their users find the best allocation of those financial resources that they can afford to spend in order to be competitive in corresponding markets. It seems that the administration of every college/university would benefit from having similar tools at its disposal to meet the challenges that the education market of potential new students currently poses and will pose in the future. This is the case since all the colleges/universities compete, particularly, for new students interested in good higher education.

One should draw attention of college/university administrations to the fact that if properly advertised, an obligation to offer blended courses that are based on recorded lectures and seminars of professors from the most famous universities in the world is a factor positively affecting the student enrollment. Moreover, adopting such a strategic decision of providing thus designed and advertised blended courses is likely to affect positively the college/university position in the education market in general, along with its budget, due to at least two reasons.

First, almost all the potential college/university new students who are interested in acquiring knowledge (rather than only a diploma) dream on studying at Harvard, MIT, Cambridge, Oxford, Princeton, Stanford and other world-famous universities. Yet, many of these students are unable to study there due to either low high school grades or financial reasons (or both). If a particular college/university could convince potential new students of the above-mentioned kind that they would listen to the same lecturers on most of the courses as do students from these famous universities while being (a) offered effective tutorials prior to the start of every course, and (b) explained all the course nuances in a simple manner understandable to them, this would make a difference in making enrollment decisions by these potential new students that are favorable to this college/university.

Second, if all such potential college/university new students were offered to study both these courses and tutorials at an affordable cost—which may, eventually, be lower than the cost of studying the same subjects off line at other colleges/universities—their intent to enroll at this college/university would become even stronger.

Thus, the question that a college/university administration—interested in running blended courses there in principle—needs to answer is: How to make the above education strategy a reality, and how to convince potential new students to enroll at this college/university?

Two sets of problems are to be addressed by the interested college/university to answer this question.

The first set includes problems associated with finding such a structure of each particular blended course (which is based on the above-mentioned recordings) that would help the college/university administration convince the potential new students that from the very first lectures, they would become sure to succeed in studying the course. (Certainly, the administration should make it clear that this success can come only if the enrolled students (a) strictly follow recommendations of the teaching personnel assigned to run this course, and (b) bend every effort to succeed.)

The second set includes problems associated with financial aspects of organizing and running blended courses in a manner making it financially affordable to both the college/university and the students.

Addressing problems from the first set (associated with choosing the structure of each blended course) implies conducting comprehensive studies on how different categories of students focus on both the subject of a lecture in general and particular information or techniques discussed in the course. Also, the studies are to determine the best way to run the lecture by choosing an optimal sequence of fragments from the recorded courses and the explanations of the scope of these fragments that should either precede a particular fragment or immediately follow it. These studies should lead to designing testing questions to be asked by the teacher running the course before going to the next fragment. However, all this should be done in a manner that would not turn the lecture into a discussion with the audience that may consume much of the lecture time. Finally, the structure of tutorials offered by the college/university to let the students succeed in studying each blended course designed in the chosen manner should be determined and announced in advance.

Published recommendations of the teachers, along with, possibly, even special consultations of these teachers, can make a difference in the effectiveness of such study results. Certainly, the experience and pedagogical skills of those to be chosen to run blended courses matter a great deal. It's especially so if every teacher chosen to teach a blended course manages to test a particular version (or even different particular versions) of this blended course that she/he proposes to run on groups of the students similar (or at least close) to those who are expected to study the course at the college/university.

Addressing problems from the second set implies the determination of

(a) the minimal budget to organize and run the teaching of a set of chosen blended courses to secure such a percentage of the students (who are to study courses from this set) expected to succeed in studying each particular course from the set that would not be lower than a certain desirable one, and
(b) the maximal percentage of the students (who are to study blended courses from this set of the chosen courses) expected to succeed in studying all the courses from the set under a particular budget.

Though with respect to the first set of problems, there is a room (if not a necessity) for the use of mathematical methods, these problems are currently discussed in scientific publications on this subject mostly at the level of hypotheses [19–21]. The authors of these publications either only set or set and verify the proposed hypotheses by conducting polls among both the students and the teachers. Problems from the second set are rarely considered in scientific publications at all, and no helpful quantitative analysis of these problems have so far been offered. The latter leaves college/university administrations "unarmed" in dealing with the economics of organizing and running blended courses or incorporating other forms of distance learning into the curricula there.

The present paper focuses on those problems from the second set of the above-mentioned ones that are associated with finding and analyzing financial options available to a college/university administration interested in organizing and running blended courses of the considered kind. Particularly, the paper aims at providing a description of a decision-support tool for college/university administrations that lets them quantitatively estimate the budget needed to design and run a set of blended courses of any chosen structure. Proceeding from this budget, the administration of an interested college/university can certainly calculate tuition fees for those its potential new students who are to pay for their education in line with the college/university financial policies being currently in force. (However, a description of corresponding calculation schemes and methodologies implementing these policies lies beyond the subject of the present paper.)

The proposed tool can help administrations of interested colleges/universities develop an optimal strategy of organizing and running blended courses with the use of available recorded materials (courses) from leading universities in the world (as well as, certainly, with the use of those from any other universities). It's implemented by means of standard software packages that can particularly be run on personal computers.

As far as the author is aware, the proposed decision-making tool is the first one capable of solving particular economic problems that the administration of a college/university interested in organizing and running blended courses faces in its attempt to make the distance learning a part of the education process there.

Besides the Introduction, the paper includes three more sections. In Sect. 2, a brief review of research publications in five important (from the author's viewpoint) areas associated with organizing the use of recorded courses in distance learning in general and in running blended courses at colleges/universities in particular is presented. In Sect. 3, a new mathematical model proposed by the author for calculating an optimal strategy of hiring teachers to run a set of blended courses with the use of recorded materials is described. Also, in this section, two integer programming problems, formalizing those outlined in the Introduction, are formulated on the basis of this model. Section 4 indicates, in particular, a set of problems associated with running blended courses that could present interest for college/university administrations and should be researched in the future.

2 A Brief Review of Research Publications in Five Major Areas Related to the Use of Blended Courses in the Education Process

Despite (a) the importance of using online courses in blended learning as a way to dramatically improve the quality of education worldwide, and (b) the obvious necessity to address the economics of hiring new university teachers to run particular blended courses, both those from abroad and from other universities in the country, these problems don't seem to have been studied quantitatively in scientific publications. As far as the author is aware, only qualitative considerations of the above two aspects of online education, both in general and with respect to blended courses, have so far been presented either in the form of case studies or in that of open discussions. As one can see from the publications cited in the Introduction, papers covering at least five important aspects of education related to blended learning—(a) online distance learning advantages in general, (b) blended courses specifics, (c) the effectiveness of blended learning, (d) peer-effect in blended learning, and (e) choosing the structure of blended learning—are those on only qualitative aspects of the corresponding topics.

The aim of the review presented below is to give the reader an impression on (a) the substance of the topics covered by contemporary publications on blended learning, and (b) the state of affairs in this field, which has motivated the proposed quantitative analysis of the problem of using available online courses in designing blended ones.

Online Distance Learning Problems An important mission of online education—which is close to the topic of the present paper—consists of providing opportunities for bringing a high quality of education offered by the most prestigious colleges/universities in the world to colleges/universities of a (currently) modest level of education quality. By reducing their expenses, the latter colleges/universities may be able to make a higher level of education quality financially affordable to both them and their students by widely using online courses developed by leading colleges/universities in the world [3], particularly, by incorporating recorded fragments from these online courses into blended courses that they can offer. To implement this mission, as well as other social missions of online education, two groups of problems associated with distance learning are to be studied.

The first group of these problems includes those associated with "distance learners" such as communications among online learners [2], approaches to structuring assessments of studying parts of an online course, which affect student's learning strategies [22], creating appropriate social environments affecting the motivation to learn [23] and encouraging the presence of the so-called "massive learning" phenomenon [24], choosing the structure of instructions provided in all forms of online learning [25], along with teacher-student relationships that affect student's achievements [26], the effectiveness of online learning in the form of discussions vs. so-called "silent learning" [27], pedagogical aspects of the knowledge perception by the learners and the behavior of participants of the online education, along with micro- and macro-level environment surrounding this activity [28].

The second group of the problems includes those associated with teachers and administrations of colleges/universities involved in running distance learning courses such as preparing teachers for running online courses [29], encouraging teachers to switch to teaching online [30] or/and to using online materials in blended educational courses, particularly, by providing information on the effectiveness of online education [31] and comparing this effectiveness with that of traditional learning [32], along with examples of successfully substituting off-line classes with distance learning [33], describing principal advantages of online learning [34], designing learning communities [35], and providing accessibility to online education to people with disabilities [36].

Both groups of the above problems are studied in the framework of surveys of answers to some questionnaires [37–40] or case studies [22, 23, 36, 41].

Blended Courses Specifics A comparison of the students' performance in traditional studies and in blended learning is offered in [4], where the authors, particularly, suggest that students with high great point averages in prior studies achieve better results in studying blended courses. Based upon reports of 74 second-year students on their self-control and self-regulated learning skills, the authors of [5] conclude that these skills, along with the actual participation in (attending) the course, are key factors to predict the final grades for an online course. In [6], predicting the final outcome of studying a blended course and detecting students who are at risk of poorly studying a particular course is studied by considering 29 "usage variables". The author of [6] suggests that only 14 of these variables turned out to be significant, including four variables that allowed one to predict the learning outcome with a reasonable accuracy. An analysis of the blended learning impact on studying both STEM and non-STEM disciplines, presented in [42], suggests that studying STEM disciplines in the framework of blended learning is more effective than studying them in the framework of traditional studies. At the same time, a comparison of exam and quiz results in blended and traditional courses, presented in [7] for a set of introductory economic courses, did not discover substantial advantages of blended learning.

Eight problems that the teachers who conduct blended courses face are described in [43] as those being part of three inductive categories—instructional processes, community concerns, and technical issues—detected as a result of processing the response of 117 teachers from four Universities in Turkey to questions they were asked in the interviews. Results of a study reported in [44] suggest that only a minority of all the students use particular tools in studying blended courses supported by content management systems (CMS), and the students regulate the use of these tools as the blended course unfolds. Students' approaches to inquiry and collaborations in the course of working on a blended course are studied in [45], and a division of the students into subgroups within a group of more than 200 students, with similar approaches to the inquiries and learning technologies within each subgroup, is reported there with respect to a particular one-semester blended course. The authors of this study believe that such a division helps explain why students from some of the subgroups are more successful in studying the course than students from the other ones.

Four key challenges in designing blended courses—(1) incorporating flexibility, (2) stimulating interaction, (3) facilitating students' learning processes, and (4) fostering an emotional learning climate—and approaches to resolving corresponding problems are outlined in a review of 640 sources and 20 studies, presented in [46].

Patterns of the students' evaluation of and experience with online, blended, and face-to-face courses are compared in [47].

The Effectiveness of Blended Learning A case study related to implementing a blended learning approach to developing a course is presented in [48], where both students' responses to the blended learning environment and thoughts of the course author on the requirements that a successful blended course should meet are discussed. An approach to integrating online and face-to-face learning with blended learning, which was successfully implemented at Suleyman Demirel University via a University Learning Management Systems (LMS) with respect to a particular computer engineering course, is described in [9]. A set of bottlenecks in contemporary higher education, which is described in [49], is considered by the authors, particularly, from the viewpoint of the ability of blended courses to contribute to effectively solving corresponding problems of managing higher education.

Twenty research studies analyzed in the framework of a review, presented in [50], let the review authors suggest that there exist two groups of main factors, each affecting the solving of creative problems in a blended learning environment, that should be considered. The first group includes four particular factors affecting the solving process, whereas the remaining five factors, determining the blended learning environment, form the second group. Results of an experiment with the MAgAdI on-line system—an adaptive, integrated into the learning process web environment, developed to support the learning processes in which several knowledge fields, courses, and teachers get involved—are discussed in [8]. The authors of [8] assert that the use of that on-line system helps successfully blend traditional teaching methods with various learning environments.

Peer Effects in Learning Peer effects—as a phenomenon associated with an affection that the interaction with peers may have on a person' behavior in a group learning—have some general features. This is why these effects are currently studied mostly in general though they may have certain specifics in blended learning and affect compositions of learners (who take blended courses) to achieve better academic results.

Fundamentals of peer effects in higher education, including the definition of this phenomenon, are discussed in [10], where the authors assert that these effects affect the economics of higher education, mostly since (according to these authors), they "eliminate awkward anomalies in the institutional behavior of colleges and universities and in the economic structure of higher education as an industry if they exist." Three interaction patterns of peer talks—a tutor-led question-and-answer pattern, a cumulative-exploratory pattern, and a dispute-exploratory one—within a particular educational environment in which third-year students are to train first-year ones are identified in [51]. The authors of that paper believe that using the identified patterns may improve the effectiveness of studies with respect to all the subjects of learning.

The underlying mechanism of peer effects is studied in [11] with respect to a group of first year students for which the academic abilities of different student subgroups are measured and compared. According to the authors of that paper, (a) male, minority, and low-ability students are affected by their peers the most, and (b) the transfer of general knowledge within all the subgroups has more effect than that of specific knowledge.

A theoretical model attempting to explain (and compare) different peer effects in different fields of studies based upon researching the behavior of the first year students in a middle-sized public university in Italy is presented in [52]. Peer effects in students' academic performance are studied in [53] based upon a set of publications in the field, and one of the findings, presented in that paper, suggests that high ability students have higher positive peer effect on other high ability students at both school and college/university level. At the college/university level, the findings suggest that the background of roommates affects students' academic performance.

Choosing the Structure of a Blended Course An approach to structuring a blended-learning bachelor program in electrical engineering, designed for a group of students who study this subject alongside with working (as being employed), is discussed in [21]. Several options for information exchange such as establishing a connection with the course of mathematics, retrieving references needed to study the course, electronic module questionnaires, and a feedback channel functioning during the whole course are used in structuring the course. A methodology (Q-methodology) to designing blended education courses is considered in [19] to identify learning perspectives of the students enrolled, along with peculiarities of their perceptions of the course, which helps estimate possible student achievements as a result of studying the course. The author of [19] identifies four perception types and offers his observations on the academic achievements of the students with some of those perception types.

A mathematical thinking approach underlying the structure of a blended course in part of a calculus course related to multivariable calculus is proposed in [54], where the authors assert that creating a blended learning environment is adequate to developing mathematical thinking in students. A set of guidelines that are based on theories of instructional design for writing instructors, offered in [55], are aimed at supporting effective student learning. The authors of that publication suggest that in designing the course, the instructors should focus on five issues: (1) a substantiation of the need for the course, (2) the structure of the audience to learn the course, (3) advantages of developing the course in an on-line form, (4) basic pedagogical principles, and (5) available resources.

A concept of blended learning, along with basic elements of its teaching mode frame, is discussed in [56]. A set of guides, documents, and publications related to practical aspects of designing blended courses, along with common principles of designing such courses, an appropriate terminology, and strategies to use to meet accreditation requirements, is considered in [57]. The authors of that publication discuss approaches to using online technologies and to the course implementation, along with the problems and difficulties to bear in mind in designing blended courses, and outline directions of further research in this field. Detailed instructions on designing blended courses to achieve learning objectives, along with certain tools and templates, are offered in the books [20] and [58]. The books contain both theoretical and practical considerations related to designing blended courses and reflect the experience of their authors in developing such courses and in converting traditional courses into blended learning ones.

As one can see from the presented review, and as mentioned in the Introduction to this paper, all the discussions and reasonings in all the above five areas are conduced at the level of philosophical and pedagogical observations and suggestions. No tools for either a formalized analysis or a conversion of these discussions and reasonings into practical activities have so far been proposed.

At the same time, it is clear that the college/university competitiveness in the market of potential new students cannot be based on such observations and discussions only. College/university administrations that possess tools for making strategic decisions based on a formalized financial analysis of which economically affordable decisions can and which ones should be made will undoubtedly have substantial competitive advantages in the above-mentioned market.

3 The Statements of the Problems and Their Mathematical Formulation

A University/College (for the sake of definiteness, a University further in this section of the paper) intents to start teaching blended courses in one of foreign languages, for instance, in English, and it's to decide how to organize this activity. The number of blended courses to be taught in English equals K, and the number of teachers-native speakers of English language—can (a) cover all these K courses, and (b) be hired by the University from abroad—equals (or doesn't exceed) L. Besides hiring teachers from abroad to run these K blended courses, the University considers a possibility to offer to teach some courses from this set of K blended courses or all the K blended courses to teachers-native speakers of the country's language who speak English well. To this end, the University considers potential candidates to teach blended courses from among those who are currently employed either by the University or by other universities in the country.

The University plans to buy (or to take from the open sources) recorded online courses that are taught by distinguished professors from leading universities in the world and to use these recordings in designing all the above-mentioned K blended courses, no matter who will finally be chosen to teach these courses. Each of the teachers (a) invited from abroad, (b) invited from other universities in the country, and (c) currently employed by the University, who are invited to teach each particular course from the set of K blended courses, is to teach it in one and the same manner. That is, she/he is to teach each such course in the form of lectures and seminars, and these lectures and seminars are to be based on or substantially use the above-mentioned acquired materials from the recorded online courses.

Each of the teachers considered by the University to be invited to teach courses from the set of K blended courses (from among those who can be chosen from professors currently working at the University, or can be hired from other universities in the country, or can be invited from abroad) can teach no more than a certain number of these blended courses. If a teacher from the University is assigned to teach some courses from the set of K blended courses, her/his existing assignments for teaching courses in the country's language are to be covered by other teachers either currently working at the University or by those to be hired from other universities in the country (though not by any of those invited to teach courses from the set K) on the hourly basis. Each potential candidate to teach any particular course from the set of K blended courses is tested by experts recognized by the University. Such experts are to estimate a percentage of the students who are likely to succeed in studying this course should this candidate be selected to teach the course.

The University is interested in estimating two (indicated in the Introduction) numbers associated with organizing and running the above-mentioned K blended courses:

(1) What is the minimal budget to organize the teaching of these K blended courses in English to secure a percentage of the University students (who are to study courses from this set) expected to succeed in studying each particular course (from these K blended courses) to be not lower than a certain desirable percentage?

(2) What is the maximal percentage of the students (who are to study courses from this set) expected to succeed in studying each particular course from these K blended courses under any particular budget that the University can afford to spend to organize and run these courses?

All the teachers-native speakers of the country's language who are potentially capable of teaching courses in English from the set of K blended courses (both from the University and from other universities in the country) are called course developers further in the paper. If any course from the K blended courses is to be taught by a teacher-native speaker of English (invited from abroad), the students assigned to study this course are to take (a) an advanced course in English language, and (b) corresponding tutorials to be prepared to understanding fragments from the corresponding recorded courses, prior to the commencement of the course.

Let

- K be the number of blended courses that the University plans to teach in English yearly, say, in the next T years,
- B be the yearly budget allocated by the University to cover the expenses associated with developing and running the set of K blended courses within T years,
- B_0 be the cost of recorded online courses that the University acquires to use in developing the set of K blended courses,
- M be the number of potential course developers who are currently employed by the University,

- R be the number of teachers currently working at other universities in the country who are interested in working at the University, and who are considered by the University as potential course developers,
- L be the number of teachers-native speakers of English from abroad who can cover the needs of the University in teaching courses from the set of K blended courses and who the University can financially afford to invite,
- c_i be the basic yearly salary of course developer i (who is a teacher currently employed by the University) and who is chosen (assigned) to teach courses from the set of K blended courses, $i \in \overline{1, M}$,
- ∇_{ik} be the additional yearly salary of course developer i from the University, assigned to teach courses from the set of K blended courses, for developing and teaching course k, $k \in \overline{1, K}$, $i \in \overline{1, M}$,
- b_r be the basic yearly salary of course developer r from another university in the country invited to teach courses from the set of K blended courses, $r \in \overline{1, R}$,
- δ_{rk} be the additional yearly salary of developer r, invited from another university in the country to teach courses from the set of K blended courses, for developing and teaching course k, $k \in \overline{1, K}$, $r \in \overline{1, R}$,
- g_r be the relocation cost associated with hiring course developer r from another university in the country to teach courses from the set of K blended courses, $r \in \overline{1, R}$,
- a_l be the basic yearly salary of teacher l to be invited from abroad to teach courses from the set of K blended courses in English, $l \in \overline{1, L}$,
- Δ_{lk} be the additional yearly salary of teacher l, invited from abroad to teach courses from the set of K blended courses in English, for teaching course k, $k \in \overline{1, K}$, $l \in \overline{1, L}$,
- h_l be the relocation cost associated with the invitation of teacher l from abroad to teach courses from the set of K blended courses in English, $l \in \overline{1, L}$,
- d_i be the per hour salary of a (currently employed by the University) teacher, who is to substitute course developer i (who is assigned to teach courses from the set of K blended courses) in all the activities associated with teaching the courses "vacated" by course developer i, $i \in \overline{1, M}$ (if there are such course developers),
- t_{ik} be the number of hours per year that course developer i "vacates" (as a result of switching to teaching course k from the set of K blended courses) that are to be covered by other teachers (either by those who are currently employed at the University or by those to be hired from other universities in the country), $i \in \overline{1, M}$, $k \in \overline{1, K}$,
- α_{ik} be the expert estimate of a percentage of the students expected to succeed in studying blended course k that is taught by course developer i from the University, $i \in \overline{1, M}$, $k \in \overline{1, K}$,
- β_{rk} be the expert estimate of a percentage of the students expected to succeed in studying blended course k that is taught by course developer r invited from another university in the country, $r \in \overline{1, R}$, $k \in \overline{1, K}$,
- γ_{lk} be the expert estimate of a percentage of the students expected to succeed in studying blended course k that is taught by teacher l invited from abroad, $l \in \overline{1, L}$, $k \in \overline{1, K}$,

- ω_k be a desirable (targeted) by the University percentage of the students expected to succeed in studying blended course k, $k \in \overline{1, K}$,
- u_k be a Boolean variable that equals 1 if course k from the set of K blended courses is to be taught by a teacher invited from abroad (so that a special English language course and tutorials are to be run by the University for the students who choose to study blended course $k \in \overline{1, K}$) and equals 0, otherwise,
- f_k be the yearly cost of running the advanced English language course and the corresponding tutorials for the students who choose to study blended course k, $k \in \overline{1, K}$,
- x_{ik} be a Boolean variable that equals 1 if course developer i from the University is assigned to teach blended course k, $i \in \overline{1, M}$, $k \in \overline{1, K}$, and equals 0, otherwise,
- s_r be a Boolean variable that equals 1 if course developer r to be invited from another university in the country is qualified to teach courses from the set of K blended courses and equals 0, otherwise, $r \in \overline{1, R}$,
- y_{rk} be a Boolean variable that equals 1 if course developer r from another university in the country is invited to teach blended course k, $k \in \overline{1, K}$, $r \in \overline{1, R}$ and equals 0, otherwise,
- w_l be a Boolean variable that equals 1 if teacher l from abroad is invited to teach courses from the set of K blended courses and equals 0, otherwise, $l \in \overline{1, L}$,
- z_{lk} be a Boolean variable that equals 1 if teacher l from abroad is invited to teach blended course k and equals 0, otherwise, $l \in \overline{1, L}$, $k \in \overline{1, K}$,
- $M\Theta(i) \subset \overline{1, K}$ be a subset of courses from the set of K blended courses that course developer i (from the University) can't teach, $i \in \overline{1, M}$,
- $R\Theta(r) \subset \overline{1, K}$ be a subset of courses from the set of K blended courses that course developer r (to be invited from another university in the country) can't teach, $r \in \overline{1, R}$,
- $L\Theta(l) \subset \overline{1, K}$ be a subset of courses from the set of K blended courses that teacher l (to be invited from abroad) can't teach, $l \in \overline{1, L}$,
- ν_i be the maximal number of courses from the set of K blended courses that course developer i (from the University) can teach concurrently, $i \in \overline{1, M}$,
- μ_r be the maximal number of courses from the set of K blended courses that course developer r (to be invited from another university in the country) can teach concurrently, $r \in \overline{1, R}$,
- π_l be the number of courses from the set of K blended courses that teacher l (to be invited from abroad) can teach concurrently, $l \in \overline{1, L}$, and
- q_k be the yearly budget that the University can spend for additional salaries to be paid to the teacher who teaches course k from the set of K blended courses (along with the English language course and corresponding tutorials, if need be) and to the one who substitutes this teacher (if there is one).

Assumptions

1. The numbers of potential candidates from other universities in the country to choose from to (a) invite to become blended course developers, and (b) replace teacher i for the hours "vacated" by this teacher are known large integers.

Course developers $i,\ i \in \overline{1, M}$ and $j,\ j \in \overline{1, R}$ are paid an extra salary for developing and running courses from the set of K blended courses. This developing is done in line with the structure of each blended course they are invited to teach, which is to be provided by the University administration. However, the basic salary of developer $i,\ i \in \overline{1, M}$ remains the same as it was before the assignment to teach blended courses (and it's not a part of the budget B, allocated to cover the expenses associated with hiring teachers to teach courses from the set of K blended courses).

2. Teachers invited from abroad to teach courses from the set of K blended courses come with already developed such courses (in line with the structure of these courses to be provided by the University administration in advance). Nevertheless, they are paid extra salaries $\nabla_{lk},\ l \in \overline{1, L},\ k \in \overline{1, K}$ for teaching these courses while their basic salaries are specified by corresponding contracts sighed between each of them and the University administration.

   All the salaries paid to each teacher invited by the University—both to each of those invited from other universities in the country and to each of those invited from abroad—are part of the budget B, allocated to cover the expenses associated with hiring teachers to develop and run courses from the set of K blended courses.
3. For the sake of simplicity, it's further assumed that both the tutorials and special English language courses are run for blended course k only if this course is to be taught by a teacher invited from abroad (though, generally, at least the tutorials may be run if a teacher of any of the two other kinds is to run this course), introducing Boolean variables similar to u_k may reflect this case in the system of constraints (1), (2), to be presented further in this paper).
4. All the contracts with every teacher invited from abroad to teach courses from the set of K blended courses and with every blended course developer from other universities in the country, hired by the University to teach blended courses from this set, are signed for T years.
5. In considering the use of recorded fragments of online lectures in designing blended courses, along with the invitation of outside teachers to teach blended courses (including those invited from abroad), the University administration chooses a total set of blended courses that it plans to start offering at the University yearly within T years so that no new blended courses are added by the University within these T years.
6. All the information needed for estimating the values of the parameters α_{ik}, $i \in \overline{1, M},\ k \in \overline{1, k},\ \beta_{rk},\ r \in \overline{1, R},\ k \in \overline{1, k}$, and $\gamma_{lk},\ l \in \overline{1, L},\ k \in \overline{1, k}$ is considered to be available with respect to all the teachers invited to teach blended courses from the set K at the University. Teachers from abroad who are considered to be invited to teach blended courses in line with the required structure of these courses (to be provided by the University administration) are asked to provide some of their recorded lectures on the same subjects in advance, to let the University administration and the experts make corresponding estimates. To make these estimates, the University arranges testing of the

provided recorded materials on a selected group of its students to find out what kinds of training courses and tutorials are to be run to help the students better understand both the substance of the blended courses and the language to be used in presenting (the designed) blended courses to the audience, which these courses are prepared for.

The systems of constraints

$$
\begin{gathered}
\sum_{i=1}^{M} x_{ik} + \sum_{r=1}^{R} y_{rk} + \sum_{l=1}^{L} z_{lk} = 1,\ k \in \overline{1,K},\\
\sum_{i=1}^{M} x_{ik} \le 1,\ k \in \overline{1,K},\ \sum_{k=1}^{K} x_{ik} \le \nu_i,\ i \in \overline{1,M},\\
x_{ik} = 0,\ k \in M\Theta(i) \subset \overline{1,K},\\
y_{rk} \le s_r \le \sum_{k=1}^{K} y_{rk},\ r \in \overline{1,R},\ k \in \overline{1,K},\\
\sum_{r=1}^{R} y_{rk} \le 1,\ \ k \in \overline{1,K},\ \sum_{k=1}^{K} y_{rk} \le \mu_r,\ r \in \overline{1,R},\\
y_{rk} = 0,\ k \in R\Theta(r) \subset \overline{1,K},\\
\sum_{l=1}^{L} z_{lk} - u_k = 0,\ k \in \overline{1,K},\\
z_{lk} \le w_l \le \sum_{k=1}^{K} z_{lk},\ l \in \overline{1,L},\ k \in \overline{1,K},\\
\sum_{l=1}^{L} z_{lk} \le 1,\ k \in \overline{1,K},\ \sum_{k=1}^{K} z_{lk} \le \pi_l,\ l \in \overline{1,L},\\
z_{lk} = 0,\ k \in L\Theta(l) \subset \overline{1,K},\\
\sum_{l=1}^{L} \Delta_{lk} z_{lk} + \sum_{r=1}^{R} \delta_{rk} y_{rk} +\\
\sum_{i=1}^{M} \nabla_{ik} x_{ik} + \sum_{i=1}^{M} d_i t_{ik} x_{ik} + f_k u_k \le q_k,\ k \in \overline{1,K},
\end{gathered}
\tag{1}
$$

and

$$\sum_{l=1}^{L} w_l(a_l + h_l) + \sum_{k=1}^{K}\sum_{l=1}^{L} \Delta_{lk} z_{lk} + \sum_{r=1}^{R} s_r(b_r + g_r) + \sum_{k=1}^{K}\sum_{r=1}^{R} \delta_{rk} y_{rk} + \\ \sum_{k=1}^{K}\sum_{i=1}^{M} \nabla_{ik} x_{ik} + \sum_{k=1}^{K}\sum_{i=1}^{M} d_i t_{ik} x_{ik} + \sum_{k=1}^{K} f_k u_k \le B - B_0 \tag{2}$$

should hold for the first year in the set of T years (also, see concluding Remark 9).

Taking into account the system of constraints (1) and (2), the two problems stated earlier in this section of the paper can mathematically be formulated as follows:

Problem 1 The problem consists of maximizing the minimum percentage of the total number of students expected to succeed in studying a course from the set $\overline{1, K}$ (when these courses are considered to be equally important from the University's viewpoint), and this problem is formulated proceeding from the fixed yearly budget B (for T consecutive years). This problem is the Boolean programming problem

$$\min_{k \in \overline{1,K}} \left(\sum_{i=1}^{M} \alpha_{ik} x_{ik} + \sum_{r=1}^{R} \beta_{rk} y_{rk} + \sum_{l=1}^{L} \gamma_{lk} z_{lk} \right) \to \max_{(x_{ik}, y_{rk}, z_{lk}, u_k, s_r, w_l)} \tag{3}$$

under the systems of constraints (1), (2).

Problem 2 The problem consists of minimizing the maximal value of the yearly budget for organizing and running courses from the set of K blended courses (which is that for the first of the T consecutive years), provided the desirable (targeted) percentages of the students expected to succeed in studying the courses from this set (determined by the numbers ω_k, $k \in \overline{1, K}$) are attained. This problem is the Boolean programming problem

$$\sum_{l=1}^{L} w_l(a_l + h_l) + \sum_{k=1}^{K}\sum_{l=1}^{L} \Delta_{kl} z_{kl} + \sum_{r=1}^{R} s_r(b_r + g_r) + \sum_{k=1}^{K}\sum_{r=1}^{R} \delta_{kr} y_{kr} + \\ \sum_{k=1}^{K}\sum_{i=1}^{M} \nabla_{ik} x_{ik} + \sum_{k=1}^{K}\sum_{i=1}^{M} d_i t_{ik} x_{ik} + \sum_{k=1}^{K} f_k u_k \to \min_{(x_{ik}, y_{rk}, z_{lk}, u_k, s_r, w_l)} \tag{4}$$

under the system of constraints (1) and the additional system of constraints

$$\sum_{i=1}^{M} \alpha_{ik} x_{ik} + \sum_{r=1}^{R} \beta_{rk} y_{rk} + \sum_{l=1}^{L} \gamma_{lk} z_{lk} \ge \omega_k, \ k \in \overline{1, K}. \tag{5}$$

In both problems, it is assumed that the systems of constraints (1), (2) (in Problem 1) and (1), (2), (5) (in Problem 2) are compatible, which can be verified, and appropriate corrections in the systems of constraints (1), (2) can be made with the use of the technique proposed, in particular, in [59].

In Problems 1 and 2, it is assumed that the parameters α_{ik}, β_{rk}, and γ_{lk}—which are expert estimates of percentages of the students who are to take course $k \in K$ and are expected to succeed in studying this course if the course is taught by a developer from the University, or by a developer from another university in the country, or by a teacher invited from abroad, respectively—are known numbers provided to the University administration by expects recognized by the University.

4 Concluding Remarks

1. It's clear that, generally, Boolean programming problems (1)–(3) and (1), (2), (4), (5) can also serve as mathematical formulations of problems associated with finding an optimal composition of the teachers to run any set of offline courses at a college/university, for instance, in physics, mathematics, biology, etc., without the use of recordings of any lectures on corresponding subjects. If this is the case, any training courses and tutorials in the corresponding subjects may or may not be run, and teachers from abroad may or may not be invited to teach the corresponding offline courses. To formulate both problems in this case, some variables and parameters, being present in the system of constraints (1), (2), should be omitted.

   Such a possibility to use the proposed models and Boolean programming problems stems from the obvious observation: The nature of stuffing a set of blended courses with teachers and that of stuffing a set of offline courses with ones is the same, and so should be their mathematical formalization. Indeed, in both cases, an optimal combination of the teachers from any available set of them to teach courses within certain financial limits should be determined. However, stuffing offline courses at, for instance, a modest university differs from stuffing blended ones there.

   Indeed, to make blended courses attractive to potential new students, the college/university administration should

   (a) properly advertise these courses as those to be taught with the use of fragments of lectures given by professors from famous universities in the world,
   (b) convince the potential new students that both the tutorials and the training courses to accompany each offered blended course (if there are ones to accompany it) are designed to let every interested student succeed in studying this course,
   (c) pay extra salaries to the chosen teachers for developing each blended course (in line with its structure to be provided by the college/unversity administration), and
   (d) apply a reliable procedure of estimating the ability of each potential teacher to teach a blended course

to let the expert estimates of the percentage of the students expected to succeed in studying this course be not lower than a particular desirable number (which is known for standard offline courses on the same subjects).

Finally, the blended courses to be included in the college/university curricula should be designed and run in such a manner that the students who are the most successful in studying the offered blended courses may eventually become prepared to continue their education in those world-famous universities, whose recorded fragments of lectures were part of these blended courses. For instance, upon graduation, bachelor students may succeed in getting admitted to these universities for master programs, whereas master students may get admitted for Ph.D. studies there. Needless to say that if this happened, it would be an excellent contribution to the university's prestige and would sharply contrasted the described strategy of designing and running blended courses from that of running traditional offline ones. The college/university may even organize subgroups of the most advanced enrolled students interested in making these moves to the world-leading universities to help them succeed in implementing their desire. This can be done, particularly, by offering them specially designed blended courses with a more intensive use of the recorded online lectures given by distinguished professors at those universities.

2. While designing blended courses based on fragments of recorded lectures of distinguished lecturers from world-leading universities is a good option for advanced potential and current college/university students, blended courses for both less prepared potential and current students may use recordings of ordinary lecturers, including those offered by college/university teachers. In both cases, problems (1)–(3) and (1), (2), (4), (5) (with changes reflecting a particular strategy of designing and running blended courses) can be used for receiving corresponding quantitative financial estimates.
3. The formulations of Problems 1 and 2 suggest that the proposed decision support tool may help a college/university to substantiate the budget needs in talks with legal entities that care about the quality of education that this college/university provides and its prestige (such as local and federal administrations for public colleges/universities and contributing sponsors for private ones). Particularly, this tool helps the college/university substantiate the size of $B - B_0$—the college/university's budget that is planned by the college/university to be spend for organizing and running K blended courses—particularly, in the talks with these legal entities associated with the competitiveness of the college/university in the market of new potential students.
4. In this paper, the competitiveness of a college/university in a market of potential students is considered with respect to new potential students only. However, it's clear that, generally, the competitiveness of a college/university also depends on how the education process is arranged for the students who have been with this college/university at least for some time. Appropriate arrangements motivate such students to complete their education where they are currently enrolled rather than encourage them to transfer to another college/university to receive a corresponding degree there. With respect to blended courses, which can be either

core ones or electives, the same features that attract the attention of potential new students should be present in the blended courses offered to already enrolled college/university students. Also, one should bear in mind that the quality of teaching these courses affects the intents of new potential students to enroll at the college/university, since such intents are partly motivated by the information on this quality received from its current students.

5. The proposed scheme of using fragments from recorded online courses is only one of the options to use such recordings. Since most of these courses are available free of charge, the college/university administration may decide to use online courses as a substitution for offline ones, without designing and running new (blended) courses. From the author's teaching experience, it seems hard to believe that the recorded online lectures can always substitute the presence of a lecturer live in auditorium for a majority of the students studying these courses while (at least) not positively affecting the percentage of the students who are likely to study corresponding online courses successfully. In any case, it looks reasonable to "test" both approaches to using the recorded online courses before making a final decision on this matter.
6. Organizing tutorials to prepare particular groups of college/university students for successfully studying new blended courses is always a challenge. The major problem here is associated with timely preparing the students to (a) understand the facts to be presented in a particular blended course, and (b) be comfortable with the style of the material presentation by the teacher chosen to run this blended course. The time left for this preparation much depends on when the selection of the teachers to run the courses is completed, and the results of providing corresponding tutorials substantially depend on who runs the tutorials for each new course from the set of K blended courses. The latter may become a complicated problem when the teacher chosen to run a particular blended course is not currently employed at the college/university.

Also, one needs to emphasize that a teacher chosen to teach a blended course at a college/university is free to develop these courses on her/his own, as long as she/he adheres to the structure of the course determined by the college/university administration. (However, discussing approaches to developing particular blended courses in line with their assigned structures lies beyond the scope of this paper.)

Finally, in the model (1), (2), it was assumed that the tutorials and English language courses are organized and run only for blended courses to be taught by the teachers invited from abroad. However, generally, this may not be the case. Both kinds of course developers (from the University and from other universities in the country) may require to run both tutorials for the blended courses to be taught by them and the English language courses to be offered by corresponding departments at the University.

Let $\tilde{M} \subset M$ and $\tilde{R} \subset R$ be subsets of those course developers who require to organize and run tutorials (with or without additional courses in English language to be organized by the college/university), let $\tilde{f}_{ik}^{U}$ and $\tilde{f}_{rk}^{OU}$ be the costs or running these tutorials, and let $\tilde{\psi}_{ik}^{U}$ and $\tilde{\psi}_{rk}^{OU}$ be the costs or running additional English language courses, requested by the developers from the college/university and by those from other colleges/universities in the country, respectively. Then, for instance, the first (of the T) inequalities in (2) will take the form

$$\sum_{l=1}^{L} w_l(a_l+h_l)+\sum_{k=1}^{K}\sum_{l=1}^{L}\Delta_{lk}z_{lk}+\sum_{r=1}^{R}s_r(b_r+g_r)+\sum_{k=1}^{K}\sum_{r=1}^{R}\delta_{rk}y_{rk}+$$
$$\sum_{k=1}^{K}\sum_{i=1}^{M}\nabla_{ik}x_{ik}+\sum_{k=1}^{K}\sum_{i=1}^{M}d_i t_{ik}x_{ik}+\sum_{k=1}^{K}f_k u_k+\sum_{i\in\tilde{M}}\sum_{k=1}^{K}\tilde{f}_{ik}^{U}x_{ik} \quad (6)$$
$$+\sum_{r\in\tilde{R}}\sum_{k=1}^{K}\tilde{f}_{rk}^{OU}y_{rk}+\sum_{i\in\tilde{M}}\sum_{k=1}^{K}\tilde{\psi}_{ik}^{U}x_{ik}+\sum_{r\in\tilde{R}}\sum_{k=1}^{K}\tilde{\psi}_{rk}^{OU}y_{rk}\le B-B_0.$$

Here, it’s assumed that if developer i^* from the set $\tilde{M}$ doesn’t require to organize and run tutorials for blended course $k^* \in \overline{1,K}$, the coefficient $\tilde{f}_{i^*k^*}^{U}$ equals zero. The same assumption holds for the coefficient $\tilde{f}_{r^{**}k^{**}}^{OU}$ (if developer r^{**} from the set $\tilde{R}$ doesn’t require to organize and run tutorials for blended course k^{**}, $k^{**} \in \overline{1,K}$) and for the coefficients $\tilde{\psi}_{i^*k^*}^{U}$ and $\tilde{\psi}_{r^{**}k^{**}}^{OU}$.

7. Generally, the college/university administration may exercise two approaches to using recorded fragments of online lectures in designing blended courses, which differ in the budget for running these courses. That is, it can offer the same set of blended courses on a yearly basis within, say, T years so that no new blended courses are added within these T years (see Assumption 5 from Sect. 3). The other one implies widening the spectrum of blended courses yearly (or even more often) depending on the results at the end of a particular year (or even on those of each year), including financial results associated with running blended courses.

   One can easily be certain that the proposed model can be used in formalizing corresponding mathematical problems in the framework of the second approach though the structure of the system of constraints in the corresponding problems will be different. That is, this system will have a block-diagonal structure binding variables related to each particular year (during T years) within a separate block, along with a block binding all the variables of the system of constraints [60].
8. The proposed tool demonstrates the potential of an adequate mathematical modelling, systems analysis, operations research techniques, and standard optimization software in solving practical problems arising in the economics of education.
9. One should bear in mind that if inequality (2) holds for the first year in the set of T years, the restriction for a yearly budget for all the K courses will also hold for all the other years from this set (since all the expenses associated with the relocation of the invited teachers, which are reflected in (2), as well as the expenses B_0, take place in this first year).

10. In the models underlying the formulations of Problems 1 and 2, it's assumed that both the set of potential course developers from other colleges/universities in the country and that from those to be invited from abroad are known to the college/university administration in advance, before these two problems are solved. Also, it's assumed that both the tutorials (for each special course) and the English language courses, provided by the teachers, are taken by the students during the period of T years only once. Finally, it's assumed that any substitution of the teachers from other colleges/universities in the country and from abroad for the years from the set of T years can be done only by the college/university teachers (unless the relocation expenses of an invited teacher, not currently employed by the college/university, are covered by it), and the values c_i, $i \in \overline{1, M}$, are taken into account.

One of further research directions of studying both problems considered in this paper should concern the development of approaches to solving these problems under uncertainty conditions. This uncertainty is associated with the impossibility to determine the exact values of expert estimates of the parameters α_{ik}, β_{rk}, γ_{lk}—percentages of the students expected to succeed in studying each of K blended courses with respect to every teacher considered by the college/university administration as a candidate for teaching this course—in principle. Though the proposed tool lets the college/university administration conduct multiple calculations for any number of sets of these expert estimates that the administration may be interested to explore, one should understand that information on the values of these estimates is that of a probabilistic nature. Moreover, establishing any particular regularities and parameters of distribution laws describing the dynamics of these parameters seems to present considerable difficulties.

At the same time, there exist approaches to treating similar information (on parameter values in corresponding mathematical models) that let formulate problems under uncertainty conditions similar to those considered in this paper as linear or mixed programming ones solving which can be done by using standard software packages, available even on PCs [61]. However, the applicability of such approaches to considered problems depends on whether assertions similar to those presented in [61] can be mathematically proven.

References

1. Kolowich, S.: Exactly how many students take online courses. The Chronicle of Higher Education, January **16** (2014)
2. Joksimovic, S., Gasevic, D., Loughin, T.M., Kovanovic, V., Hatala, M.: Learning at distance: effects of interaction traces on academic achievement. Comput. Educ. **87**, 204–217 (2015)
3. Carnoy, M., Kuz'minov, Y.: Online learning: how it affects the university structure and economics. Panel discussion. Educ. Stud. Moscow, **3**, 8–43 (2015)
4. Asarta, C.J., Schmidt, J.R.: Comparing student performance in blended and traditional courses: does prior academic achievement matter? Internet High. Educ. **32**, 29–38 (2017)

5. Zhu, Y., Au, W., Yates, G.: University students' self-control and self-regulated learning in a blended course. Internet High. Educ. **30**, 54–62 (2016)
6. Zacharis, N.Z.: A multivariate approach to predicting student outcomes in web-enabled blended learning courses. Internet High. Educ. **27**, 44–53 (2015)
7. Olitsky, N.H., Cosgrove, S.B.: The effect of blended courses on student learning: evidence from introductory economics courses. Int. Rev. Econ. Educ. **15**, 17–31 (2015)
8. Alvarez, A., Martin, M., Fernandez-Castro, I., Urretavizcaya, M.: Blending traditional teaching methods with learning environments: Experience, cyclical evaluation process and impact with MAgAdI. Comput. Educ. **68**, 129–140 (2013)
9. Yigit, T., Koyun, A., Yuksel, A.S., Cankaya, I.A.: Evaluation of blended learning approach in computer engineering education. Procedia Soc. Behav. Sci. **141**, 807–812 (2014)
10. Winston, G.C., Zimmerman, D.J.: Peer effects in higher education. In: The Economics of Where to Go, When to go, and How to Pay for it, pp. 395–421. University of Chicago Press (2004)
11. Griffith, A.L., Rask, K.N.: Peer effects in higher education: a look at heterogeneous impacts. Econ. Educ. Rev. **39**, 65–77 (2014)
12. Bazerman, M., Moore, D.: Judgment in Managerial Decision Making, 8th edn. Wiley (2012)
13. Gilboa, I.: Making Better Decisions: Decision Theory in Practice, 1st edn. (Wiley/Blackwell, 2010)
14. Saaty, T.: Decision Making for Leaders: The Analytic Hierarchy Process for Decisions in a Complex World, 3rd Revised edn. RWS Publications (2012)
15. Noyes, J., Cook, M.: Decision Making in Complex Environments, 1st edn. CRC Press (2017)
16. de Haan, A., de Heer, P.: Solving Complex Problems: Professional Group Decision-Making Support in Highly Complex Situations, 2nd Revised edn. Eleven International Publishing (2015)
17. Hammond, J., Keeney, R., Raiffa, H.: Smart Choices: A Practical Guide to Making Better Decisions. Crown Business (2002)
18. Adair, J.: Decision Making and Problem Solving (Creating Success). Kogan Page (2013)
19. Kim, J.Y.: A study on learners' perceptional typology and relationships among the learner's types, characteristics, and academic achievement in a blended e-education environment. Comput. Educ. **59**(2), 304–315 (2012)
20. Linder, K.E.: The Blended Course Design Workbook: A Practical Guide, Workbook edn. Stylus Publishing (2016)
21. Kalberer, N., Petendra, B., Bohmer, C., Schibelbein, A., Beck-Meuth, E.M.: Evaluation process and quality management in a blended-learning bachelor's programme. Procedia Soc. Behav. Sci. **228**, 131–137 (2016)
22. Zlatovic, M., Balaban, I., Kermek, D.: Using online assessments to stimulate learning strategies and achievement of learning goals. Comput. Educ. **91**, 32–45 (2015)
23. Barak, M., Watted, A., Haick, H.: Motivation to learn in massive open online courses: examining aspects of language and social engagement. Comput. Educ. **94**(3), 49–80 (2016)
24. Sendra-Portero, F., Torales-Chaparro, O.E., Ruiz-Gomez, M.J., Martinez-Morillo, M.: A pilot study to evaluate the use of virtual lectures for undergraduate radiology teaching. Eur. J. Radiol. **82**(5), 888–893 (2013)
25. Yilmaz, O.: The effects of "live online courses" on students' achievement at distance education. Procedia Soc. Behav. Sci. **55**(5), 347–354 (2012)
26. Song, H., Kim, J., Luo, W.: Teacher-student relationship in online classes: a role of teacher self-disclosure. Comput. Hum. Behav. **54**, 436–443 (2016)
27. Shukor, N.A., Tasir, Z., der Meijden, H.V.: An examination of online learning effectiveness using data mining. Procedia Soc. Behav. Sci. **172**, 555–562 (2015)
28. Cho, Y.H., Choi, H., Shin, J., Yu, H.C., Kim, Y.K., Kim, J.Y.: Review of research on online learning environments in higher education. Procedia Soc. Behav. Sci. **191**(2), 2012–2017 (2015)
29. Beach, P.: Self-directed online learning: a theoretical model for understanding elementary teachers' online learning experiences. Teach. Teacher Educ. **61**, 60–72 (2017)

30. Rienties, B., Brouwer, N.Z., Lygo-Baker, S.: The effects of online professional development on higher education teachers' beliefs and intentions towards learning facilitation and technology. Teach. Teacher Educ. **29**(1), 122–133 (2013)
31. Karaman, S., Kucuk, S., Aydemir, M.: Evaluation of an online continuing education program from the perspective of new graduate nurses. Nurse Educ. Today **34**(5), 836–841 (2014)
32. Abdelaziz, M., Kamel, S.S., Karam, O., Abdelrahman, A.: Evaluation of e-learning program versus traditional lecture instruction for undergraduate nursing students in a faculty of nursing. Teach. Learn. Nurs. **6**(2), 50–58 (2011)
33. Pei, L., Wu, H.: Does online learning work better than offline learning in undergraduate medical education? A systematic review and meta-analysis. Med. Educ. Online **24**(1), 1–13 (2019)
34. Mubarak, A., Al-Arimi, A.K.: Distance learning. Procedia Soc. Behav. Sci. **152**, 82–88 (2014)
35. Tsiotakis, P., Jimoyiannis, A.: Critical factors towards analyzing teachers' presence in on-line learning communities. Internet High. Educ. **28**(1), 45–58 (2016)
36. van Rooij, S.W., Zirkle, K.: Balancing pedagogy, student readiness and accessibility: a case study in collaborative online course development. Internet High. Educ. **28**(1), 1–7 (2016)
37. Markova, T., Glazkova, I., Zaborova, E.: Quality issues of online distance learning. Procedia Soc. Behav. Sci. **237**, 685–691 (2017)
38. Cohen, A., Baruth, O.: Personality, learning, and satisfaction in fully online academic courses. Comput. Hum. Behav. **72**, 1–12 (2017)
39. Boling, E.C., Hough, M., Krinsky, H., Saleem, H., Stevens, M.: Cutting the distance in distance education: perspectives on what promotes positive, online learning experiences. Internet High. Educ. **15**(2), 118–126 (2012)
40. Moore, J.L., Dickson-Deane, C., Galyen, K.: E-learning, online learning, and distance learning environments: are they the same? Internet High. Educ. **14**(2), 129–135 (2011)
41. Sanders-Smith, S., Smith-Bonahue, T., Soutullo, O.: Practicing teachers' responses to case method of instruction in an online graduate course. Teach. Teacher Educ. **54**(2), 1–11 (2016)
42. Vo, H.M., Zhu, C., Diep, N.A.: The effect of blended learning on student performance at course-level in higher education: a meta-analysis. Stud. Educ. Eval. **53**, 17–28 (2017)
43. Ocak, M.A.: Why are faculty members not teaching blended courses? Insights from faculty members. Comput. Educ. **56**(3), 689–693 (2011)
44. Lust, G., Elen, J., Clarebout, G.: Regulation of tool-use within a blended course: Student differences and performance effects. Comput. Educ. **60**(1), 385–395 (2013)
45. Ellis, R.A., Pardo, A., Han, F.: Quality in blended learning environments - significant differences in how students approach learning collaborations. Comput. Educ. **102**, 90–102 (2016)
46. Boelens, R., De Wever, B., Voet, M.: Four key challenges to the design of blended learning: a systematic literature review. Educ. Res. Rev. **22**, 1–18 (2017)
47. Dziuban, C., Moskal, P.: A course is a course is a course: factor invariance in student evaluation of online, blended and face-to-face learning environments. Internet High. Educ. **14**(4), 236–241 (2011)
48. Nazarenko, A.L.: Blended learning vs traditional learning: what works? (a case study research). Procedia Soc. Behav. Sci. **200**, 77–82 (2015)
49. Slechtova P., Hana, V., Jan, V.: Blended learning: promising strategic alternative in higher education. Procedia Soc. Behav. Sci. **171**, 1245–1254 (2015)
50. Sophonhiranrak, S., Suwannatthachote, P., Ngudgratoke, S.: Factors affecting creative problem solving in the blended learning environment: a review of the literature. Procedia Soc. Behav. Sci. **174**, 2130–2136 (2015)
51. Havnes, A., Christiansen, B., Bjork, I.T., Hessevaagbakke, E.: Peer learning in higher education: patterns of talk and interaction in skills centre simulation. Learn. Cult. Soc. Interaction **8**, 75–87 (2016)
52. Brunello, G., De Paola, M., Scoppa, V.: Residential peer effects in higher education: does the field of study matter? Econ. Enquiry **48**(3), 621–634 (2010)

53. Sacerdote, B.: Peer effects in education: How might they work, how big are they, and how much do we know thus far? In: Handbook of the Economics of Education, vol. 3, pp. 249–277 (2010)
54. Kashefi, H., Ismail, Z., Yusof, Y.M.: Overcoming students obstacles in multivariable calculus through blended learning: a mathematical thinking approach. Procedia Soc. Behav. Sci. **56**, 579–586 (2012)
55. Savenye, W.C., Olina, Z., Niemczyk, M.: So you are going to be an online writing instructor: issues in designing, developing, and delivering an online course. Comput. Compos. **18**(4), 371–385 (2001)
56. Mo, Z.: Research on the teaching model application tendency of blended learning in the displayed design course. IERI Procedia **2**, 204–208 (2012)
57. McGee, P., Reis, A.: Blended course design: a synthesis of best practices. J. Asynchronous Learn. Netw. **16**(4), 7–22 (2012)
58. Vai, M., Sosulski, K.: Essentials of Online Course Design: A Standards-Based Guide (Essentials of Online Learning), 2nd edn. Routledge (2015)
59. Belenky, A.: Analyzing the potential of a firm: an operations research approach, Autom. Remote Control **35**(13), 1405–1424 (2002)
60. Hu, T.C., Kahng, A.B.: Linear and Integer Programming Made Easy, 1st edn. Springer (2016)
61. Belenky, A.: Two classes of games on polyhedral sets in systems economic studies. In: Network Models in Economics and Finance, pp. 35–80. Springer (2014)

SARAH-Based Variance-Reduced Algorithm for Stochastic Finite-Sum Cocoercive Variational Inequalities

Aleksandr Beznosikov and Alexander Gasnikov

1 Introduction

In this paper we focus on the following unconstrained variational inequality (VI) problem:

$$\text{Find } z^* \in \mathbb{R}^d \text{ such that } F(z^*) = 0, \tag{1}$$

where $F : \mathbb{R}^d \to \mathbb{R}^d$ is some operator. This formulation is broad and encompasses many popular classes of tasks arising in practice. The simplest, however, widely encountered example of the VI is the minimization problem:

$$\min_{z \in \mathbb{R}^d} f(z).$$

To represent it in the form (1), it is sufficient to take $F(z) = \nabla f(z)$. As another also popular practical example, we can consider a saddle point or min-max problem:

$$\min_{x \in \mathbb{R}^{d_x}} \max_{y \in \mathbb{R}^{d_y}} g(x, y).$$

Here we need to take $F(z) = [\nabla_x g(x, y), -\nabla_y g(x, y)]$.

A. Beznosikov
Moscow Institute of Physics and Technology, Dolgoprudny, Russia

HSE University, Moscow, Russia

A. Gasnikov (✉)
Moscow Institute of Physics and Technology, Dolgoprudny, Russia

IITP RAS, Moscow, Russia

Caucasus Mathematical Center, Adyghe State University, Maikop, Russia
e-mail: gasnikov@yandex.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_3

From a machine learning perspective, it is interesting not the deterministic formulation (1), but the stochastic one. More specifically, we want to consider the setup with the operator $F(z) = \mathbb{E}_{\xi\sim\mathcal{D}}\left[F_\xi(z)\right]$, where ξ is a random variable, $\mathcal{D}$ is some distribution, $F_\xi : \mathbb{R}^d \to \mathbb{R}^d$ is a stochastic operator. But it is often the case (especially in practical problems) that the distribution $\mathcal{D}$ is unknown, but we have some samples from $\mathcal{D}$. Then, one can replace with a finite-sum Monte Carlo approximation, i.e.

$$F(z) = \frac{1}{n}\sum_{i=1}^{n} F_i(z). \tag{2}$$

In the case of minimization problems, statements of the form (1) + (2) are also called empirical risk minimization [37]. These types of problems arise both in classical machine learning problems such as simple regressions and in complex, large-scale problems such as neural networks [23]. When it comes to saddle point problems, in recent times the so-called adversarial approach has become popular. Here one can highlight Generative Adversarial Networks (GANs) [13] and the adversarial training of models [25, 40].

Based on the examples mentioned above, it can be noted that for operators of the form (2), computing the full value of F is a possible but not desirable operation, since it is typically very expensive compared to computing a single operator F_i. Therefore, when constructing an algorithm for the problem (1) + (2), one wants to avoid computing (or compute very rarely) the full F operator. This task can be solved by a stochastic gradient descent (SGD) framework. Currently, stochastic methods for minimization problems already have a huge background [14]. The first methods of this type were proposed back in the 1950s by Robbins and Monro [35]. For example, in the most classic variant, SGD could be written as follows:

$$z^{k+1} = z^k - \eta v^k, \tag{3}$$

where $\eta > 0$ is a predefined step-size and $v^k = \nabla f_i(z^k)$, where $i \in [n]$ is chosen randomly [38]. In this case, the variance of v_t is the main source of slower convergence or convergence only to the neighbourhood of the solution [7, 21, 31].

But for minimization problems of the finite-sum type, one can achieve stronger theoretical and practical results compared to the method (3). This requires the use of a variance reduction technique. Recently, many variance-reduced variants of SGD have been proposed, including SAG/SAGA [10, 34, 36], SVRG [3, 19, 39], MISO [27], SARAH [18, 29, 30, 32], SPIDER [11], STORM [9], PAGE [24]. The essence of one of the earliest and best known variance-reduced methods SVRG is to use $v^k = \nabla f_i(z^k) - \nabla f_i(\tilde{z}) + \nabla f(\tilde{z})$, where $i \in [n]$ is picked at random, where $i \in [n]$ is picked at random and the point $\tilde{z}$ is updated very rarely (hence we do not need to compute the full gradient often). With this type of methods it is possible to achieve a linear convergence to the solution. But for both convex and non-convex smooth minimization problems, the best theoretical guarantees of convergence are given by other variance-reduced technique SARAH (and its modifications: SPIDER, STORM, PAGE).

In turn, stochastic methods are also investigated for variational inequalities and saddle point problems [4–6, 12, 15–17, 20, 28], including methods based on variance reduction techniques [1, 2, 5, 6, 8, 22, 33]. Most of these methods are based on the SVRG approach. At the same time, SARAH-based methods have not been explored for VIs. But as we noted earlier, these methods are the most attractive from the theoretical point of view for minimization problems. The purpose of this paper is to partially close the question of SARAH approach for stochastic finite-sum variational inequalities.

2 Problem Setup and Assumptions

Notation We use $\langle x, y\rangle := \sum_{i=1}^{n} x_i y_i$ to denote standard inner product of $x, y \in \mathbb{R}^d$ where x_i corresponds to the i-th component of x in the standard basis in $\mathbb{R}^d$. It induces ℓ_2-norm in $\mathbb{R}^d$ in the following way $\|x\|_2 := \sqrt{\langle x, x\rangle}$.

Recall that we consider the problem (1), where the operator F has the form (2). Additionally, we assume

Assumption 1 (Cocoercivity) *Each operator F_i is ℓ-cocoercive, i.e. for all $u, v \in \mathbb{R}^d$ we have*

$$\|F_i(u) - F_i(v)\|^2 \le \ell\langle F_i(u) - F_i(v), u - v\rangle. \tag{4}$$

This assumption is somehow a more restricted analogue of the Lipschitzness of F_i. For convex minimization problems, ℓ-Lipschitzness and ℓ-cocoercivity are equivalent. Regarding variational inequalities and saddle point problems, see [26].

Assumption 2 (Strong Monotonicity) *The operator F is μ-strongly monotone, i.e. for all $u, v \in \mathbb{R}^d$ we have*

$$\langle F(u) - F(v); u - v\rangle \ge \mu\|u - v\|^2. \tag{5}$$

For minimization problems this property means strong convexity, and for saddle point problems strong convexity–strong concavity.

3 Main Part

For general Lipschitzness variational inequalities, stochastic methods are usually based not on SGD, but on the Stochastic Extra Gradient method [20]. But due to the fact that we consider cocoercive VIs, it is sufficient to look at SGD like methods for this class of problems. For example, [26] considers SGD, [6]—SVRG. Following this reasoning, we base our method on the original SARAH [29].

Algorithm 1 SARAH [29] for stochastic cocoercive variational inequalities

```
Parameters: Stepsize $\gamma > 0$, number of iterations $K$, $S$.
Initialization: Choose $\tilde{z}^0 \in \mathbb{R}^d$.
for $s = 1, 2, \dots, S$ do
    $z^0 = \tilde{z}^{s-1}$
    $v^0 = F(z^0)$
    $z^1 = z^0 - \gamma v^0$
    for $k = 1, 2, \dots, K-1$ do
        Sample $i_k$ independently and uniformly from $[n]$
        $v^k = F_{i_k}(z^k) - F_{i_k}(z^{k-1}) + v^{k-1}$
        $z^{k+1} = z^k - \gamma v^k$
    end for
    $\tilde{z}^s = z^K$
end for
```

Next, we analyse the convergence of this method. Note that we will use the vector v^K in the analysis, but in reality this vector is not calculated by the algorithm. Our proof are heavily based on the original work on SARAH [29]. Lemma 1 gives an understanding of how $\|v^k\|^2$ behaves during the internal loop of Algorithm 1.

Lemma 1 *Suppose that Assumptions 1 and 2 hold. Consider SARAH (Algorithm 1) with $\gamma \leq \frac{1}{\ell}$. Then, we have*

$$\mathbb{E}[\|v^K\|^2] \leq (1 - \gamma\mu)^K \mathbb{E}[\|F(z^0)\|^2].$$

Proof We start the proof with an update for v^k:

$$\|v^k\|^2 = \|v^{k-1}\|^2 + \|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2 + 2\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), v^{k-1}\rangle.$$

Next, we use an update for z^k and make a small rearrangement

$$\begin{aligned}\|v^k\|^2 =& \|v^{k-1}\|^2 + \|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2 - \frac{2}{\gamma}\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), z^k - z^{k-1}\rangle \\ =& \|v^{k-1}\|^2 + \|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2 - \frac{1}{\gamma}\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), z^k - z^{k-1}\rangle \\ & - \frac{1}{\gamma}\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), z^k - z^{k-1}\rangle.\end{aligned}$$

Taking the full mathematical expectation, we obtain

$$\begin{aligned}\mathbb{E}[\|v^k\|^2] =&\mathbb{E}[\|v^{k-1}\|^2] + \mathbb{E}[\|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2]\\ &- \frac{1}{\gamma}\mathbb{E}[\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), z^k - z^{k-1}\rangle]\\ &- \frac{1}{\gamma}\mathbb{E}[\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), z^k - z^{k-1}\rangle].\end{aligned}$$

Independence of the i_k generation gives

$$\begin{aligned}\mathbb{E}[\|v^k\|^2] =&\mathbb{E}[\|v^{k-1}\|^2] + \mathbb{E}[\|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2]\\ &- \frac{1}{\gamma}\mathbb{E}[\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), z^k - z^{k-1}\rangle]\\ &- \frac{1}{\gamma}\mathbb{E}[\langle \mathbb{E}_{i_k}[F_{i_k}(z^k) - F_{i_k}(z^{k-1})], z^k - z^{k-1}\rangle]\\ =&\mathbb{E}[\|v^{k-1}\|^2] + \mathbb{E}[\|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2]\\ &- \frac{1}{\gamma}\mathbb{E}[\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), z^k - z^{k-1}\rangle]\\ &- \frac{1}{\gamma}\mathbb{E}[\langle F(z^k) - F(z^{k-1}), z^k - z^{k-1}\rangle].\end{aligned}$$

With Assumptions 1 and 2, we get

$$\begin{aligned}\mathbb{E}[\|v^k\|^2] \leq&\mathbb{E}[\|v^{k-1}\|^2] + \mathbb{E}[\|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2]\\ &- \frac{1}{\gamma\ell}\mathbb{E}[\|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2]\\ &- \frac{\mu}{\gamma}\mathbb{E}[\|z^k - z^{k-1}\|^2]\\ =&(1-\gamma\mu)\mathbb{E}[\|v^{k-1}\|^2] + \left(\frac{\gamma\ell - 1}{\gamma\ell}\right)\mathbb{E}[\|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2].\end{aligned}$$

In the last step we substitute $z^{k-1} - z^k = \gamma v^k$. The choice of $0 < \gamma \leq \frac{1}{\ell}$ gives

$$\mathbb{E}[\|v^k\|^2] \leq (1-\gamma\mu)\mathbb{E}[\|v^{k-1}\|^2].$$

Running recursion and using $v^0 = F(z^0)$, we finish the proof. □

The following lemma gives how different v^K and $F(z^K)$ are in the inner loop of Algorithm 1.

Lemma 2 *Suppose that Assumption 1 holds. Consider SARAH (Algorithm 1). Then, we have*

$$\mathbb{E}[\|F(z^K) - v^K\|^2] \leq \frac{\gamma\ell}{2-\gamma\ell}\mathbb{E}[\|F(z^0)\|^2].$$

Proof Let us consider the following chain of reasoning:

$$\begin{aligned}
\mathbb{E}[\|F(z^k) - v^k\|^2] =& \mathbb{E}[\|[F(z^{k-1}) - v^{k-1}] + [F(z^k) - F(z^{k-1})] - [v^k - v^{k-1}]\|^2] \\
=& \mathbb{E}[\|F(z^{k-1}) - v^{k-1}\|^2] + \mathbb{E}[\|F(z^k) - F(z^{k-1})\|^2] \\
&+ \mathbb{E}[\|v^k - v^{k-1}\|^2] \\
&+ 2\mathbb{E}[\langle F(z^{k-1}) - v^{k-1}, F(z^k) - F(z^{k-1})\rangle] \\
&- 2\mathbb{E}[\langle F(z^{k-1}) - v^{k-1}, v^k - v^{k-1}\rangle] \\
&- 2\mathbb{E}[\langle F(z^k) - F(z^{k-1}), v^k - v^{k-1}\rangle] \\
=& \mathbb{E}[\|F(z^{k-1}) - v^{k-1}\|^2] + \mathbb{E}[\|F(z^k) - F(z^{k-1})\|^2] \\
&+ \mathbb{E}[\|v^k - v^{k-1}\|^2] \\
&+ 2\mathbb{E}[\langle F(z^{k-1}) - v^{k-1}, F(z^k) - F(z^{k-1})\rangle] \\
&- 2\mathbb{E}[\langle F(z^{k-1}) - v^{k-1}, \mathbb{E}_{i_k}[v^k - v^{k-1}]\rangle] \\
&- 2\mathbb{E}[\langle F(z^k) - F(z^{k-1}), \mathbb{E}_{i_k}[v^k - v^{k-1}]\rangle] \\
=& \mathbb{E}[\|F(z^{k-1}) - v^{k-1}\|^2] - \mathbb{E}[\|F(z^k) - F(z^{k-1})\|^2] \\
&+ \mathbb{E}[\|v^k - v^{k-1}\|^2] \\
\leq& \mathbb{E}[\|F(z^{k-1}) - v^{k-1}\|^2] + \mathbb{E}[\|v^k - v^{k-1}\|^2].
\end{aligned}$$

Here we also use that

$$\mathbb{E}_{i_k}[v^k - v^{k-1}] = \mathbb{E}_{i_k}[F_{i_k}(z^k) - F_{i_k}(z^{k-1})] = F(z^k) - F(z^{k-1}).$$

Running recursion and using $v^0 = F(z^0)$, we have

$$\mathbb{E}[\|F(z^K) - v^K\|^2] \leq \sum_{k=1}^{K} \mathbb{E}[\|v^k - v^{k-1}\|^2]. \tag{6}$$

In the same way as in Lemma 1, we can derive

$$\begin{aligned}
\|v^k\|^2 =& \|v^{k-1}\|^2 + \|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2 + 2\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), v^{k-1}\rangle \\
=& \|v^{k-1}\|^2 + \|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2 - \frac{2}{\gamma}\langle F_{i_k}(z^k) - F_{i_k}(z^{k-1}), z^k - z^{k-1}\rangle
\end{aligned}$$

$$\begin{aligned}
\le &\|v^{k-1}\|^2 + \|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2 - \frac{2}{\gamma\ell}\|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2 \\
= &\|v^{k-1}\|^2 + \left(\frac{\gamma\ell - 2}{\gamma\ell}\right)\|F_{i_k}(z^k) - F_{i_k}(z^{k-1})\|^2 \\
= &\|v^{k-1}\|^2 + \left(\frac{\gamma\ell - 2}{\gamma\ell}\right)\|v^k - v^{k-1}\|^2.
\end{aligned}$$

After a small rewriting and with the full expectation, we get

$$\mathbb{E}[\|v^k - v^{k-1}\|^2] \le \frac{\gamma\ell}{2-\gamma\ell}\mathbb{E}[\|v^{k-1}\|^2 - \|v^k\|^2].$$

By substituting this into the expression (6) and using $v^0 = F(z^0)$, we finish the proof. □

Let us combine Lemmas 1 and 2 into the main theorem of this paper.

Theorem 1 *Suppose that Assumptions 1 and 2 hold. Consider SARAH (Algorithm 1) with $\gamma = \frac{2}{9\ell}$ and $K = \frac{10\ell}{\mu}$. Then, we have*

$$\mathbb{E}[\|F(\tilde{z}^s)\|^2] \le \frac{1}{2}\mathbb{E}[\|F(\tilde{z}^{s-1})\|^2].$$

Proof We start from

$$\mathbb{E}[\|F(z^K)\|^2] \le 2\mathbb{E}[\|F(z^K) - v^K\|^2] + 2[\mathbb{E}\|v^K\|^2].$$

Applying Lemma 1 and 2, we have

$$\begin{aligned}
\mathbb{E}[\|F(z^K)\|^2] &\le \left[\frac{2\gamma\ell}{2-\gamma\ell} + 2(1-\gamma\mu)^K\right]\mathbb{E}[\|F(z^0)\|^2] \\
&\le \left[\frac{2\gamma\ell}{2-\gamma\ell} + 2\exp(-\gamma\mu K)\right]\mathbb{E}[\|F(z^0)\|^2].
\end{aligned}$$

Here we also use that $\gamma\mu \in (0; 1)$ (for $\gamma \le \frac{2}{9\ell}$) and then $(1-\gamma\mu) \le \exp(-\gamma\mu)$. The substitution γ and K gives

$$\mathbb{E}[\|F(z^K)\|^2] \le \frac{1}{2}\mathbb{E}[\|F(z^0)\|^2].$$

We know that $z^0 = \tilde{z}^{s-1}$ and $z^K = \tilde{z}^s$ and have

$$\mathbb{E}[\|F(\tilde{z}^s)\|^2] \le \frac{1}{2}\mathbb{E}[\|F(\tilde{z}^{s-1})\|^2].$$

□

Since we need to find a point z such that $F(z) \approx F(z^*) = 0$, we can easily get an estimate on the oracle complexity (number of F_i calls) to achieve precision ε.

Corollary 1 *Suppose that Assumptions 1 and 2 hold. Consider SARAH (Algorithm 1) with $\gamma = \frac{2}{9\ell}$ and $K = \frac{10\ell}{\mu}$. Then, to achieve ε-solution ($\mathbb{E}\|F(\tilde{z}^S)\|^2 \sim \varepsilon^2$), we need*

$$\mathcal{O}\left(\left[n + \frac{\ell}{\mu}\right] \log_2 \frac{\|F(z^0)\|^2}{\varepsilon^2}\right) \quad \textit{oracle calls.}$$

Proof From Theorem 1 we need the following number of outer iterations:

$$S = \mathcal{O}\left(\log_2 \frac{\|F(z^0)\|^2}{\varepsilon^2}\right).$$

At each outer iteration we compute the full operator one time, and at the remaining $K - 1$ iterations we call the single operator F_i two times per one inner iteration. Then, the total number of oracle calls is

$$S \times (2 \times (K - 1) + n) = \mathcal{O}\left(\left[n + \frac{\ell}{\mu}\right] \log_2 \frac{\|F(z^0)\|^2}{\varepsilon^2}\right).$$

□

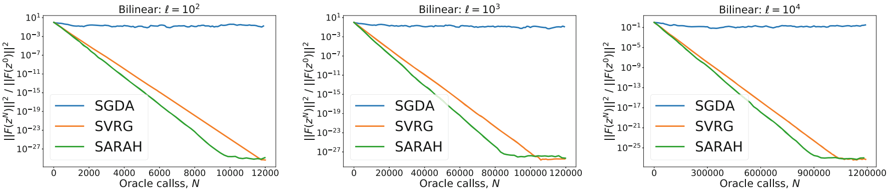
Note that the obtained oracle complexity coincides with the similar complexity for SVRG from [6]. It is interesting to see how these methods behave in practice.

4 Experiments

The aim of our experiments is to compare the performance of different methods for stochastic finite-sum cocoercive variational inequalities. In particular, we use SGD from [26], SVRG from [6] and SARAH. We conduct our experiments on a finite-sum bilinear saddle point problem:

$$g(x, y) = \frac{1}{n}\sum_{i=1}^{n}\left[g_i(x, y) = x^\top A_i y + a_i^\top x + b_i^\top y + \frac{\lambda}{2}\|x\|^2 - \frac{\lambda}{2}\|y\|^2\right], \qquad (7)$$

where $A_i \in \mathbb{R}^{d\times d}$, $a_i, b_i \in \mathbb{R}^d$. This problem is λ-strongly convex–strongly concave and, moreover, L-smooth with $L = \|A\|_2$ for $A = \frac{1}{n}\sum_{i=1}^n A_i$. We take $n = 10$, $d = 100$ and generate matrix A and vectors a_i, b_i randomly, $\lambda = 1$. For this problem the cocoercivity constant $\ell = \frac{\|A\|_2^2}{\lambda}$. The steps of the methods are selected for best convergence. For SVRG and SARAH the number of iterations for the inner loops is



(a) (b) (c)

Fig. 1 Bilinear problem (7): Comparison of state-of-the-art SGD-based methods for stochastic cocoercive VIs. (**a**) Small ℓ. (**b**) Medium ℓ. (**c**) Large ℓ

taken as $\frac{\ell}{\lambda}$. We run three experiment setups: with small $\ell \approx 10^2$, medium $\ell \approx 10^3$ and big $\ell \approx 10^4$.

See Fig. 1 for the results. We see that SARAH converges better than SVRG, and SGD converges much slower.

Acknowledgments The authors would like to congratulate Boris Mirkin on his jubilee and wish him good health and new scientific advances.

The work of A. Beznosikov was supported by the strategic academic leadership program 'Priority 2030' (Agreement 075-02-2021-1316 30.09.2021). The work of A. Gasnikov was supported by the Ministry of Science and Higher Education of the Russian Federation (Goszadaniye), No. 075-00337-20-03, project No. 0714-2020-0005.

References

1. Alacaoglu, A., Malitsky, Y.: Stochastic variance reduction for variational inequality methods. Preprint. arXiv:2102.08352 (2021)
2. Alacaoglu, A., Malitsky, Y., Cevher, V.: Forward-reflected-backward method with variance reduction. Comput. Optim. Appl. **80** (2021). https://doi.org/10.1007/s10589-021-00305-3
3. Allen-Zhu, Z., Yuan, Y.: Improved SVRG for non-strongly-convex or sum-of-non-convex objectives. In: International Conference on Machine Learning, pp. 1080–1089. PMLR (2016)
4. Beznosikov, A., Samokhin, V., Gasnikov, A.: Distributed saddle-point problems: lower bounds, optimal and robust algorithms. Preprint. arXiv:2010.13112 (2020)
5. Beznosikov, A., Gasnikov, A., Zainulina, K., Maslovskiy, A., Pasechnyuk, D.: A unified analysis of variational inequality methods: variance reduction, sampling, quantization and coordinate descent. Preprint. arXiv:2201.12206 (2022)
6. Beznosikov, A., Gorbunov, E., Berard, H., Loizou, N.: Stochastic gradient descent-ascent: unified theory and new efficient methods. Preprint. arXiv:2202.07262 (2022)
7. Bottou, L., Curtis, F.E., Nocedal, J.: Optimization methods for large-scale machine learning. Siam Rev. **60**(2), 223–311 (2018)
8. Chavdarova, T., Gidel, G., Fleuret, F., Lacoste-Julien, S.: Reducing noise in gan training with variance reduced extragradient. **32**, 393–403 (2019)
9. Cutkosky, A., Orabona, F.: Momentum-based variance reduction in non-convex SGD. Preprint. arXiv:1905.10018 (2019)
10. Defazio, A., Bach, F., Lacoste-Julien, S.: Saga: a fast incremental gradient method with support for non-strongly convex composite objectives. In: Advances in Neural Information Processing Systems, pp. 1646–1654 (2014)

11. Fang, C., Li, C.J., Lin, Z., Zhang, T.: Spider: near-optimal non-convex optimization via stochastic path integrated differential estimator. Preprint. arXiv:1807.01695 (2018)
12. Gidel, G., Berard, H., Vignoud, G., Vincent, P., Lacoste-Julien, S.: A variational inequality perspective on generative adversarial networks. Preprint. arXiv:1802.10551 (2018)
13. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Commun. ACM **63**(11), 139–144 (2020)
14. Gorbunov, E., Hanzely, F., Richtárik, P.: A unified theory of SGD: Variance reduction, sampling, quantization and coordinate descent. In: International Conference on Artificial Intelligence and Statistics, pp. 680–690. PMLR (2020)
15. Gorbunov, E., Berard, H., Gidel, G., Loizou, N.: Stochastic extragradient: general analysis and improved rates. In: International Conference on Artificial Intelligence and Statistics, pp. 7865–7901. PMLR (2022)
16. Hsieh, Y.G., Iutzeler, F., Malick, J., Mertikopoulos, P.: On the convergence of single-call stochastic extra-gradient methods. In: Advances in Neural Information Processing Systems, vol. 32, pp. 6938–6948. Curran Associates, Inc. (2019)
17. Hsieh, Y.G., Iutzeler, F., Malick, J., Mertikopoulos, P.: Explore aggressively, update conservatively: Stochastic extragradient methods with variable stepsize scaling. 33, 16223–16234 (2020)
18. Hu, W., Li, C.J., Lian, X., Liu, J., Yuan, H.: Efficient smooth non-convex stochastic compositional optimization via stochastic recursive gradient descent. In: Advances in Neural Information Processing Systems. vol. 32, pp. 6929–6937. Curran Associates, Inc. (2019)
19. Johnson, R., Zhang, T.: Accelerating stochastic gradient descent using predictive variance reduction. In: Advances in Neural Information Processing Systems, vol. 26, pp. 315–323 (2013)
20. Juditsky, A., Nemirovski, A., Tauvel, C.: Solving variational inequalities with stochastic mirror-prox algorithm. Stoch. Syst. **1**(1), 17–58 (2011)
21. Khaled, A., Richtárik, P.: Better theory for SGD in the nonconvex world. Preprint. arXiv:2002.03329 (2020)
22. Kovalev, D., Beznosikov, A., Borodich, E., Gasnikov, A., Scutari, G.: Optimal gradient sliding and its application to distributed optimization under similarity. Preprint. arXiv:2205.15136 (2022)
23. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
24. Li, Z., Bao, H., Zhang, X., Richtárik, P.: Page: a simple and optimal probabilistic gradient estimator for nonconvex optimization. In: International Conference on Machine Learning, pp. 6286–6295. PMLR (2021)
25. Liu, X., Cheng, H., He, P., Chen, W., Wang, Y., Poon, H., Gao, J.: Adversarial training for large neural language models. Preprint. arXiv:2004.08994 (2020)
26. Loizou, N., Berard, H., Gidel, G., Mitliagkas, I., Lacoste-Julien, S.: Stochastic gradient descent-ascent and consensus optimization for smooth games: convergence analysis under expected co-coercivity. In: Advances in Neural Information Processing Systems, vol. 34, pp. 19095–19108 (2021)
27. Mairal, J.: Incremental majorization-minimization optimization with application to large-scale machine learning. SIAM J. Optim. **25**(2), 829–855 (2015)
28. Mishchenko, K., Kovalev, D., Shulgin, E., Richtárik, P., Malitsky, Y.: Revisiting stochastic extragradient. In: International Conference on Artificial Intelligence and Statistics, pp. 4573–4582. PMLR (2020)
29. Nguyen, L.M., Liu, J., Scheinberg, K., Takáč, M.: SARAH: a novel method for machine learning problems using stochastic recursive gradient. In: International Conference on Machine Learning, pp. 2613–2621. PMLR (2017)
30. Nguyen, L.M., Liu, J., Scheinberg, K., Takáč, M.: Stochastic recursive gradient algorithm for nonconvex optimization. Preprint. arXiv:1705.07261 (2017)
31. Nguyen, L.M., Nguyen, P.H., Richtárik, P., Scheinberg, K., Takác, M., van Dijk, M.: New convergence aspects of stochastic gradient algorithms. J. Mach. Learn. Res. 20(176), 1–49 (2019), http://jmlr.org/papers/v20/18-759.html

32. Nguyen, L.M., Scheinberg, K., Takáč, M.: Inexact SARAH algorithm for stochastic optimization. Optim. Methods Softw. **36**(1), 237–258 (2021)
33. Palaniappan, B., Bach, F.: Stochastic variance reduction methods for saddle-point problems. In: Advances in Neural Information Processing Systems, pp. 1416–1424 (2016)
34. Qian, X., Qu, Z., Richtárik, P.: Saga with arbitrary sampling. In: International Conference on Machine Learning, pp. 5190–5199. PMLR (2019)
35. Robbins, H., Monro, S.: A stochastic approximation method. Ann. Math. Stat. **22**(3), 400–407 (1951)
36. Schmidt, M., Le Roux, N., Bach, F.: Minimizing finite sums with the stochastic average gradient. Math. Program. **162**(1–2), 83–112 (2017)
37. Shalev-Shwartz, S., Ben-David, S.: Understanding machine learning: from theory to algorithms. Cambridge University Press (2014)
38. Shalev-Shwartz, S., Singer, Y., Srebro, N., Cotter, A.: Pegasos: primal estimated sub-gradient solver for SVM. Math. Program. **127**(1), 3–30 (2011)
39. Yang, Z., Chen, Z., Wang, C.: Accelerating mini-batch SARAH by step size rules. Inf. Sci. **558**, 157–173 (2021)
40. Zhu, C., Cheng, Y., Gan, Z., Sun, S., Goldstein, T., Liu, J.: Freelb: enhanced adversarial training for natural language understanding. Preprint. arXiv:1909.11764 (2019)

Dimensionality Reduction Using Pseudo-Boolean Polynomials for Cluster Analysis

Tendai Mapungwana Chikake and Boris Goldengorin

1 Introduction

In the fields of data visualization and cluster analysis, dimensionality reduction mechanisms play pivotal roles in providing better understanding of relations and clusters in input data. Dimensionality reduction techniques work by transforming data from high-dimensional spaces into low-dimensional spaces such that the low-dimensional representations retain meaningful properties of the original data, ideally close to their intrinsic dimensions [1].

Real-world data is often available in high-dimensional spaces which are usually cognitively and computationally hard to process [2]. Human observers as well as presentation mediums readily available, like 2-dimensional papers or screens, are presented with representational challenges whenever data is available in higher than 3-dimensional spaces and consequently the identity of classes, useful or noisy features becomes harder to discover [2].

Data scientists often spend enormous amounts of time and effort digging for relevant features that determine classes or those features that bring useless noise in

T. M. Chikake (✉)
Department of Discrete Mathematics, Moscow Institute of Physics and Technology, Moscow, Russian Federation
e-mail: tendaichikake@phystech.edu

B. Goldengorin
Department of Mathematics, New Uzbekistan University, Tashkent, Uzbekistan

The Scientific and Educational Mathematical Center «Sofia Kovalevskaya Northwestern Center for Mathematical Research», Pskov State University, Pskov, Russia

Department of Discrete Mathematics, Phystech School of Applied Mathematics and Informatics, Moscow Institute of Physics and Technology, Dolgoprudny, Moscow region, Russia
e-mail: b.goldengorin@newuu.uz; goldengorin.bi@mipt.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_4

datasets. Automated systems like artificial neural networks can be used in the feature selection processes [3], but often require large amounts of data and challenges like feature superposition [4], underfitting, overfitting, and interpretation concerns arise [5].

In fields where large numbers of observations and/or large numbers of variables exist such as signal processing, computer vision, speech recognition, neuroinformatics, and bioinformatics, usage of dimension reduction techniques is crucial [6]. Dimensionality reduction simplifies cluster analysis tasks for both human and machine processors [6].

In this work, we observe that our powerful dimensionality reduction method can assist in reducing the abuse of statistical methods and/or artificial neural networks in tasks that can be solved in combinatorial steps. We show this by qualifying our dimensionality reduction method, followed by linear clustering of reduced samples. This advantage is available because our reduction method has invariability properties, and it is intuitively easy to interpret. The problems of invariability and interpretability are among the top problems of currently available dimensionality reduction methods [7]. Dimensionality reduction tools that lack interpretability and/or invariability can be disfavoured in critical tasks such as clustering models that input medical tests/measurements to predict a diagnosis. Our work is directed to solving such challenges.

Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other clusters [8]. Clustering is a major task of exploratory data analysis, and a common technique for statistical data analysis, used in many fields, including pattern recognition, image analysis, information retrieval, bioinformatics, data compression, computer graphics and machine learning [8].

Abuse of statistical methods and/or artificial neural networks often arise in cases where data is minimal. This abuse may result in overfitted, irreproducible or hard-to-interpret solutions which may be undesirable to use in some critical tasks.

The invariant manipulation based on formulation of pseudo-Boolean polynomials presented in this work, can enable cluster analysts to extract simple rules of associations in data without the need for machine learning. The formulation of pseudo-Boolean polynomials is very simple, computationally efficient, invariant to ordering, and easy to explain and reproduce.

Our overall contributions in this paper are:

1. Qualifying the usage of the reduction property of penalty-based pseudo-Boolean polynomials formulation for dimensionality reduction of multidimensional data where it is feasible.
2. Reducing overdependence on data-driven approaches in solving problems that can be solved with combinatorial steps.

Dimensionality reduction using pseudo-Boolean polynomials formulation, revolves around the manipulation of the *reduction* and *equivalence* properties of penalty-based pseudo-Boolean polynomials [9].

We present our results on classical Wisconsin Diagnostic Breast Cancer (WDBC) [10] and Iris Flower datasets [11], which have too few samples, such that the usage of a data-driven methods like artificial neural networks for clustering would result in abuse.

The Iris Flower dataset [11] has samples of size 1×4 and present challenges of identifying clusters by plotting on a Cartesian plane for the analyst while the Wisconsin Diagnostic Breast Cancer (WDBC) [10] dataset, has samples of size 1×30 that result in 30-Dimensionality representation that would be incomprehensible for Cartesian plot based analysis.

Our proposed method is limited to data whose samples can be represented as cost matrices where each cell represents a cost relationship of its respective column and row. In the experiment sections, we show how these complex datasets can be reduced to 2×1 and 3×1 dimensionality which are easily analysed on a Cartesian plane and Cartesian space respectively. Simple linear demarcations are then used to qualify the label of a given sample without any machine learning process or non-linear alteration of data.

2 Related Work

Principal Component Analysis (PCA) and the T-distribute Stochastic Neighbour Embedding (t-SNE) are arguably the most popular dimensionality reduction methods in cluster analysis tasks. The choice of usage of either, is usually case based. The T-distribute Stochastic Neighbour Embedding is often used for visualizing high dimensional data. It works by converting similarities between data points to joint probabilities and tries to minimize the Kullback-Leibler divergence between the joint probabilities of the low-dimensional embedding and the high-dimensional data [12]. T-SNE has a cost function that is not convex, i.e., with different initializations different reductions may result [12]. The non-convex nature of the cost function in the t-SNE tool is a major drawback in comparison with the Principal Component Analysis method as well as the pseudo-Boolean polynomials-based reduction that we qualify in this paper.

Dimensionality reduction by pseudo-Boolean polynomials formulation ensures unique reduced Hammer-Beresnev polynomials, regardless of possible difference in ordering of input matrices [9]. Consequently, our method is guaranteed to output the same reductions, regardless of difference in initializations or input orderings.

Principal Component Analysis (PCA), is an orthogonal linear transformation that transforms data to a new coordinate system such that the greatest variance by some scalar projection of the data comes to lie on the first coordinate (called the first principal component), the second-greatest variance on the second coordinate, and so on [13]. The major drawback of PCA is that it changes the distances involved in our data because it reduces dimensions in a way that preserves large pairwise distance better than small pairwise distance [14]. These changes can be very sensitive to our algorithms, especially when working with Euclidean distance-based algorithms.

Modifications to the Principal Component Analysis (PCA) method exists like the kernel principal component analysis (kernel PCA) [1] and Graph-based kernel PCA [15], that can be employed in a nonlinear manner but the distance and data dependence concerns prevail.

Other techniques include Linear discriminant analysis (LDA) [16], Generalized discriminant analysis (GDA) [17], and Uniform manifold approximation and projection (UMAP) [18]. All of these techniques, raise all or some of the concerns outlined above as they are largely dependent on data distribution to operate.

The pseudo-Boolean polynomials approach presented in this work, operates in isolation for each individual sample, such that no distribution biases are introduced to any given sample. The combinatorial operations are constant and invariant to ordering across all samples.

3 Methods

In mathematics and optimization, a pseudo-Boolean function is a function of the form $f : \mathbf{B^n} \rightarrow \mathbb{R}$, where $\mathbf{B} = \{0, 1\}$ is a Boolean domain and n is a non-negative integer called the degree of the function [19].

We utilise a penalty based formulation of pseudo-Boolean polynomials described in [20] for data aggregation which provides us an invariant dimensionality reduction property that we present in this work.

Goldengorin et al. [9] highlights fundamental reduction properties of pseudo-Boolean polynomials which guarantee the maintenance of the underlying initial information while reducing the size of the problem.

We extend the objective goal in [20] by describing our problem as a problem of minimising the cost of describing a given sample and thereby reducing the dimensionality of the particular sample.

Given a sample where observables (e.g. physical quantities, types of measurements, etc.) $I = \{1, 2, ..., m\}$ (ordered by their correlation strengths to labels), their physical measurements $J = \{1, 2, ..., n\}$, and p the maximum dimensionality size desired, to minimise the cost of describing the given sample, our task is to find a set $S \subseteq I$ with $|S| = p$ minimising the prespecified objective function.

We define the instance of the problem by an $m \times n$ matrix $C = [c_{ij}]$ of costs (distances, bandwidth, time, (dis)similarities, (in)significance, etc.), $j \in J$ and $i \in I$, and the goal is to find a set $S \subseteq I$ with $|S| = p$, such that we minimise total cost

$$f_c(S) = \sum_{j \in J} min\{\mathrm{c_{ij}} | i \in S\}, \tag{1}$$

with the assumption that entries of C are non-negative and finite [20].

According to AlBdaiwi et al. [20], the objective function $f_c(S)$ of this kind of problem can be formulated in terms of pseudo-Boolean polynomials and from

[19] all pseudo-Boolean polynomials can be uniquely represented as multilinear polynomials of the form

$$f(\mathbf{y}) = \sum_{S \subseteq I} \mathrm{c_S} \prod_{i \in S} \mathrm{y_i} \tag{2}$$

Pseudo-Boolean polynomials formulation is achievable in polynomial time and allows us to achieve compact representations of relatively large problems [20].

From (2), $\prod_{i \in S} y_i$ is the term of the monomial $c_S \prod_{i \in S} y_i$. Monomials with the same term are called similar monomials [9], and they can be added together in a process called *reduction* [9] which is central to our dimensionality reduction solution as it allows us to reduce the number of columns in the initial cost matrix.

In addition to compacting large problems, there exists different instances that have similar (reduced) Hammer–Beresnev polynomials, mainly because similar monomials can be aggregated and disaggregated [9].

These properties are essential in cluster analysis because samples which might look dissimilar in higher dimensional space, can actually converge to similarity in their reduced pseudo-Boolean polynomials form.

This work seeks to exploit these fundamental properties: representational reduction and *equivalence* as dimensionality reduction and clustering mechanisms respectively.

By treating measurable attributes of multidimensional data, like physical measurements, pixel positioning and intensity distribution in image data, and other describable/measurable attributes as *information costs* of describing samples, we can formulate for each sample, a cost matrix C which we can manipulate and reduce, in an ordering invariant manner, by pseudo-Boolean polynomials formulation and achieve lower dimension representation of each sample independent of any other samples in the dataset.

The *equivalence* [9] property and other distance comparisons can then be applied on the reduced data representation for cluster analysis.

Visualizing a matrix sample of size $1 \times n$ where $n \in \{1, 2, 3\}$ is easily comprehensive by scatter plotting along 1-D, 2-D, and 3-D planes respectively. If classes are present in the data, we can identify linear or non-linear lines or plane separators that demarcate boundaries of clusters in the data.

The pseudo-Boolean approach to dimensionality reduction in measured features for sample clustering is a penalty-based approach that relies on the fact that we require attributes that positively distinguish underlying classes for each instance to be sufficiently represented based on their importance to the classifier.

The task seeks to minimize measurements that *insignificantly* contribute to the identity of a sample in a specific class.

A sample is described by an $m \times n$ matrix $C = [c_{i,j}]$ where I represents the measured feature while J represent the measurement such that columns of the matrix contain homogenous quantities. In reducing the dimensionality of samples in the Iris Flower dataset [11] for example, I represents measured features, *sepal, and petal* while J represent the measurements *width and length* thereof.

We define the decisive insignificance S of an attribute to classifying a sample into a specific cluster as

$$f_c(S) = \sum_{j \in J} min\{c_{ij} | i \in S\} \tag{3}$$

and the dimensionality reduction task is the problem of finding

$$S^* \in arg\,min\{f_c(S) : \emptyset \subset S \subseteq I, |S| = p\}, \tag{4}$$

where p is the output dimension size, which we however choose to be $|J|$ such that no measurement is lost.

By processing our samples into pseudo-Boolean polynomials we achieve the sample representations with the least possible information costs needed to represent them.

4 Experimental Setup

4.1 The Iris Flower Dataset

Table 1 shows the Iris Flower dataset [11], a classical and popular dataset in the machine learning community. The task on this dataset is to classify Iris plants into three species (*Iris setosa, Iris versicolor,* and *Iris virginica*) using the lengths and widths measurements of their petals and sepals. The dataset contains 150 samples.

Our method requires that data is structured as matrices, where columns are measurements and rows are the features measured. For this dimensionality reduction task, we first reshape the structure of all instances from 1×4 sized instances to 2×2 sized instances, where rows represent the measurement type *(Sepal, Petal)* and the columns represent the measurements *(Length and Width)* of the features collected from 3 different species:

Table 1 Iris dataset

ID	Sepal length (cm)	Sepal width (cm)	Petal length (cm)	Petal width (cm)	Target
1	5.4	3.0	4.5	1.5	Versicolor
2	7.7	2.8	6.7	2.0	Virginica
3	5.2	3.4	1.4	0.2	Setosa
4	4.8	3.4	1.9	0.2	Setosa
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
150	6.6	3.0	4.4	1.4	Versicolor

Table 2 Transformed instance

	Length (cm)	Width (cm)
Sepal	5.4	3.0
Petal	4.5	1.5

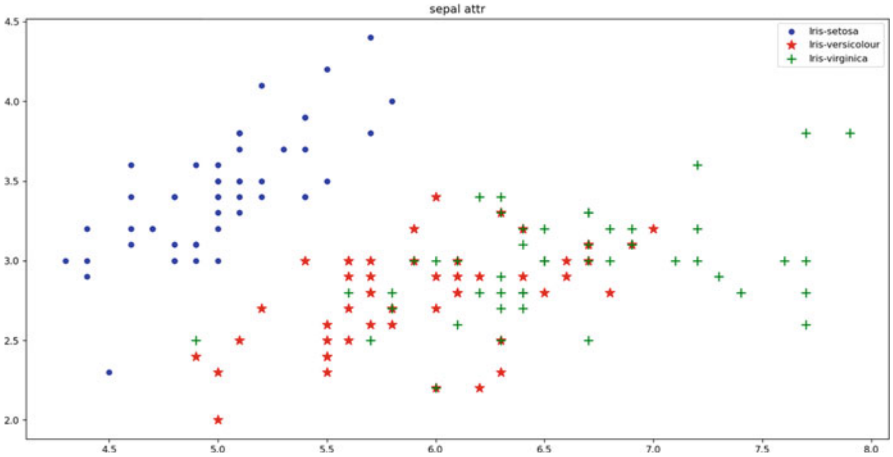


Fig. 1 Scatter on sepal length-width

We transform the samples into costs matrices by making rows of measured features (Sepal, Petal) and columns of measurements (Length, Width) resulting in 2×2 costs matrices as shown in Table 2.

One cluster in the Iris Flower dataset, *Iris-Setosa* is linearly separable from others while *Iris-virginica* and *Iris-versicolour* are not linearly separable between each other as shown in Figs. 1 and 2.

Figure 1 shows the scatter plot on lengths and widths of sepal measurements of the samples while Fig. 2 shows the scatter plot on lengths and widths of petal measurements of the samples.

Applying the pseudo-Boolean polynomials reduction program, we reduce all samples to 2×1 matrices. After applying combinations of like terms and dropping of zero columns, the pseudo-Boolean polynomials of the instance in Table 2 reduces to

$$\begin{bmatrix} 6.0 \\ 2.4y_2 \end{bmatrix}$$

Applying the reduction method on every sample, results in a list of samples reduced to 2×1 matrices of the form $a + by_2$ which can be plotted and classified on a Cartesian plane by simple boundary lines.

Our results on this dataset also expose the overfitting flaw which may be overlooked in works that abuse artificial neural networks such as the reported 100% clustering accuracy reported in [21].

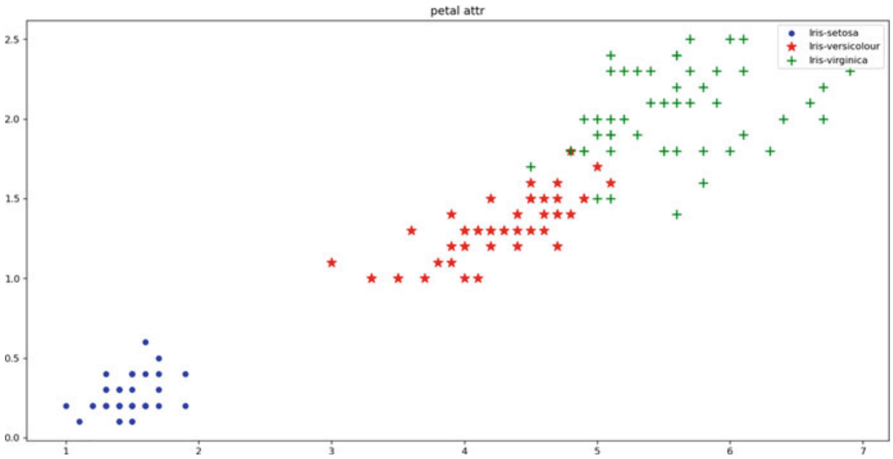


Fig. 2 Scatter on petal length-width

4.2 *The Wisconsin Diagnostic Breast Cancer (WDBC) Dataset*

The Wisconsin Diagnostic Breast Cancer (WDBC) [10] dataset, is another classical and popular dataset in the machine learning community. The dataset has 569 instances of 30 real-valued features that describe characteristics of the cell nuclei present in digitized images extracted by a fine needle aspirate (FNA) on a breast mass [22].

Ten real-valued features were computed for each cell nucleus:

1. radius (mean of distances from centre to points on the perimeter)
2. texture (standard deviation of greyscale values)
3. perimeter
4. area
5. smoothness (local variation in radius lengths)
6. compactness ($perimeter^2/area - 1.0$)
7. concavity (severity of concave portions of the contour)
8. concave points (number of concave portions of the contour)
9. symmetry
10. fractal dimension ("coastline approximation" – 1)

The *mean, standard error (se), and "worst"* or largest (mean of the three largest values) of these features were computed for each image, resulting in 30 features [22].

The task on this dataset is to predict whether a breast cancer diagnosis is *benign* or *malignant* based on these features.

The best known predictive accuracy (97.5%) was obtained by using a separating plane in the 3-D space of Worst Area, Worst Smoothness and Mean Texture features using repeated tenfold cross-validations [22], and this classifier has correctly diagnosed 176 consecutive new patients as of November 1995 [10].

The separating plane was obtained using Multisurface Method-Tree (MSM-T) [23], a classification method which uses linear programming to construct a decision tree where relevant features are selected using an exhaustive search in the space of 1–4 features and 1–3 separating planes [23].

Samples of size 3×10, result in 30-Dimensionality representations, that are hard to visualise on a Cartesian plot and consequently hard for the analyst to identify the features that are useful in making an accurate diagnosis.

To qualify our dimensionality reduction tool, we input each sample as a $3 \times n; n \in 1, 2, 3, \ldots, 10$ matrix where rows J represent the measurement type (*Mean*, *Standard error (se)*, *Worst*) and the columns I the measured features.

Applying the pseudo-Boolean polynomials formulation, we reduce all samples to 3×1 matrices which can be plotted and classified on a Cartesian space by a simple boundary plane. Since our formulation runs in polynomial time, we can iterate all possible arrangements of measured features and discover the set of measured features which has the best fitting plane that separates samples into *benign* or *malignant*.

5 Results and Discussion

5.1 The Iris Flower Dataset

Figure 3 illustrates the identification of cluster boundaries by simple identification of lines that best separate the aggregated samples on a Cartesian plane. Plotting the coefficients of the reduced instances, where the terms *1* and *y2* are abscissa and ordinate respectively, allows us to visualize properly the possible clusters from the data.

The most important thing to note is that this reduction correctly represents the original instances with less information cost, and allows us to discover a pair of straight lines that separate the respective clusters.

All instances that lie in the region $y \leq \frac{x}{4} + 2$, are classified safely as *Iris-setosa*, while the line $y = \frac{13x}{20} + 5$, separates *Iris-versicolor* from *Iris-virginica*.

It is also important to note that, some instances which seemed distinct of each other, actually have similar reduced pseudo-Boolean polynomials form. e.g.

$$\begin{bmatrix} 5.4 & 3.4 \\ 1.7 & 0.2 \end{bmatrix}$$

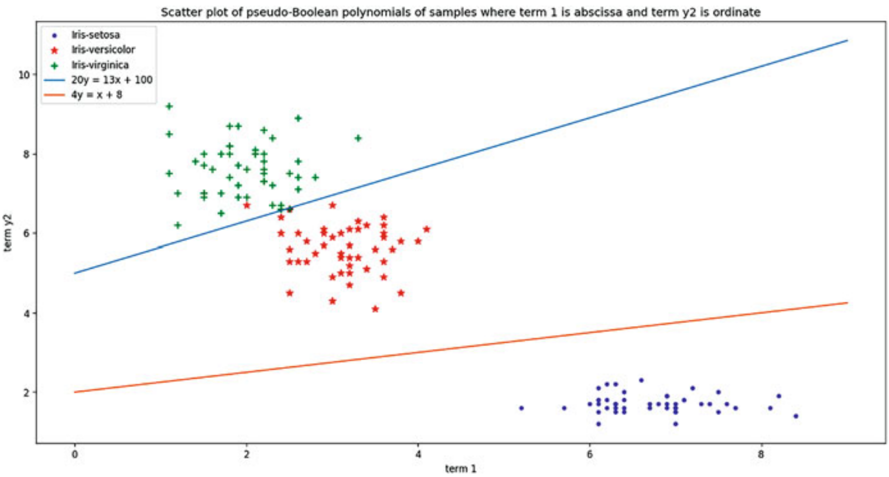


Fig. 3 Cartesian plot on resultant 2 × 1 dimensions after reduction by pseudo-Boolean polynomials formulation and the best boundary lines

and

$$\begin{bmatrix} 5.1 & 3.7 \\ 1.5 & 0.4 \end{bmatrix}$$

all reduce to

$$\begin{bmatrix} 1.9 \\ 6.9y_2 \end{bmatrix}$$

and in perfect confirmation of the *equivalence* condition; the instances also lie in the same cluster.

Looking at the single outlier,

$$\begin{bmatrix} 5.1 & 3.7 \\ 1.5 & 0.4 \end{bmatrix}$$

which was reduced to

$$\begin{bmatrix} 2.0 \\ 6.7y_2 \end{bmatrix}$$

and classified as *Iris-virginica* instead of *Iris-versicolor* we observe the faulty nature of learning-based cluster methods such as support-vector machines and [21]'s X-Boosted artificial neural network, which output 100% cluster accuracy on some test runs. The original measurements of this outlier, perfectly fit the cluster *Iris-virginica* as there are samples of *Iris-virginica* that have very little difference in measured values to this outlier, while its values are also significantly distinct from the other *Iris-versicolor* samples. The incorrectly clustered instance might be attributed to misclassification by the persons who labelled the dataset, or perhaps just a naturally occurring outlier.

This finding exposes, the overfitting concerns that arise from learning-based methods, like the Space Vector Machine (SVM) and [21]'s X-Boosted artificial neural network, that would report 100% accuracy in some tests. Additionally, changing the train/test data, result in varied accuracies when these methods are used, thereby losing the invariant and reproducibility attributes that our method guarantees.

5.2 The Wisconsin Diagnostic Breast Cancer (WDBC) Dataset

Plotting the coefficients of the reduced instances just as in the previous dataset, allows us to visualize properly the possible clusters from the data.

The reduction correctly represents the original instances with less information cost, and allows us to discover a combination of features with the best plane that separates the respective clusters.

Of all the combinations of features explored, the combination with the best separating plane (95.4% accuracy) consisted of

1. radius
2. texture
3. perimeter
4. smoothness
5. compactness
6. concavity
7. symmetry
8. fractal dimension

excluding *area* and *concave points* features (Fig. 4).

Instances with the shortlisted features are separable by a plane $z = 85x - 2y - 0.4$ at an accuracy of 95.4%.

Although, falling short of the accuracy (97.5%) reported in [10], our method manages to present the dimensionality reduction capacity of a simple and invariant method that is solely based on manipulation of orderings.

As shown in the experiments results reported in this paper, our method can be used for unsupervised clustering of multidimensional data as well as in feature selection processes in cluster analysis. Our method allows us to have a better understanding of how multidimensional features contribute to classification of samples in an invariant and explainable manner and in some cases achieve unbiased and unsupervised clustering in cluster analysis processes.

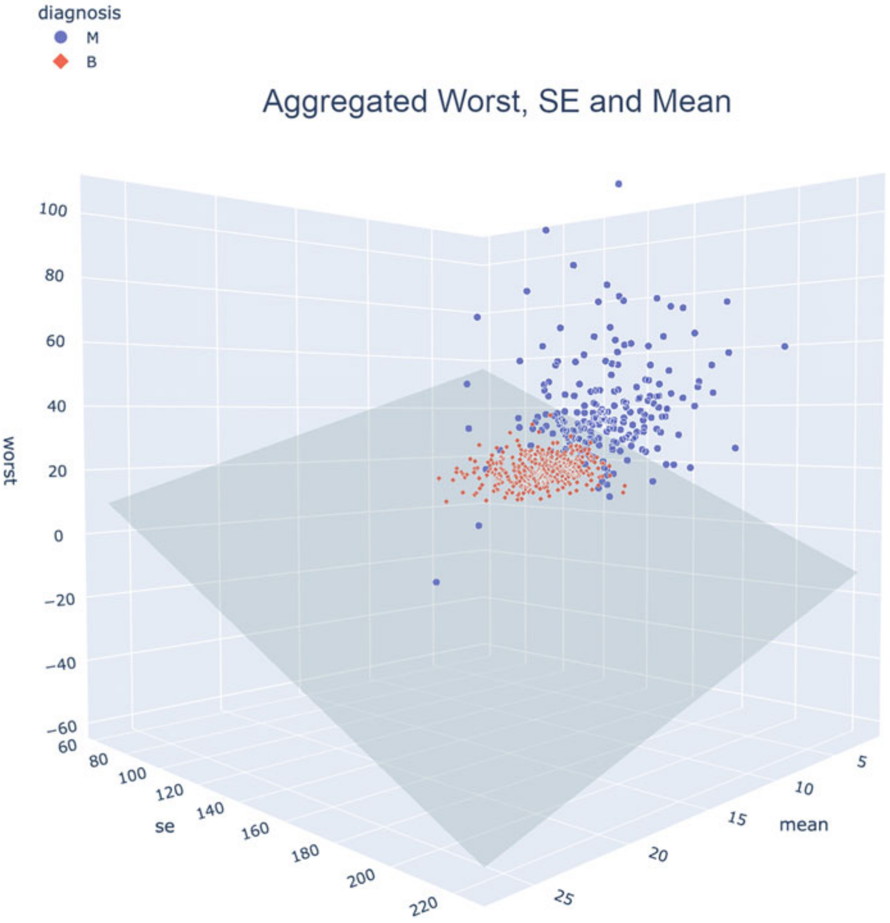


Fig. 4 Cartesian plot on resultant 3 × 1 dimensions after reduction by pseudo-Boolean polynomials formulation and the best separating plane

6 Conclusion

In this paper, we managed to showcase a combinatorial method for dimensionality reduction for cluster analysis, based on the formulation and reduction of pseudo-Boolean polynomials.

We tested our method on simple datasets, and managed to show that we can classify data samples with competitive accuracies by simple and linear data slices (lines and planes). Our proposed solution is invariant and interpretable while avoiding biases that may be involved when we use statistical methods because each sample is reduced in an independent manner, solely based on its own description.

It can be noted that dimension reduction using pseudo-Boolean polynomials on high-level features is a powerful tool for lossless dimension reduction and has potential of accelerating low memory representation of complex data that empowers other complex tasks like interpretable unsupervised clustering in computer vision, bioinformatics, natural language processing and machine learning.

We expect to reproduce even better state-of-the-art accuracies on other data science tasks, when we apply more powerful tools like decision trees and artificial neural networks on instances in their reduced pseudo-Boolean polynomials forms.

The biggest takeaway from our findings is the invariability and interpretability nature of the dimension reduction process using pseudo-Boolean polynomials formulation.

Acknowledgments Tendai Mapungwana Chikake and Boris Goldengorin's research was supported by Russian Science Foundation project No. 21-71-30005.

Boris Goldengorin acknowledges Scientific and Educational Mathematical Center "Sofia Kovalevskaya Northwestern Center for Mathematical Research" for financial support of the present study (agreement No 075-02-2023-937, 16.02.2023).

References

1. Schölkopf, B., Smola, A., Müller, K.R.: Nonlinear component analysis as a Kernel eigenvalue problem. Neural Comput. **10**(5), 1299–1319 (1998)
2. Johnstone, I., Titterington, D.: Statistical challenges of high-dimensional data. Philos. Trans. A Math. Phys. Eng. Sci. **367**, 4237–53 (2009)
3. Notley, S., Magdon-Ismail, M.: Examining the use of neural networks for feature extraction: a comparative analysis using deep learning, support vector machines, and K-nearest neighbor classifiers. arXiv preprint arXiv:1805.02294 (2018)
4. Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., Grosse, R., McCandlish, S., Kaplan, J., Amodei, D., Wattenberg, M., Olah, C.: Toy Models of Superposition. arXiv e-prints arXiv:2209.10652 (2022). https://doi.org/10.48550/arXiv.2209.10652
5. Burnham, K.P., Anderson, D.R., Burnham, K.P.: Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach, 2nd edn. Springer, New York (2002)
6. Kopp, W., Akalin, A., Ohler, U.: Simultaneous dimensionality reduction and integration for single-cell ATAC-seq data using deep learning. Nat. Mach. Intell. **4**(2), 162–168 (2022)
7. Sarveniazi, A.: An actual survey of dimensionality reduction. Am. J. Comput. Math. **04**(02), 55–72 (2014)
8. Sinharay, S.: An overview of statistics in education. In: Peterson, P., Baker, E., McGaw, B. (eds.) International Encyclopedia of Education, 3rd edn., pp. 1–11. Elsevier, Oxford (2010)
9. Goldengorin, B., Krushinsky, D., Pardalos, P.M.: Cell Formation in Industrial Engineering, Springer Optimization and Its Applications, vol. 79. Springer, New York (2013)
10. Street, W.N., Wolberg, W.H., Mangasarian, O.L.: Nuclear feature extraction for breast tumor diagnosis. In: Acharya, R.S., Goldgof, D.B. (eds.) Biomedical Image Processing and Biomedical Visualization, vol. 1905, pp. 861–870. International Society for Optics; Photonics; SPIE (1993)
11. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Ann. Eugenics **7**(2), 179–188 (1936)

12. Maaten, L.J.P.v.d., Hinton, G.E.: Visualizing high-dimensional data using t-SNE. J. Mach. Learn. Res. **9**(Nov), 2579–2605 (2008)
13. Jolliffe, I.: Principal Component Analysis. Springer (2002)
14. Jiang, H., Eskridge, K.M.: Bias in principal components analysis due to correlated observations. In: Conference on Applied Statistics in Agriculture (2000)
15. Bengio, Y., Monperrus, M., Larochelle, H.: Nonlocal estimation of manifold structure. Neural Comput. **18**(10), 2509–2528 (2006)
16. McLachlan, G.J.: Discriminant Analysis and Statistical Pattern Recognition. Wiley Series in Probability and Mathematical Statistics. Wiley, New York (1992)
17. Singh, S., Silakari, S.: Generalized discriminant analysis algorithm for feature reduction in cyber-attack detection system. Int. J. Comput. Sci. Inform. Secur. **6**(1), 173–180 (2009)
18. McInnes, L., Healy, J., Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018)
19. Boros, E., Hammer, P.L.: Pseudo-boolean optimization. Discrete Appl. Math. **123**(1–3), 155–225 (2002)
20. AlBdaiwi, B.F., Ghosh, D., Goldengorin, B.: Data aggregation for p-median problems. J. Comb. Optim. **21**(3), 348–363 (2011)
21. Sarkar, T.: Xbnet: An extremely boosted neural network. Intell. Syst. Appl. **15**, 200097 (2022). https://doi.org/10.1016/j.iswa.2022.200097. https://www.sciencedirect.com/science/article/pii/S2667305322000370
22. William Wolberg, O.M.: Breast Cancer Wisconsin (Diagnostic) (1993). https://doi.org/10.24432/C5DW2B. https://archive.ics.uci.edu/dataset/17
23. Bennett, K.P.: Decision tree construction via linear programming. Computer Sciences Technical Report # 1067, University of Wisconsin, 14 pp., January (1992)

Pseudo-Boolean Polynomials Approach to Edge Detection and Image Segmentation

Tendai Mapungwana Chikake, Boris Goldengorin, and Alexey Samosyuk

1 Introduction

In digital image processing and computer vision, image segmentation is the process of partitioning a digital image into multiple image segments, also known as image regions or image objects [1]. The process has close relationship to blob extraction, which is a specific application of image processing techniques, whose purpose is to isolate (one or more) objects (aka. regions) in an input image [2].

The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyse [1].

There exist classical and AI based methods for the purpose of segmentation/blob-extraction. These methods converge into semantic [3], instance [4] and panoptic [5] segmentation. The underlying techniques of these methods can be classified into threshold filtering, clustering, differential motion subtraction, histogram, partial-differential equation solving, graph partitioning, supervised neural-network association and edge detecting methods. Our proposed method lies at the intersection of edge detection and threshold filtering methods.

T. M. Chikake · A. Samosyuk
Department of Discrete Mathematics, Moscow Institute of Physics and Technology, Moscow, Russian Federation
e-mail: tendaichikake@phystech.edu; alexeysamosyuk@phystech.edu

B. Goldengorin (✉)
Department of Mathematics, New Uzbekistan University, Tashkent, Uzbekistan

The Scientific and Educational Mathematical Center «Sofia Kovalevskaya Northwestern Center for Mathematical Research», Pskov State University, Pskov, Russia

Department of Discrete Mathematics, Phystech School of Applied Mathematics and Informatics, Moscow Institute of Physics and Technology, Dolgoprudny, Moscow region, Russia
e-mail: b.goldengorin@newuu.uz; goldengorin.bi@mipt.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_5

Image segmentation plays a pivotal role, as a preprocessing step in localizing region of interest in content-based image retrieval, anomaly detection in industrial imagery, medical imaging, object detection and recognition tasks.

Our proposed method utilise the *reduction*, *equivalence* and *degree* properties of penalty-based pseudo-Boolean polynomials for purposes of extracting regions of interest which provide course segmentation that can be extended to image segmentation. We derive our methodology from [6]'s work on penalty-based pseudo-boolean formulation for purposes of data aggregation on p-median problems in the context of image processing.

Our method bases its operation on a combination of threshold filtering and edge detection by deterministically grouping masks which convey regions of colour gradient shift.

Localizing regions of interest in a blind manner is a difficult task, often requiring pattern recognition of big image data to derive useful utility. Our approach avoids learning of patterns from data and operate in a deterministic manner.

In our method, we partition an input image into small patches which we treat as *information cost* matrices. We aggregate these matrices into their smallest possible pseudo-Boolean polynomials and group together equivalent instances, thereby delineating spatially contrasting regions in image representations.

We verify our method by applying it to simple images containing primitive shapes and then scale up to simple natural scene images where we show that the method can competitively extract segments.

Our proposed method seeks to

- introduce a blind preprocessing step for semantic segmentation
- assist with unsupervised annotation of segmentation datasets
- promote deterministic approaches to computer vision solutions.

The proposed method requires tuning of three parameters to balance performance and fine-tuned segmentation results.

2 Related Work

Currently, our proposed method acts as a supportive step to a final segmentation processor. This means that our method is still a complementary processing tool to complete an image segmentation task.

Complete semantic segmentation requires the clustering of parts of an image together and proposing an object class while instance segmentation is concerned with detecting and delineating each distinct object of interest appearing in an image [7].

Our current solution cannot yet attach object classes but can distinctly delineate some individual objects in an image. The method achieves this by separating neighbouring pixels into blob or edge region based on the degree of a pseudo-Boolean polynomial calculated on patches extracted from the image.

The resulting masks can be refined into classifications or delineations of distinct objects by an additional process which we are still working on. Our end goal is to achieve a solution for complete segmentation.

By the time of writing, popular solutions to instance and semantic segmentations are mostly based on either Mask R-CNN [8] or the U-Net Convolutional Network [9]. Both solutions are learning-based methods and require large amounts of labelled ground truth data. Neural network based methods also require accelerated computer processors to perform in reasonable time. Our solution avoids learning rules from data and can be easily run on non-accelerated CPUs.

Since our method is based on classifying edge and blob regions in an image, the Canny edge detector [10], introduced in 1986 by John F. Canny, is a closely related technique. The Canny edge detector is an edge detection operator which uses a multi-stage algorithm to detect a wide range of edges in images. The Canny method detects edges by first applying a Gaussian filter to smooth the image in order to remove the noise, followed by the finding of the intensity gradients in the image [11]. After these steps, the method applies a gradient magnitude threshold filtering or lower bound cut-off suppression to get rid of spurious response to edge detection and then apply a double threshold to determine potential edges and conclude its process by tracking edges by hysteresis [10]. The Canny algorithm is adaptable to various environments because its parameters allow it to be tailored to recognition of edges of differing characteristics depending on the particular requirements of a given implementation [11]. The optimised Canny filter is recursive, and can be computed in a short, fixed amount of times, but the implementation of the Canny operator does not give a good approximation of rotational symmetry and therefore gives a bias towards horizontal and vertical edges [11]. In contrast, our method, calculate the pseudo-Boolean polynomial in normal and transposed patch matrices and selects the pseudo-Boolean polynomial form with the highest degree during the blob/edge classification step, thereby avoiding the edge direction bias.

From an indirect perspective we can look at blind aggregation of possible distinct objects in image data in terms of blob extraction. Popular solutions using this paradigm, have underlying usage of either of the popular blob extraction methods which include Laplacian of Gaussian [12], Difference of Gaussian [13], or Determinant of a Hessian [14], among other methods.

Laplacian of Gaussian is a blob extraction method which determines the blobs by using the Laplacian of Gaussian filters [12]. The Laplacian is a 2-D isotropic measure of the second spatial derivative of an image which highlights regions of rapid intensity change and is therefore, often used for edge detection [12]. We often apply the Laplacian after an image smoothening method with something approximating a Gaussian smoothing filter in order to reduce its sensitivity to noise. In our method, the Gaussian smoothing filter step is an optionally used when applied to noisy image instances.

The Difference of Gaussian method determines blobs by using the difference of two differently sized Gaussian smoothed images and follows generally most of the concept of the Laplacian of Gaussian [12].

Determinant of a Hessian to aggregate regions of possible segmentation is achieved by determining blobs using the maximum in the matrix of the Hessian determinant [14].

Generally, blob extraction based methods, propose small and numerous regions of interest making them hard to extend into spatial segmentation.

On the other hand, our method aggregates equivalent regions, by assigning zero or pseudo-Boolean polynomials with lower degree as blob regions and edge otherwise. Regions initially described in different colour distributions in the pixel array often output high-order pseudo-Boolean polynomials which indicate contour regions. This property allows us to extract larger and fewer spatial regions of interest in an image, making our method stand out against the other blob aggregation methods.

3 Methods

In mathematics and optimization, a pseudo-Boolean function is a function of the form $f : \mathbf{B^n} \rightarrow \mathbb{R}$, where $\mathbf{B} = \{0, 1\}$ is a Boolean domain and n is a non-negative integer called the degree of the function [15].

Goldengorin et al. [16] highlights fundamental reduction and equivalence properties of penalty-based pseudo-Boolean polynomials which guarantee the maintenance of underlying initial information while reducing the problem complexity.

We utilise this formulation on image patches to achieve compact representations of patches, and based on the *degrees* and *equivalence* [16] properties of the resultant pseudo-Boolean polynomials, we can qualify an edge/blob classifier on input patches.

Given an image patch represented by an $m \times n$ sized matrix which we treat as an information cost matrix C, our first task is to determine the minimum possible way of representing this cost in a way that allows us to compare if the patch is extracted from a blob region or a contour region.

A patch that overlaps regions of contrasting information (i.e. overlapping an edge) in an image, results in a representation that is costly in comparison to one that lies over a blob region. We claim this assertion because matrix values in blob regions are usually equivalent or have very small differences between each other and since [16]'s pseudo-Boolean polynomial formulation is penalty-based, similar costs cancel out.

Reducing the patches to their smallest pseudo-Boolean polynomials provides comparable instances which can be tested for equivalence as well. This property is termed *equivalence* in [16] and can be used to fine-tune edges or detect similar regions on the image matrix.

The Pseudo-Boolean representation according to [6] requires the generation of a coefficients matrix and its respective terms' matrix, whose combination creates monomials of the pseudo-Boolean polynomial.

After the generation of these matrices, most of the processing: local aggregation, reduction of columns, and p-truncation processes are heavily dependent on the terms' matrix.

Using an example instance to illustrate the formulation process, we take a 4 × 5 patch from an image.

Let

$$C = \begin{bmatrix} 8 & 8 & 8 & 5 \\ 12 & 7 & 5 & 7 \\ 18 & 2 & 3 & 1 \\ 5 & 18 & 9 & 8 \end{bmatrix}$$

The terms encoding function takes as input the permutations (Π) matrix which is an index ordering of the input matrix.

$$\Pi = \begin{bmatrix} 4 & 3 & 3 & 3 \\ 1 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 \\ 3 & 4 & 4 & 4 \end{bmatrix}$$

Using Π, we sort the initial cost matrix C

$$\text{sorted}\, C = \begin{bmatrix} 5 & 2 & 3 & 1 \\ 8 & 7 & 5 & 5 \\ 12 & 8 & 8 & 7 \\ 18 & 18 & 9 & 8 \end{bmatrix}$$

and derive the $\varDelta C$ matrix

$$\Delta C = \begin{bmatrix} 5 & 2 & 3 & 1 \\ 3 & 5 & 2 & 4 \\ 4 & 1 & 3 & 2 \\ 6 & 10 & 1 & 1 \end{bmatrix}$$

Using the Π matrix we calculate the terms' matrix

$$\mathbf{y} = \begin{bmatrix} \\ y_4 & y_3 & y_3 & y_3 \\ y_1 y_4 & y_2 y_3 & y_2 y_3 & y_1 y_3 \\ y_1 y_2 y_4 & y_1 y_2 y_3 & y_1 y_2 y_3 & y_1 y_2 y_3 \end{bmatrix}$$

and derive the resulting pseudo-Boolean polynomial

$$\begin{bmatrix} 5 & 2 & 3 & 1 \\ 3y_4 & 5y_3 & 2y_3 & 4y_3 \\ 4y_1y_4 & 1y_2y_3 & 3y_2y_3 & 2y_1y_3 \\ 6y_1y_2y_4 & 10y_1y_2y_3 & 1y_1y_2y_3 & 1y_1y_2y_3 \end{bmatrix}$$

We then perform local aggregation by summing similar terms and get a compact representation of the initial instance as

$$\begin{bmatrix} 0 & 0 & 11 \\ 0 & 11y_3 & 3y_4 \\ 2y_1y_3 & 4y_2y_3 & 4y_1y_4 \\ 0 & 12y_1y_2y_3 & 6y_1y_2y_4 \end{bmatrix}$$

which in this particular example has 50% less cost compared to the initial instance when expressed as polynomial.

We then search for the equivalent matrix with the minimum number of columns and in this particular example, it is already reduced to this state.

When pseudo-Boolean polynomials are reduced to their smallest instance, an equivalence property is apparent because different instances of similar information converge into a similar reduced pseudo-Boolean polynomials.

This property is central to the blob aggregation task, and in this task these regions of equivalence in the reduced pseudo-Boolean polynomials context, occupy the set of blobs.

Below are examples of cost matrices, which are initially different but converge into similar reduced instances.

$$\begin{bmatrix} 138 & 138 & 138 & 136 \\ 139 & 139 & 138 & 137 \\ 142 & 141 & 139 & 138 \\ 142 & 140 & 139 & 138 \end{bmatrix}$$

and

$$\begin{bmatrix} 136 & 136 & 138 & 140 \\ 138 & 137 & 138 & 140 \\ 140 & 139 & 140 & 141 \\ 139 & 139 & 140 & 141 \end{bmatrix}$$

reduce to

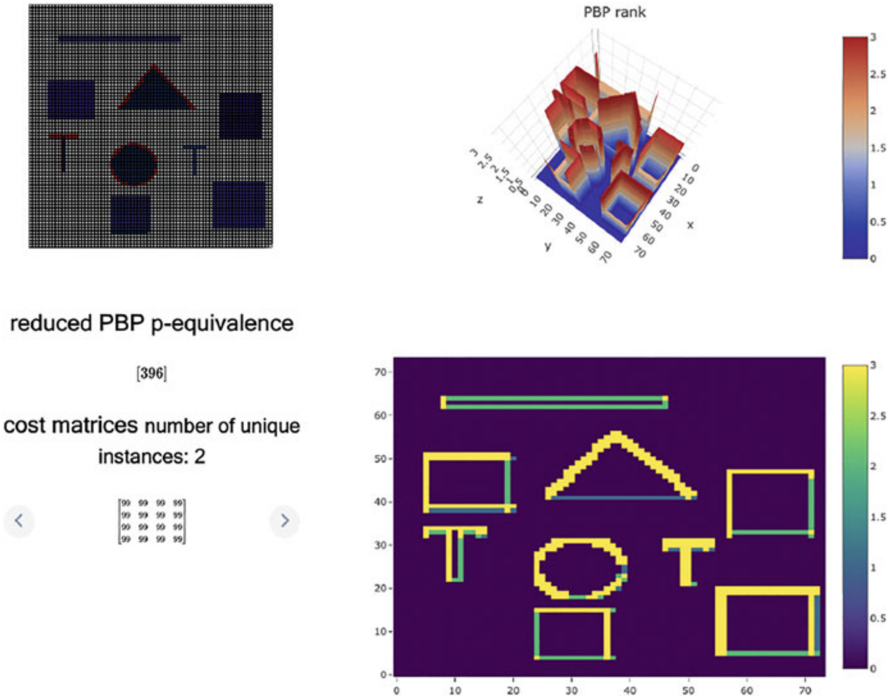


Fig. 1 Pseudo-Boolean polynomial degree plotted for an image of primitive shapes

$$\begin{bmatrix} 550 \\ 3y_1 \\ 6y_1y_2 \\ 1y_1y_2y_4 \end{bmatrix}$$

By plotting the degree of the pseudo-Boolean polynomials of each patch on a surface plot, we observe contours available in spatial features of a given image as shown in Fig. 1.

For our edge/blob classifier, we use the pseudo-Boolean polynomial degree r to classify whether the particular patch lies over a contour or a blob region. We select $p < m$ to be the cut-off threshold for classification and based on this value, we can alter how fine/course should our edges be. If the degree r of the pseudo-Boolean polynomial calculated on the patch is higher than p then the patch lies over an edge or blob otherwise. The edge/blob classifier if simply

$$f(r, p) = \begin{cases} \text{edge} & \text{if}\, r < p, \\ \text{blob} & \text{otherwise} \end{cases} \tag{1}$$

Given an image, broken into small patches of size 4 × 4, the maximum possible degree is 3, and we can select $p = 1$, such that we have a binary set of patches, where patches, whose pseudo-Boolean polynomials reduce to a constant are described as blob regions, and the rest as edge points.

Using *equivalence* [6], we can fine-tune the classified edges, should neighbouring patches exhibit contrasting edge/blob classes by equating the blobs in favour of the edge or vice-versa depending on how fine we require our edges to be.

4 Experimental Setup

Given an image of size 200 × 200 containing basic primitive shapes of continuous colour shown in Fig. 2a, we extract patches of significantly smaller sizes, e.g. 6 × 6 as shown in Fig. 2b.

We then apply the formulation and reduction of pseudo-Boolean polynomials on each patch and group them into a binary set $S = \{\text{Blob}, \text{Edge}\}$ based on the pseudo-Boolean polynomial degree.

Patches whose pseudo-Boolean polynomial degree $r < p$ are considered **Blobs** and **Edge** otherwise.

Patches with constant pixel values $x \in [0, 255]$ like

$$\begin{bmatrix} 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 \end{bmatrix}$$

are guaranteed to converge to constant (zero-degree pseudo-Boolean polynomials),

$$\begin{bmatrix} 396 \end{bmatrix}$$

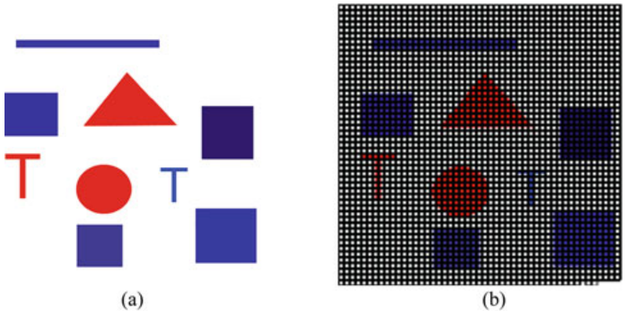


Fig. 2 Input image of primitives and its patching process. (**a**) Input image. (**b**) Showing 6 × 6 sized patches

while some, with varied pixel values which cancel out like

$$\begin{bmatrix} 254 & 254 & 19 & 84 \\ 254 & 254 & 19 & 84 \\ 254 & 254 & 19 & 84 \\ 254 & 254 & 19 & 84 \end{bmatrix}$$

also converge to constant (zero-degree pseudo-Boolean polynomials)

$$\begin{bmatrix} 611 \end{bmatrix}$$

Consequently these patches are grouped among the set of patches whose information cost is described as blob region.

There are patches which contain varied pixel data which may converge to high-order pseudo-Boolean polynomials like

$$\begin{bmatrix} 254 & 254 & 6 & 17 \\ 254 & 254 & 6 & 17 \\ 254 & 254 & 6 & 17 \\ 254 & 254 & 6 & 123 \end{bmatrix}$$

and

$$\begin{bmatrix} 254 & 254 & 6 & 17 \\ 254 & 254 & 6 & 123 \\ 254 & 254 & 6 & 123 \\ 254 & 254 & 6 & 123 \end{bmatrix}$$

which converge to

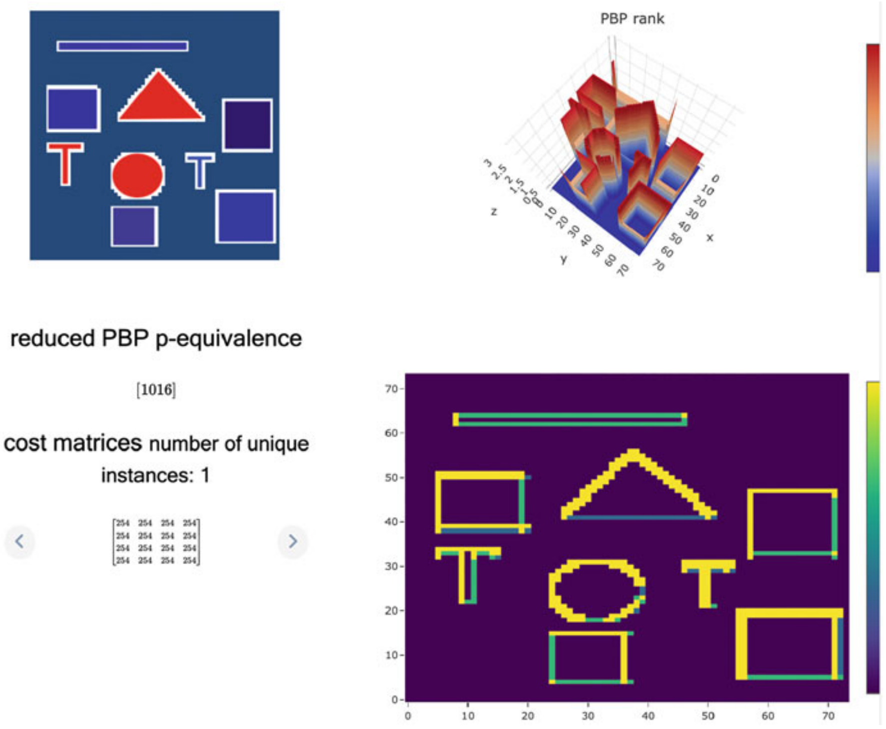
$$\begin{bmatrix} 531 \\ 106y_1y_2y_3 \end{bmatrix}$$

a 3rd-degree pseudo-Boolean polynomial which would be classified as lying on edge regions, should the cut-off threshold be given as $r = 2$. In the particular simple case of the image in Fig. 2a, patches which lie at shape edges converge to these non-zero degree pseudo-Boolean polynomials.

By colour-coding the patches which belong to the set of patches which have pseudo-Boolean polynomial's degree $r = 0$, we can observe regions of colour continuation, which we can classify as blobs as shown in Fig. 3.

As can be observed from Fig. 3, the method allows us to find border points (white coloured) on simple objects found in the image.

Applying the same process on a natural and unprocessed image may not result in desirable or useful aggregation as natural images often have less drastic



reduced PBP p-equivalence

[1016]

cost matrices number of unique instances: 1

$$\begin{bmatrix} 254 & 254 & 254 & 254 \\ 254 & 254 & 254 & 254 \\ 254 & 254 & 254 & 254 \\ 254 & 254 & 254 & 254 \end{bmatrix}$$

Fig. 3 Colour coding *blue* to discovered, zero-degreed pseudo-Boolean polynomials of input instances

transition of spatial features. For this reason we ported the use of a Gaussian filter as a preprocessing step, followed by a pixel set aggregation step which is essentially a multivalued threshold processing. Instead of raw pixels as input in our patches, we group ranges into *sets of pixels* whose size depends on the variance of pixel distribution in the image to promote group convergence of pseudo-Boolean polynomials.

We create these $f(x) =$ *sets of pixels* as

$$f(x) = \begin{cases} 0 & \text{if } x < 5, \\ 1 & \text{if } x \in [5, 10), \\ 2 & \text{if } x \in [10, 15), \\ \ldots & \\ \ldots & \\ \ldots & \\ 51 & \text{if } x > 250, \end{cases} \tag{2}$$

thereby reducing the information cost range from [0, 255] to [0, 51] for instance.

Natural images tend to have smooth transitions of pixel values for neighbouring pixels at atomic level due to anti-aliasing in RGB representation of image data, thereby reducing the chances of neighbouring pixel patches converging into equivalent groups and consequently coarse edges but the Gaussian filter preprocessing together with the grouping of pixel ranges promotes finer edges.

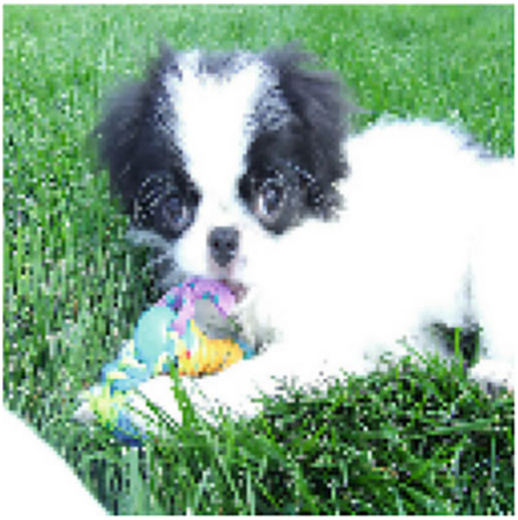
5 Results and Discussion

We apply our aggregation process on natural images and observe the need for the Gaussian filter as well as the pixel set aggregation preprocessing to achieve useful segmentation. Figure 4 shows input image of a noisy and natural image that we pass through the segmentation processes without preprocessors as shown in Fig. 5, and with processors as shown in Fig. 6.

As can be observed in Fig. 6, a Gaussian preprocessing and pixel grouping step allow us to achieve better edge and blob extraction. A Set of pixels, each of size 40, limit our cost range from [0, 255] to [0, 7] and encourage pronunciation of contrasting regions.

Additionally, we can observe that including a costly operation which aggregates those reduced pseudo-Boolean polynomials into equivalent groups finds numerous, insignificantly small and unuseful groups in the setups which exclude the preprocessing steps, while larger equivalent groups can be aggregated in the setups which involve all the preprocessing steps.

Fig. 4 Input natural image for preprocessing comparison



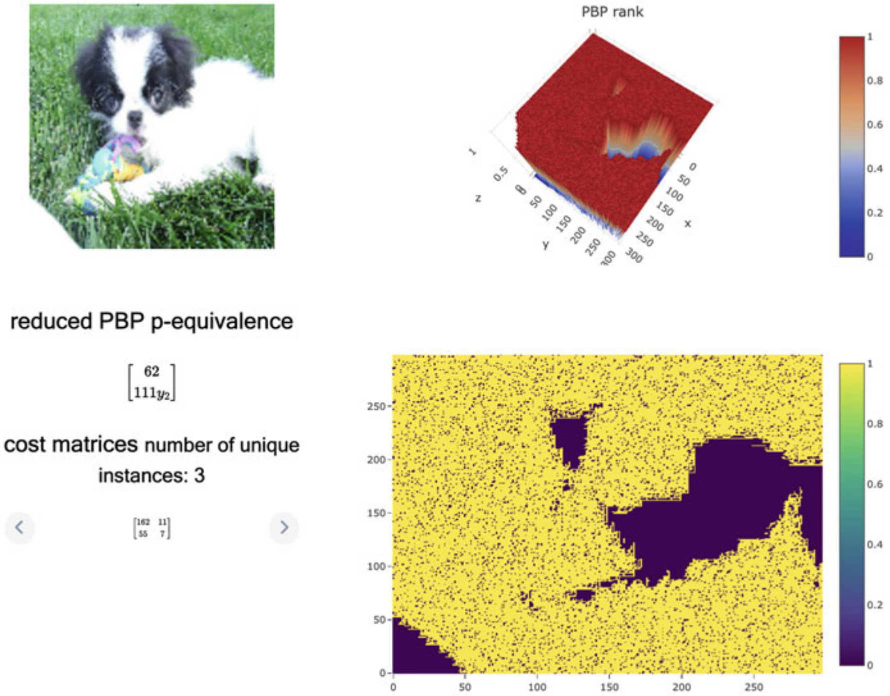

PBP rank
z
y
x
1
0.5
0
50
100
150
200
250
300
0.8
0.6
0.4
0.2
reduced PBP p-equivalence
$\begin{bmatrix} 62 \\ 111y_2 \end{bmatrix}$
cost matrices number of unique instances: 3
<
$\begin{bmatrix} 162 & 11 \\ 55 & 7 \end{bmatrix}$
>
250
200
150
100
50
0

Fig. 5 Segmentation without the pre-processing steps

5.1 Dubai Landscape Dataset

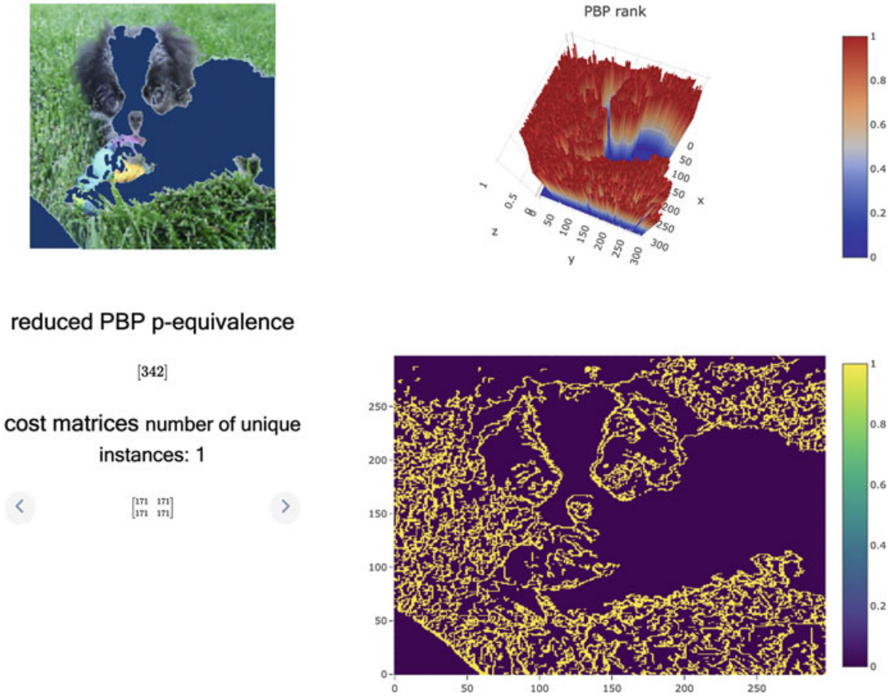
We apply our method with selected processing parameters on the Dubai landscape dataset [17]. In this experiment we show that our method can extract edges of landscape features on satellite imagery which can be used for semantic segmentation.

Humans in the Loop published an open access dataset annotated for a joint project with the Mohammed Bin Rashid Space Center in Dubai, UAE, which consists of aerial imagery of Dubai obtained by MBRSC satellites and annotated with pixel-wise semantic segmentation in 6 classes [17].

The full solution on this dataset requires placing labels on each segmented region, however our current method can only segment regions on boundary edges.

The distinctive advantages of our method against the state-of-art neural network based solutions in instance segmentation are:

- blind segmentation(no learning is involved, which can be prone to overfitting/under fitting issues)
- faster and CPU friendly segmentation
- explainable mathematical steps to segmentation.



reduced PBP p-equivalence

[342]

cost matrices number of unique instances: 1

$\begin{bmatrix} 171 & 171 \\ 171 & 171 \end{bmatrix}$

Fig. 6 Segmentation with all the pre-processing steps included

Our method is also not limited to a given dataset, since it works in a blind manner, which does not require prior familiarity of related image data except for purposes of choosing the threshold ranges. Provided with images which contain contrasting features, we can guarantee that our method will propose segmentations of features in pure mathematical and deterministic steps.

Figure 7 show an example processing of our method on an image sample in the Dubai Landscape dataset [17].

Our method brings to limelight unbiased and fast computer vision in a segmentation task. The parameters required to achieve useful segmentation using our method are limited to:

1. Gaussian filter kernel size,
2. pixel thresholding size and,
3. patch sizes.

The optimal choice of these parameters is the only limitations for the generalization of our proposed method, and we propose in our future work, an automized process for selecting these parameters.

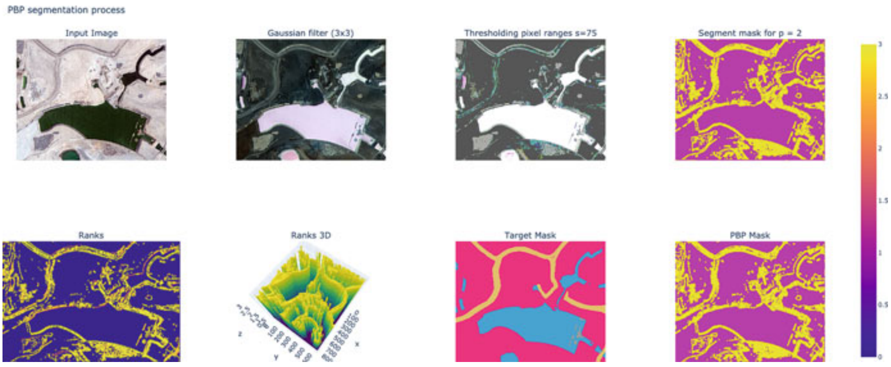


Fig. 7 Example segmentation [Dubai landscape]

The performance of the method, based on the choice of the patch size parameter is linearly dependent: the smaller the patch size, the finer the segmentation but longer processing, and vice-versa.

6 Conclusion

In this article, we presented our proposed method of formulating pseudo-Boolean polynomials on image patches which results in unsupervised edge detection, blob extraction and image segmentation processes. We managed to show that our proposed method, works in a fast, unbiased and competitively accurate manner in segmenting contrasting regions in image data. We plan to automate the choosing of processing parameters so that our method achieves full functionality as a general purpose image segmentation tool and one of the major tasks in our next challenges is focused on grouping blob regions based on colour histograms to provide labels and consequently achieve complete semantic segmentation.

Acknowledgments Tendai Mapungwana Chikake and Boris Goldengorin's research was supported by Russian Science Foundation project No. 21-71-30005.

Boris Goldengorin acknowledges Scientific and Educational Mathematical Center "Sofia Kovalevskaya Northwestern Center for Mathematical Research" for financial support of the present study (agreement No 075-02-2023-937, 16.02.2023).

References

1. Shapiro, L.G., Stockman, G.C.: Computer Vision. Prentice Hall, Upper Saddle River, NJ (2001)
2. Yusuf, M.D., Kusumanto, R., Oktarina, Y., Dewi, T., Risma, P.: Blob analysis for fruit recognition and detection. Comput. Eng. Appl. J. **7**(1), 25–36 (2018)

3. Guo, D., Pei, Y., Zheng, K., Yu, H., Lu, Y., Wang, S.: Degraded image semantic segmentation with dense-gram networks. IEEE Trans. Image Process. **29**, 782–795 (2020)
4. Yi, J., Wu, P., Jiang, M., Huang, Q., Hoeppner, D.J., Metaxas, D.N.: Attentive neural cell instance segmentation. Med. Image Anal. **55**, 228–240 (2019)
5. Kirillov, A., He, K., Girshick, R., Rother, C., Dollár, P.: Panoptic segmentation arXiv.org (2018). https://arxiv.org/abs/1801.00868v3
6. AlBdaiwi, B.F., Ghosh, D., Goldengorin, B.: Data aggregation for p-median problems. J. Comb. Optim. **21**(3), 348–363 (2011)
7. Chennupati, S., Narayanan, V., Sistu, G., Yogamani, S., Rawashdeh, S.A.: Learning panoptic segmentation from instance contours (2021). http://arxiv.org/abs/2010.11681. ArXiv:2010.11681 [cs]
8. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN (2018). https://doi.org/10.48550/arXiv.1703.06870. http://arxiv.org/abs/1703.06870. ArXiv:1703.06870 [cs]
9. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. Tech. Rep. arXiv:1505.04597 (2015)
10. Canny, J.: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-8**(6), 679–698 (1986)
11. Liu, K., Xiao, K., Xiong, H.: An image edge detection algorithm based on improved canny. In: Proceedings of the 2017 5th International Conference on Machinery, Materials and Computing Technology (ICMMCT 2017). Atlantis Press, Beijing, China (2017)
12. Kong, H., Akakin, H.C., Sarma, S.E.: A generalized Laplacian of Gaussian filter for blob detection and its applications. IEEE Trans. Cybern. **43**(6), 1719–1733 (2013)
13. Assirati, L., Silva, N.R., Berton, L., Lopes, A.A., Bruno, O.M.: Performing edge detection by difference of Gaussians using q-Gaussian kernels. J. Phys. Conf. Ser. **490**, 012020 (2014)
14. Li, S., Liu, C., Wang, Y. (eds.): Pattern Recognition: 6th Chinese Conference, CCPR 2014, Changsha, China, November 17–19, 2014. Proceedings, Part I, Communications in Computer and Information Science, vol. 483. Springer Berlin Heidelberg, Berlin (2014)
15. Boros, E., Hammer, P.L.: Pseudo-boolean optimization. Discrete Appl. Math. **123**(1–3), 155–225 (2002)
16. Goldengorin, B., Krushinsky, D., Pardalos, P.M.: Cell Formation in Industrial Engineering, Springer Optimization and Its Applications, vol. 79. Springer New York, New York (2013)
17. Loop, H.i.t.: Semantic segmentation dataset | humans in the loop (2020). https://humansintheloop.org/resources/datasets/semantic-segmentation-dataset-2/. Accessed: 2022-12-03

Purifying Data by Machine Learning with Certainty Levels

Shlomi Dolev and Guy Leshem

1 Introduction

Motivation A fundamental paradigm used for autonomic computing, self-managing systems, and decision-making under uncertainty and faults is machine learning. Classification of machine learning algorithms that are designed to deal with Byzantine (or malicious) data are of great interest since a realistic model of learning from examples should address the issue of Byzantine data. Previous work, as described below, tried to cope with this issue by developing new algorithms using a boosting algorithm (e.g., “AdaBoost”, “Logitboost” etc.) or other robust and efficient learning algorithms e.g., [13]. These efficient learning algorithms tolerate relatively high rates of corrupted data. In this paper we try to handle the issue using a different approach, that of introducing the certainty level measure as a tool for coping with corrupted data items, and of combining learning results in a new and unique way. We present two new approaches to increase the certainty levels of machine learning results by calculating a certainty level that takes into account the corrupted data items in the training data-set file. The first scheme is based on identifying statistical parameters when the distribution is known (e.g., normal distribution) and using an assumed bound on the number of corrupted data items to bound the uncertainty in the classification. The second scheme uses decision trees, similar to the random forest techniques, incorporating the certainty level to the leaves. The use of the certainty level measure in the leaves yields a better

S. Dolev
Ben-Gurion University of the Negev, Be’er Sheva City, Israel
e-mail: dolev@cs.bgu.ac.il

G. Leshem (✉)
Ashkelon Academic College, Ashkelon, Israel
e-mail: leshemg@cs.bgu.ac.il

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_6

collaborative classification when results from several trees are combined to a final classification.

Previous Work In the Probably Approximately Correct (PAC) learning framework, Valiant [14] introduced the notion of PAC learning in the presence of malicious noise. This is a worst-case model of errors in which some fraction of the labeled examples given to a learning algorithm may be corrupted by an adversary who can modify both example points and labels in an arbitrary fashion. The frequency of such corrupted examples is known as the malicious noise rate. This study assumed that there is a fixed probability β ($0 < \beta < 1$) of an error occurring independently on each request, but the error is of an arbitrary nature. In particular, the error may be chosen by an adversary with unbounded computational resources and knowledge of the function being learned, the probability distribution and the internal state of the learning algorithm (note that in the standard PAC model the learner has access to an oracle returning some labeled instance $(x,C(x))$for each query, where $C(x)$ is some fixed concept belonging to a given target class C and x is a randomly chosen sample drawn from a fixed distribution D over the domain X. Both C and D are unknown to the learner and each randomly drawn x is independent of the outcomes of the other draws.

In the malicious variant of the PAC model introduced by Kearns and Li [8], the oracle is allowed to 'flip a coin' for each query with a fixed bias η for heads. If the outcome is heads, the oracle returns some labeled instance (x,ℓ) antagonistically chosen from $X \times \{-1,+1\}$. If the outcome is tails, the oracle is forced to behave exactly like in the standard model returning the correctly labeled instance $(x,C(x))$ where $x \sim D$ (x is a drawn sample from the distribution D).

In both the standard and malicious PAC models the learner's goal for all inputs $\varepsilon, \Delta > 0$ is to output some hypothesis $H \in \mathcal{H}$ (where $\mathcal{H}$ is the learner's fixed hypothesis class) by querying an oracle at most m times for some $m = m(\varepsilon, \Delta)$ in the standard model, and for some $m = m(\varepsilon, \Delta, \eta)$ in the malicious model. For all targets $C \in \mathcal{C}$ and distributions D, the hypothesis H of the learner must satisfy $E_{x \sim D}[H(x) \neq C(x)] \leq \varepsilon$ with a probability of at least $1 - \Delta$ with respect to the oracle's randomization. We will call ε and Δ the accuracy and the confidence parameter, respectively. Kearns and Li [8] have also shown that for many classes of Boolean functions (concept classes), it is impossible to accurately learn ε if the malicious noise rate exceeds $\frac{\varepsilon}{1+\varepsilon}$. In fact, for many interesting concept classes, such as the class of linear threshold functions, the most efficient algorithms known can only tolerate malicious noise rates significantly lower than this general upper bound.

Despite these difficulties, the importance of being able to cope with noisy data has led many researchers to study PAC learning in the presence of malicious noise [1–3, 5, 6, 9, 13]. In Servedio [13], a PAC boosting algorithm is developed using smooth distributions. This algorithm can tolerate low malicious noise rates but requires access to a noise-tolerant weak learning algorithm of known accuracy. This weak learner, L, which takes as input a finite sample S of m labeled examples, has some tolerance to malicious noise; specifically, L is guaranteed to generate a hypothesis with non-negligible advantage provided that the frequency of noisy examples in its sample is at most 10% and that it has a high probability to learn with high accuracy in the presence of malicious noise at a rate of 1%.

Our Contribution We present a verifiable way to cope with arbitrary faults introduced by even the most sophisticated adversary, and show that the technique withstands this malicious (called Byzantine) intervention so that even in the worst case scenario the desired results of the machine learning algorithm can be achieved. The assumption is that an unknown part of a data-set is Byzantine, namely, introduced to mislead the machine learning algorithm as much as possible. Our goal is to show that we can ignore/filter the influence of the misleading portions of the malicious data-set and obtain meaningful (machine learning) results. In reality, the Byzantine portion in the data-set may be introduced by a malfunctioning device with no adversarial agenda, nevertheless, a technique proven to cope with the Byzantine data items will also cope with less severe cases. In this paper, we develop three new approaches for increasing the certainty level of the learning process, where the first two approaches identify and/or filter data items that are suspected to be Byzantine data items in the data-set (e.g., a training file). In the third approach we introduce the use of the certainty level for combining machine learning techniques (similar to the previous studies).

The first approach fits best the case in which the Byzantine data is added to the data-set, and is based on the calculation of the statistical parameters of the data-set. The second approach considers the case where part of the data is Byzantine, and extends the use of the certainty level for those cases in which no concentrations of outliers are identified. Data-sets often have several features (or attributes) which are actually columns in the training and test files that are used for cross-checks and better prediction of the outcome in both simple and sophisticated scenarios. The third approach deals with cases in which the Byzantine data is part of the data and appear in two possible modes: where part of the data in a feature is Byzantine and/or where several features are entirely Byzantine. The third technique is based on decision trees similar to the *Random Forest* algorithm [4]. After the decision trees are created from the training data, each variable from the training data passes through these decision trees, and whenever the variable arrives to a tree leaf, its tree classification is compared with its class. When the classification and the class are in agreement, a *right* variable of the leaf is incremented; otherwise, the value of a *wrong* variable of this leaf is incremented. The final classification for every variable will be determined according to the right and wrong values. This enhancement of the random forest is of an independent interest conceptually and practically, improving the well known random forest technique.

Road Map The rest of the paper is organized as follows: In the next section (Sect. 2), we describe approaches for those cases in which Byzantine data items are added to the data-set, and the ways to identify statistical parameters when the distribution of a feature is known. In Sects. 3 and 4, we present those cases in which the Byzantine adversary receives the data-set and chooses which items to add/corrupt. Section 3 describes ways to cope with Byzantine data in the case of a single feature with a classification of a given certainty level. Section 4 extends the use of the certainty level to handle several features, extending and improving the random forest techniques. The conclusion appears in Sect. 5.

2 Addition of Byzantine Data

We start with the cases in which Byzantine data is added to the data-set. Our goal is to calculate the statistical parameters of the data-set, such as the distribution parameters of the uncorrupted items in the data-set, despite the addition of the Byzantine data. Consider the next examples that derive the learning algorithm to the wrong classification, where the raw data contains one feature (or attribute) of the samples (1 vector) that obeys some distribution (e.g., normal distribution), plus additional adversary data. The histogram that describes such an addition is presented on the left side of Fig. 1, where the "clean" samples are inside the curve and the addition of corrupted data is outside the curve (marked in blue). The corrupted data items in these examples are defined as samples that cause miscalculation of statistical parameters like μ and σ and as a result, the statistical variables are less significant. Another case of misleading data added to the data-set, a special case to the one above, is demonstrated on the right side of Fig. 1. The histogram of these samples is marked in green, where the black vertical line that crosses the histogram separates samples with labels $+1$ and -1. The labels of the misleading data are inverted with relation to the labels of other data items with the same value. To achieve our goal to calculate the most accurate statistical parameters for the feature's distribution in the sample population, we describe a general method to identify and filter the histograms that may include a significant number of additional corrupted data items.

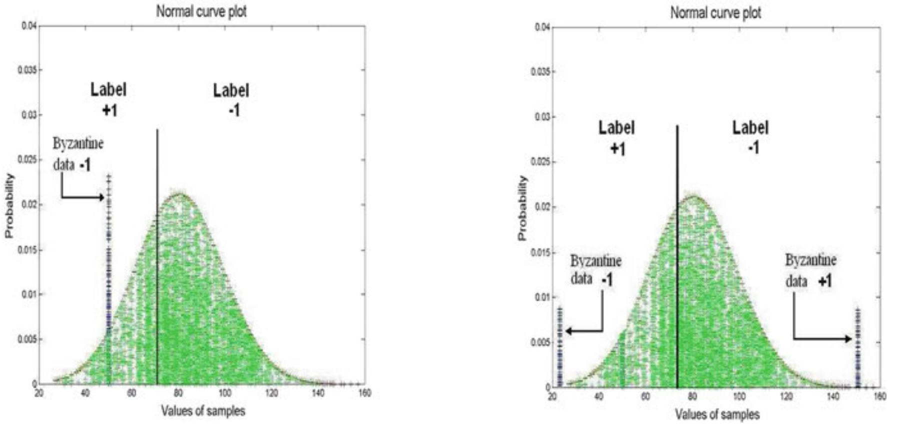


Fig. 1 Histogram of original samples with additional corrupted data outside the normal curve but in the bound of $\mu \pm 3\sigma$ (left), and outside the normal curve and outside the bound of $\mu \pm 3\sigma$ (right)

Method for Identifying Suspicious Data and Reducing the Influence of Byzantine Data This first approach is based on the assumption that we can separate "clean" data by a procedure based on the calculation of the μ and σ parameters of the uncorrupted data. According to the central limited theorem, 30 data items chosen uniformly, which we call a *batch*, can be used to define the μ and σ. Thus, the first step is to try to find at least 30 clean samples (with no Byzantine data). Note that according to the central limit theorem, the larger the set of samples, the closer the distribution is to being normal, therefore, one may choose to select more than 30 samples. We use n=30 as a cutoff point and assume that the sampling distribution is approximately normal. In the presence of Byzantine data one should try to ensure that the set of 30 samples will not include any Byzantine items. This case is similar to the case of a shipment of N objects (real data) in which m are defective (Byzantine). In probability theory and statistics, hypergeometric distribution describes the probability that in a sample of n distinctive objects drawn from the shipment, exactly k objects are defective. The probability for selecting k items that are not Byzantine is:

$$P(X = k) = \frac{\binom{m}{k}\binom{N-m}{n-k}}{\binom{N}{n}} \tag{1}$$

Note that for clean samples k=0 and the equation will be

$$P(X = 0) = \frac{\binom{N-m}{n}}{\binom{N}{n}} \tag{2}$$

In order to prevent the influence of the adversary on the estimation of μ and σ (by addition of Byzantine data), we require that the probability in equation 2 will be higher than 50% ($P > \frac{1}{2}$). Additionally, according to the Chernoff bound we will obtain a lower bound for the success probability of the majority of n independent choices of 30-sample batches (thus, by a small number of batch samplings we will obtain a good estimation for the μ and σ parameters of clean batches). The ratio between N (all samples) to m (Byzantine samples) that implies a probability to sample a clean batch that is greater than $\frac{1}{2}$ is presented in Fig. 2.

As demonstrated in Fig. 2 the ratio between N (all samples), and m (Byzantine samples) is about 2% (e.g., 20 Byzantine samples for every 1000 samples, so if this Byzantine ratio is found by the new method (as described below) the probability that any other column in the data-set will contain a Byzantine sample is very low (in other words, the confidence that in every other column the samples are "clean" is high)). Our goal is to sample a majority of "clean" batches to estimate statistical parameters such as μ and σ of the non-Byzantine samples in the data-set. The estimation of these parameters will be done according to an **Algorithm 1**.

The procedure below:

Algorithm 1 Estimate statistical parameters

```
For 1 to the chosen B do (B will be selected according to the Chernoff bound*),
    Randomly and uniformly choose a batch of size $n$ (e.g., $n = 30$) from the population of interest(e.g., one feature),
    Compute the desired batch statistic $\mu$ and $\sigma$ $\left(\mu = \frac{1}{n}\Sigma_{i=1}^{n} x_i, and, \sigma = \sqrt{\frac{1}{n-1}\Sigma_{i=1}^{n}(x_i - \overline{x})^2}\right)$,
end for
On the assumption that the distribution of the original data is normal or approximately normal, the histogram of the estimated $\mu$ and $\sigma$ is also approximately normal (according to the central limit theorem). The probability to choose a “clean” batch is higher than 50%, therefore, at least 50% or more of the estimations are clean. The value of $\hat{\mu}$ (and $\hat{\sigma}$) will be chose to be the median of the $\mu$ (and $\sigma$), thus ensuring that our choice has at least one clean batch with higher (and one with lower) $\mu$ (and $\sigma$,respectively).
$\star$ The Chernoff bound gives a lower bound for the success probability of majority agreement for $b$ independent, equally likely events, and the number of trials is determined according to the following equation: $B \geq \frac{1}{2(P-1/2)^2} ln \frac{1}{\sqrt{\epsilon}}$, where the probability $P > \frac{1}{2}$ and $\epsilon$ is the smallest probability that we can promise for an incorrect event (e.g., for the probability of a correct event at a confidence level of 95% or 99%, the probability for an incorrect event, $\epsilon$, is 0.05 or 0.01, respectively).
```

Algorithm 1: Description of a method for estimating statistical parameters like μ and σ.

Using Expected Value and Variance to Predict Distribution Shape Up to this stage, we used the central limit Theorem (CLT), stating that: the average samples of observations uniformly drawn from some population with any distribution shape is approximately distributed as a normal distribution, resulting in the expected value

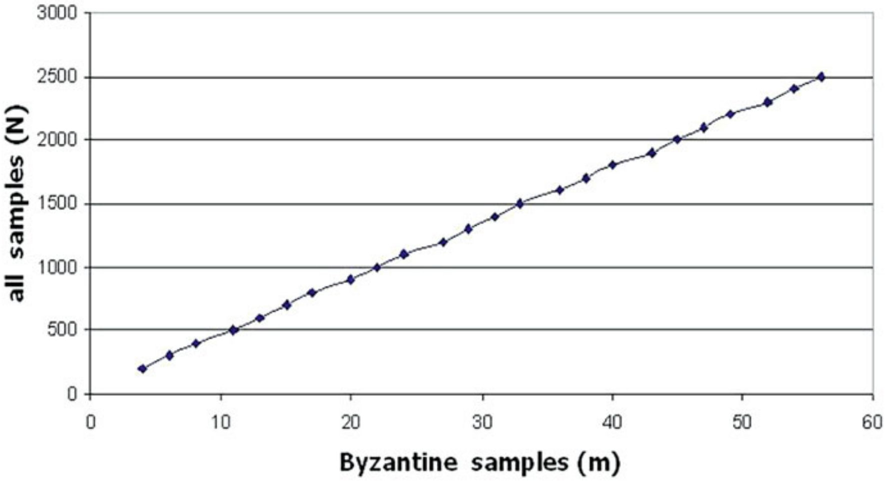


Fig. 2 Ratio between N (all samples) to m (Byzantine samples) for P $\geq \frac{1}{2}$

and the variance. Based on CLT, we were able to efficiently obtain (using Chernoff bound) the expected value and the variance of the data item values. Next, for every given number of data items, and type of distribution graph, the parameters of the graph that will respect these values (expected value, variance, distribution type, and number of data items) can be found. In the sequel, we consider the case of a distribution type of graph which reflects the normal distribution. The next stage for identifying suspicious data items is based on analysis of the overflow of data items beyond the distribution curve (Fig. 1). The statistical parameters which were found in the previous stage are used in the procedure described in **Algorithm 2**, the procedure below:

Algorithm 2 Technique for removing suspected data

1. Take the original sample population of interest (e.g., one feature from the data set) and create a histogram of that data,
2. Divide the histogram into Θ bins within the range $\mu \pm 3\sigma$ (e.g., $\Theta = 94$), and count the actual number of data items in every bin,
3. Compute the number of data items in every bin by using the integral of the normal curve according to μ and σ (which were found by the previous method (*Algorithm 1*)) multiplied by the number of "clean" samples*, and compare with the actual number,
4. If the ratio between the counted number of data items in a bin and the computed number according to the integral is higher than $1+\xi$ (e.g., ξ=0.5), the data items in this column are suspect,
5. The samples from the suspicious bins will be marked and will not be considered by the machine learning algorithm.

   $\star$ *N*, the number of the "clean" samples can be define to be 98% of the total number f data items, assuming the data-set contains at most 2% Byzantine items. Alternately one may estimate the number of the "clean" samples using the calculated μ and σ. We assume that batches μ (and σ) very near to the selected μ (and σ) represent "clean" population. Thus, the bins from the histogram with these values are probably clean. The ratio between the original number of data items in this clean bin to the integral of the normal curve for this bin can be used as an estimation for *N*.

Algorithm 2: Description of the first technique for removing suspected data items.

The suspicious bins, those with a significant overflow, are marked and will not be considered for the training process of the machine learning. The data-set after the cleaning process contains values from bins (in the data histogram) without overflow (e.g., the ratio between the integral of the normal curve to the data items in the same bin is approximately 1).

Note that when the number of extra data items in the bins (which was counted during the "cleaning" process) with overflow (data items outside the integral curve) is higher than 2% of the whole data-set, we can assume that the other bins are clean. The next section deals with the remaining uncertainty.

Experiments and Results We present results to demonstrate the qualities of our new approach, and to reveal the differences between a data-set with and without corrupted data in order to validate the proposed method experimentally. The

Table 1 Results of C4.5 algorithm, on original and corrupted data

Test No.	Dataset	Accuracy
1	Artificial data	100%
2	Artificial data with corrupted data	78%
3	Artificial data after removing suspicious data	96%

comparison between the data-sets was done with the C4.5 algorithm [12]. C4.5 uses a decision tree developed by Quinlan [11], and is an extension of Quinlan's earlier ID3 (Iterative Dichotomiser 3) algorithm [10]. The decision trees generated by C4.5 can be used for classification, and for this reason C4.5 is often referred to as a statistical classifier. For this comparison we used an available tool based on the C4.5 file from "Classification Toolbox for Matlab". The results appear in Table 1.

The data-set for the first experiment is an artificial data-set created by Matlab software with the function "normrnd" (which generates random numbers from a normal distribution with a mean parameter μ and a standard deviation parameter σ)—this database consist of uniformly distributed data and contains 1 attribute (2296 samples) and 2 classes (-1,$+1$). Note that in the presence of Byzantine data items the resulting data-set must be a worse case with relation to the original data items and the learning algorithm. Here we deliberately introduced a corrupt data-set that includes Byzantine data to demonstrate the benefits of the proposed technique. For the second and third experiments, the previous database is extended with corrupt data (such as the data-set shown in Fig. 1) that also has an inverted label in relation to the label with the same value (2296 "clean" plus 45 "corrupted", 2341 samples in total).

3 Corruption of Existing Data, Single Feature Learning with a Certainty Level

We continue considering the case where part of the data in the feature is corrupted. Our goal in this section is to find the certainty level of every sample in the distribution in the case where the upper bound on a number of corrupted data items is known. This section is actually a continuation of the previous, as both sections deal with a single feature, where the first deals with an attempt to find overflow of samples and the second, cope with unsuccessful such attempts; either due to the fact that the distribution is not known in advance, or that no overflows are found. The histogram of these samples is colored green, where the black vertical line that crosses the histogram separates samples with labels $+1$ and -1. The labels of the Byzantine data have an inverted label with relation to the label of the non-Byzantine data items with the same value. To achieve our goal we describe a general method that bounds the influence of the Byzantine data items.

Method to Bound the Influence of the Byzantine Data Items The new approach is based on the assumption that an upper ξ on the number of Byzantine data items that may exist in every bin in the distribution is known (e.g., maximum ξ equals 8 items). The certainty level ζ of each bin is calculated by the following equations:

$$\zeta_{-1} = \frac{L_{-1} - \xi}{N} \tag{3}$$

$$\zeta_{+1} = \frac{L_{+1} - \xi}{N} \tag{4}$$

Where L_{-1} is the number of data items that are labeled as -1, L_{+1} is the number of data items that are labeled as $+1$, and N is the number of data items in the bin.

Algorithm 3 Finding the certainty level

```
1. Take the original sample of size $n$ from the population of interest (e.g., one feature from the data set),
2. Sort the $n$ data items (samples) according to their value and create their histogram,
3. Count data items at every bin, where the size of bin is the value of natural number in the histogram ± 0.5 (e.g., for the natural number 73, the bin is between 72.5 to 73.5) and count the number of data items that are labeled as $-1$ and $+1$.
4. Find the certainty level $\zeta$ of each bin according to equations 3 and 4, and the assumption of the size of the maximum $\xi$.
```

Algorithm 3: Description of the method for finding the certainty level of every sample for ξ Byzantine data items in every bin in the distribution.

4 Corruption of Existing Data, Multi-Feature Learning (with a New Decision Trees Algorithm)

Our last contribution deals with the general cases in which corrupted data are part of the data-set and can appear in two modes: (*i*) An entire feature is corrupted (Fig. 3), and (*ii*) Part of the features in the data-set is corrupted and the other part is clean. Note that there are several ways to corrupt an entire feature, including: (1) inverting the classification of data items, (2) selection of random data items, and (3) producing classifications inconsistent with the classifications of other non-corrupted features. Our goal, once again, is to identify and to filter data items that are suspected to be corrupted. The first case (*i*) is demonstrated by Fig. 3, where the raw data items contain one feature and one vector of labels, where part of the features are totally

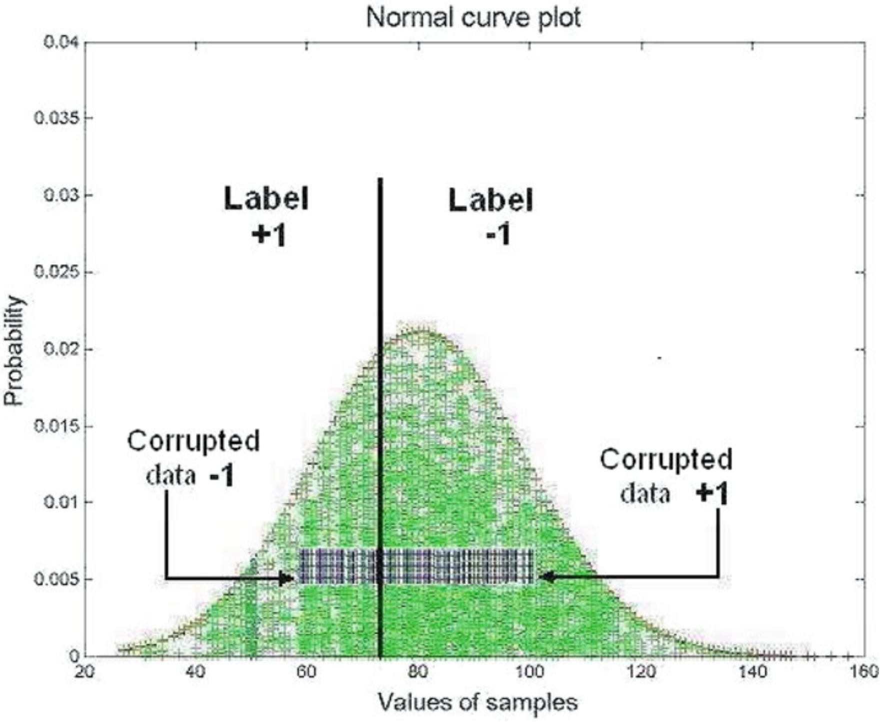


Fig. 3 Histogram of original samples with corrupted data inside the normal curve

non-corrupted and part are suspected to be corrupted (for all samples in this column there is a wrong classification).

Method to Bound the Influence of the Corrupted Data Items Our technique is based on the *Random Forest*; like the *Random Forest* algorithm [4] we use decision trees, where each decision tree that is created depends on the value of a random vector that represents a set of random columns chosen from the training data. Large numbers of trees are generated to create a Random Forest. After this forest is created, each instance from the training data set passes through these decision trees. Whenever a data set instances arrives to a tree leaf, its tree classification is compared with its class ($+1$ or -1); when the classification and the class agree the *right* instance of the leaf is incremented; otherwise the value of the *wrong* instance of this leaf is incremented, e.g., 351 instances were classified by Node 5 (leaf): 348 with the right classification and 3 with the wrong classification (Fig. 4).

Certainty Adjustment Due to Byzantine Data Bound The certainty level ζ of each leaf can be calculated based on the assumption that the upper bound on the number of corrupted data items ξ at every leaf in the tree is known. These calculations are arrived at using equations 3 and 4, where, L_{-1} is the number of

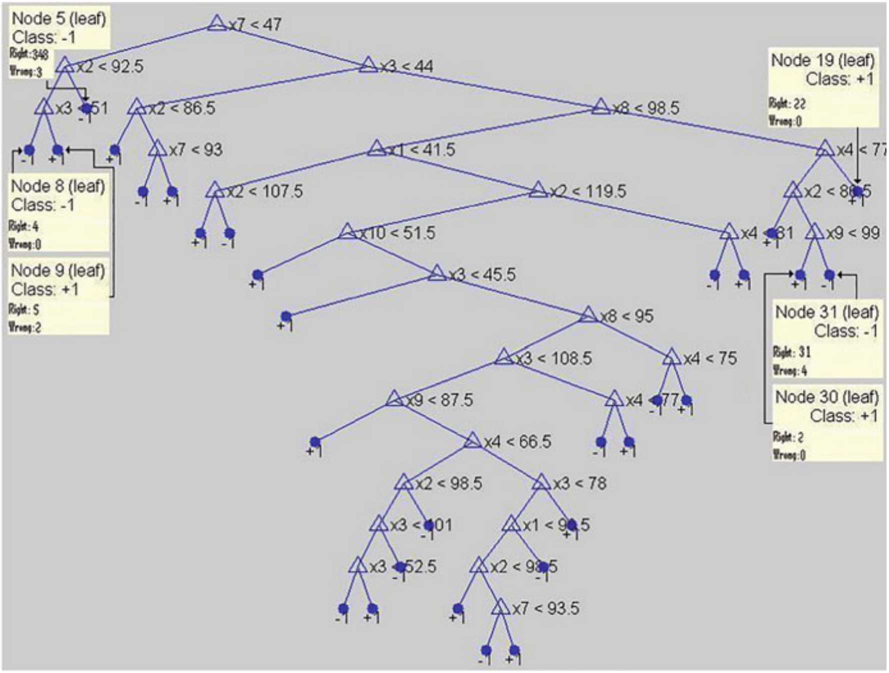


Fig. 4 Example of a decision tree for predicting the response for the instances in every leaf with right or wrong classification

variables (in the leaf) that are labeled as -1, L_{+1} is the number of instances (in the leaf) that are labeled as $+1$, and N is the total number of variables that were classified by the leaf.

In the second step, each instance from the test data set passes through these decision trees to get its classification. Each new tested instance will get a classification result and a confidence level, where the confidence level is in the terms of the (training) right and wrong numbers associated with the leaf in the tree. The final classification is a function of the vector of tuples $\langle classification; right; wrong; \rangle$ with reference to a certainty level rather than a function of the vector of $\langle classification \rangle$ which is used in the original *Random Forest* technique. In this study we show one possibility for using the vector of $\langle classification; right; wrong; \rangle$, though other functions can be used as well to improve the final classification.

Algorithm 4: Description of the method for identifying and filtering Byzantine data for multi-feature data-sets.

We tune down the certainty in each leaf using a given bound on the corrupted/Byzantine data items. The contribution of this part includes a conceptual improvement of the well known random forest technique; by re-examining all data items in the data set. The re-examination counts the number of right and wrong classifications in each leaf of the tree.

Algorithm 4 Identify and filter Byzantine data

```
1. First, select the number of trees to be generated, e.g. $K$,
2. **For** $k$=1 to $K$ **do**
3.     A vector $\theta_k$ is generated, where $\theta_k$ represents the data samples selected for creating the tree (e.g., random columns chosen from training data sets - these columns are usually selected iteratively from the set of columns, with replacement between iterations),
4.     Construct tree $T(\theta_k$,y) by using the decision tree algorithm,
5. **End for**
6. Each instance from the training data passes through these decision trees, and for every leaf the number of instances that are classified correctly (right) and incorrectly (wrong) are counted, then the percentages of right and wrong classifications are calculated,
7. Each instance from the test data set passes through these decision trees and receives a classification,
8. Each new instance will receive a result $\langle classification; right; wrong; \rangle$ from trees in the forest, right and wrong percentages from all the trees are summarized (e.g., sample 10 is classified by Tree No. 1 at Node 5 as $+1$ with 90% (or 0.9) correctness and 10% (or 0.1) incorrectness, by Tree No. 2 at Node 12 as $+1$ with 94% (or 0.94) correctness and 6% (or 0.06) incorrectness, where the total correctness of $+1$ for this sample from both trees is 92% (or 0.92) and 8% (or 0.08) for $-1$). The final classification for each instance will be determined according to the difference between the total correctness (right classifications) for $+1$ to the total incorrectness (wrong classifications) for $+1$ that are summarized from all trees$^\star$.
   $\star$ This is one option for using the *right* and *wrong* counters to determine the classification.
```

Experiments and Results In order to validate the proposed method experimentally, we present results to demonstrate the qualities of our new approach and reveal the differences when a data-set with and without corrupted data is processed. The comparison between the data-set was done with the Multi-Feature algorithm of *Algorithm 4*. Using the Matlab function *T*=treefit(*X,y*) the algorithm creates a decision tree *T* for predicting response *y* as a function of predictor *X*. *X* is an *n*-by-*m* matrix of predictor values. *y* is a vector of *n* response values (for classification). Another experiment was run to determine the number of wrong samples (at every leaf) the classification trees can handle. Thus, from the correctness and incorrectness values of every leaf (which were found in step 5 in the method above) certain numbers of samples will be subtracted (for correctness) or added (for incorrectness) and vice versa (e.g., leaf *5* at tree in Fig. 4 for class -1 was found with 348 samples with correct classification and 3 samples with incorrect classification. If 8 Byzantine samples are present in the leaf, the classification of the data will change to 340 with correct classification and 11 with incorrect classification. The results are presented in Table 2.

For the first experiment (Tests 1 and 2), we used the "Satimage" data-set which is a well known data-set for classification of a satellite image. The original data for this database was generated from data purchased from NASA by the Australian Center for Remote Sensing, and contains 36 attributes and 6 classes (2296 for training and 2000 for testing). For the second experiment (Tests 3 and 4), we used an artificial data-set which was created by Matlab software with the function "normrnd" (generates random numbers from the normal distribution with a mean

Table 2 Results of the Multi-Feature algorithm, on original and corrupted data

Test No.	Data set	Accuracy majority vote	Accuracy right and wrong
1	Setimage	88.1%	92.25%
2	Artificial data base on Setimage	88.7%	92.4%
3	Artificial data with corrupted data case 1	Low (less than 65%)	87.8%
4	Artificial data with corrupted data case 2	85.3%	92.4%
5	Setimage with 12 corrupted samples at every leaf	88.1%	89.20%

parameter μ and a standard deviation parameter σ)—this database is constructed by uniformly chosen data items, and contains 36 features (2296 samples) and 2 classes (−1,+1). For the third experiment (Tests 5 and 6), we used artificial data where some features (the entire columns) are corrupted and the other features are not (case 1). For the fourth experiment (Tests 7 and 8), we used artificial data where part of the data in the feature (column) are corrupted and some features (columns) are corrupted (case 2). For Tests 9 and 10 we used the original "Satimage" data-set. To summarize, we demonstrated that algorithm 4 significantly improve the classification process with and without Byzantine data.

5 Conclusion and Future Work

In this work we present the development (the details of the experiment results appear in [7] of three methods for dealing with corrupted data in different cases: The first method considers Byzantine data items that were added to a given non-corrupted data set. Batches of uniformly selected data items and Chernoff bound are used to reveal the distribution parameters of the original data set. The adversary, knowing our machine learning procedure, can choose, in the most malicious way on, up to the 2%. malicious data; Note, that there is no requirement for the additional noise to come from distribution different than the data items distribution. We prove that the use of uniformly chosen batches and the use of Chernoff bound reveals the parameters of the non-Byzantine data items. We propose to use certainty level that takes into account the bounded number of Byzantine data items that may influence the classification. The third method is designed for the case of several features, some of which are partly or entirely corrupted. We present an enhanced random forest technique based on certainty level at the leaves. The enhanced random forest copes well with corrupted data. We implemented a system and show that ours performs significantly better than the original random forest both with *and without* corrupted data sets; we are certain that it will be used in practice.

In the scope of distributed systems, such as sensor networks, the methods can withstand malicious data received from a small portion of the sensors, and still achieve meaningful and useful machine learning results.

References

1. Aslam, J., Decatur, S.: Specification and simulation of statistical query algorithms for efficiency and noise tolerance. J. Comput. Syst. Sci. **56**, 191–2087 (1998)
2. Auer, P.: Learning nested differences in the presence of malicious noise. Theor. Comput. Sci. **185**(1), 159–175 (1997)
3. Auer, P., Cesa-Bianchi, N.: On-line learning with malicious noise and the closure algorithm. Ann. Math. Artif. Intel. **23**, 83–99 (1998)
4. Breiman, L.: Random forests, Statistics department. Technical report, University of California, Berkeley (1999)
5. Cesa-Bianchi, N., Dichterman, E., Fischer, P., Shamir, E., Simon, U.H.: Ample-efficient strategies for learning in the presence of noise. ACM **46**(5), 684–719 (1999)
6. Decatur, S.: Statistical queries and faulty PAC oracles. In: Proc. Sixth Work. on Comp. Learning Theory, pp. 262–268 (1993)
7. Dolev, S., Leshem, G., Yagel, R.: Purifying data by machine learning with certainty levels. Technical Report August 2009, Dept. of Computer Science, Ben-Gurion University of the Negev (TR-09-06)
8. Kearns, M., Li, M.: Learning in the presence of malicious errors. SIAM J. Comput. **22**(4), 807–837 (1993)
9. Mansour, Y., Parnas, M.: Learning conjunctions with noise under product distributions. Inf. Proc. Lett. **68**(4), 189–196 (1998)
10. Mitchell, T.M.: Machine Learning. McGraw-Hill, New York (1997)
11. Quinlan, J.R.: Induction of decision trees. In: Machine Learning (1986)
12. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, Burlington (1993)
13. Servedio, A.R.: Smooth boosting and learning with malicious noise. J. Mach. Learn. Res. (4), 633–648 (2003)
14. Valiant, G.L.: A theory of the learnable. Commun. ACM **27**(11), 1134–1142 (1984)

On Impact of Data Models on Predictability Assessment of Time Series

Sergey Frenkel

1 Introduction

Significant advances in the development of forecasting systems achieved in the framework of Machine Learning (ML), primarily based on neural networks, do not eliminate the difficulties of their use in very critical cases, for example, as it turned out, when predicting the development of the COVID19 pandemic. Among the many reasons, the main ones of which are, of course, related to the lack of knowledge of the relevant subject areas, one can also point out that modern ML, in particular, those that using mathematical models based on statistical data, often do not take into account some mathematical properties of random data, which are used in specific MO systems with prediction models.

In the last decade, work on learning theory for sequence recognition (or online learning) has tried to avoid as far as possible the need to use assumptions about the exact types of probability distributions of the data. However presently most modern forecasting tools, primarily based on neural networks and Deep Learning, still use parametric statistical models, often based on Gaussian distributions for a priori estimates of the expected quality of a prediction. Assumptions about the Gaussian distribution are also essential for the SMV (Support Vector Machine [1]) algorithm [2]. It is shown that the accuracy of linear models can increase if the feature is distributed similarly to the normal law. Deep neural networks study predictive relationships using a number of non-linear layers to build intermediate representations of features, encoding previously obtained information into a latent variable z_t, while the final forecast is created only using z_t, without taking into account the possible characteristics of the data that exist within certain probabilistic

S. Frenkel (✉)
Federal Research Center “Computer Science and Control” RAS, Moscow, Russia
e-mail: fsergey@post.bgu.ac.il

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer
Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_7

models, and the main hopes for getting a suitable result are placed on computing power.

In a more general sense, in many modern tools based on neural networks (NN), the formation of a forecast of data values at points in time in the future is performed as a search for internal patterns and relationships without using any classical mathematical procedures (Markov models or other probability theory models) or underlying formal theories. Training is carried out on examples, and not on a mathematical model, although using various mathematical tools, such as linear or logistic regression, at intermediate stages – for example, algorithms based on search trees and boosting (XGB, etc.) [3]. These circumstances are often supplemented by the opacity of the transformation of the input data (i.e., the construction of features, for example, in deep learning algorithms [4]), which makes it difficult to a priori assess the impact of certain input data properties on the prediction efficacy. As the analysis shows, this can lead to incorrect forecasts, to the inefficiency of the tools and procedures of the forecasting subsystems as part of information and computing systems.

Therefore, it is important to make a preliminary assessment of how efficient a particular forecasting tool is from the point of view of the designer of a specific software forecasting tool (“Prediction instrument” – PI). Hereinafter, PI refers to software implementations of prediction algorithms.

Most research on network traffic forecasting has focused on classical statistical methods that rely heavily on the use of past (“historical”) data, and the time spent on obtaining a forecast depends largely on the computational complexity of the algorithm relative to the volume of this history.

In particular, in recent years there has been a significant increase in the number of studies using computational approaches, including machine learning methods, to predict various activities in complex computing systems – from user response on websites for various purposes, to traffic through it and/or workload prediction. However, the existing literature mainly focuses on the analysis of algorithms implemented in modern software products for solving specific problems, and there is no comprehensive review to answer many important questions, such as the impact of the different types of data models used and the specifics of the mathematical models underlying the basis of the algorithms used [3].

At the same time, most models assume that the more observations are used, the more accurate (better) the forecast. Therefore, the problem of high accuracy is associated with the problem of computational costs, and hence the speed. In cases where prediction time is critical, for example, for real-time network traffic prediction problem, the task of prediction is to minimize computational costs as much as possible while maintaining an acceptable level of accuracy (efficiency) of the predictor.

One way to solve this problem is to take into account and use the properties of mathematical time series models that model traffic and job flows, which can significantly affect the prediction. One way to ensure that the abundance of modern approaches and models is taken into account is to develop a general conceptual model of the prediction problem, which is actually absent in the modern literature.

This paper offers an overview of the indicated mathematical models for predicting the behavior of complex systems, primarily telecommunication systems.

Our main goal is to analyze the requirements for mathematical models of data and systems with processes occurring in them, whose behavior is required to be predicted using certain software tools, taking into account the availability of data on the functioning and typical ways of user actions with prediction tools.

We propose a certain categorization of modern algorithms and forecasting methods based on probabilistic models, primarily focusing on current progress in machine learning methods.

The probabilistic data models used in modern approaches for various prediction problems are considered and analyzed, and the conditions and requirements for prediction models/algorithms are formulated in the framework of probabilistic models for prediction, from the point of view of harmonization (coordination) both (data and prediction algorithms) models. Various formal-logical (semantic, ontological methods) [5] remain outside our field of vision.

A conceptual model of the process of selecting predictors for a specific data set is proposed and substantiated, on the basis of which a system of recommendations and a selection scheme can be built.

Various examples of the influence of the characteristics of random sequences and processes on the accuracy of prediction are considered and analyzed, including problems of predicting the sign of increments of random series and processes.

We consider from a unified standpoint various prediction models that are traditionally considered within the framework of various sections of theoretical computer science [6], probabilistic disciplines [6, 22], and Machine learning [3, 4].

Since there have been many reviews recently on the practical use of various professional prediction software [3, 4, 9], this paper does not provide a detailed overview, but only links to sources detailing the relevant software tools.

2 Description and Statement of the Prediction Problem

Although mathematical (probabilistic) models for predicting random sequences, series and processes have a history of more than eighty years [7], intensive research in this areas continue so far. This is due to the huge variety of tasks that are faced when processing large amounts of data. For these tasks, different criteria for prediction accuracy, different data models, different requirements for the time (efficiency) of obtaining a forecast may be required. For example, for many problems of predicting the values of financial indices, the criterion for the minimum standard deviation of the predicted data from the true values (on which the Kolmogorov-Wiener criterion and their numerous linear modifications is based [7]) may well be suitable. But it is obviously not suitable, for example, when it is required to predict the sign of the increment. In some problems, it is quite reasonable to use as a criterion some values averaged over large past samples, in others, local estimates are needed. There are prediction algorithms (predictors) that use numerical (measured) data without any assumptions about their statistical and probabilistic properties (like many algorithms that use the concept of a neural network [3]), and there are implicitly assuming data as samples from a normal stationary process, or as transitions of a Hidden Markov Net (i.e. generated by a Markov source [8]).

In this regard, we fix the following approaches to the problem of prediction present in the literature:

- as one of the mathematical problems of estimating unknown conditional probability distributions ("theoretic-probabilistic approach"),
- as one of the mathematical problem of choosing the optimal solution DM (decision making) approach,
- as an algorithmic control problem.

In principle, the practical application of prediction models requires a combination of these approaches, which in one form or another must be implemented in practice, although in the literature these approaches exist for the most part isolated. In my opinion, this is bad both for practice and for teaching students.

In the Probabilistic approach, the task of prediction is as follows:

Let at the time t, the conditional probabilities $\gamma(x_{t+1}|x_1,x_2,.,x_t)$ (or the probability distribution density $P(x_{t+1}|x_1, \ldots, x_t)$) are estimated for the implementation of the process $x_1,x_2, \ldots,x_t$. In other words, estimates of how likely it is that at time $t + 1$ we will see the value that we predict. It is clear that the more accurate (in the accepted metric) the estimate, the better the forecast. At the next time $t + 1$, we have to estimate the probabilities $\gamma(x_{t+2}|x_1 \ldots x_{t+1})$ (or the density $p(x_{t+2}|x_1, \ldots, x_{t+1})$) and so on. Mathematical model estimates $\gamma(x_{t+1}|x_1,x_2,.,x_t)$, as well as the conditional probability $\gamma()$ itself, is called a "predictor" in the literature [14].

Along with $\gamma()$, predictors are also called software products (for example, some modules of cloud MS AWS Azure ML, Google Cloud ML, etc.) that are used for prediction.

In the future, these software products will be called as previously (Introduction), "Prediction instrument")PI) which may be a subsystem of intelligent decision support (IDS), and the word "predictor" will be used in both senses with the necessary explanations, if this does not directly follow from the context.

In the DM formulation, the following prediction problem is formulated, taking into account the requirement for the efficiency (accuracy) of the forecast: for a given space of strategies B, the space of possible predictable values X, and the loss function $l(b, x)$, $b \in B$, $x \in X$, choose the predicted next value $x_t = b_t$ given the knowledge of past states $x_1,\ldots,x_{t-1}$ (from X), so that the total the loss from the prediction error ("loss") would be asymptotically close to the loss obtained by the best fixed strategy known a posteriori after looking at the entire sequence $x_1,\ldots x_t$. – i.e. a well-chosen (predicted) b_t should minimize possible a posteriori losses.

The strategy is a function that returns an output vector y for each input vector x according to the conditional distribution function $P(y|x)$ (or $\gamma()$) [14, 16].

The third statement of the problem, which we call the "algorithmic model" of prediction, consists in using various conceptual models for obtaining a prediction that do not explicitly contain any basic elements of mathematical models of the above two mathematical formulations of the prediction problem. In terms of content, conceptual models underlie the input language for describing the task of obtaining a forecast (more details below), in particular, the parameters of called functions from certain machine learning libraries [9, 17, 50], which can specify the structure of the predictor used. For example, a search tree in eXtreme Gradient Boosting Regressor aka XGBRegressor, if the user conceptual model considers the forecasting process as a search for an acceptable (by the selected performance criterion) prediction among a set of possible values in a fixed data area, or as a structure of a neural network (number of layers, excitation functions etc.), for example, using the LSTM model. XGB, from the point of view of the user of the prediction tool, is a gradient boosting machine that reads some dataset, applies to it various methods of predicting (not necessarily using any probabilistic models) the future values of the time series, using an iterative procedure to adaptively change the samples from the training data, trying different ways of predicting the results of evaluating misclassified records. This can be seen as incremental addition of prediction methods (referred to as "prediction models") until further improvement is made. These "prediction models" can be developed based on any of the above two classes of mathematical models, but they are not present in the description (input language) of the algorithmic model in any way – from the point of view of XGBRegressor, these are just more or less successful predictors (oracles).

There are dozens of other algorithmic models that operate from the user's point of view as a solution (best prediction) search engine, for example, using Linear Regression, Neural Networks;. Support Vector Machines (SVMs), Artificial Neural Networks, etc.

3 Conceptual Model of the Prediction Problem

In software algorithmic prediction models, as in other machine learning systems, algorithmic models are based on the use of experience gained as a result of solving problems at the training stage. For example, an automatically trained binary classifier is usually understood as a model that represents the characteristics of the classes learned at the training stage [10, 11].

A more or less coherent description of such models can be made on the basis of the idea of a conceptual model.

The Conceptual Model (CM) is a set of notions ("points of view") about the elements, objects (both real and model), and the goals of the modeled system, as well as the relationships between them, expressed in terms of a particular theory. From this point of view, the CM of the prediction algorithm should represent data types and assumptions about their mathematical properties.

Considering that the object of our interest is probabilistic models of prediction, the conceptual model (CM) in the paper means a (formalized or natural) description of the relationships between elements, objects (both real and model) and the goals of mathematical models and algorithms predictions, as well as the relationship between them expressed in probabilistic terms.

Note that in the framework of modern software development methodologies (Software Engineering), a conceptual model is understood as a formalized description of project requirements in one or another data modeling language (for example, UML) [12], and the corresponding language allows a certain typing and structuring of these concepts.

However, this is necessary only if there are information-logical relations between the elements of the system being developed, and if they are not taken into account, it can lead to conflicts during the functioning of the system. An example is the requirement to comply with the temporal order of occurrence of certain events [13].

Since the prediction problem as such (without details of its software implementation) does not require such a description, we will consider textual informal KMs.

The main requirements for a conceptual and mathematical model are that they must provide a representation of patterns and trends in the data that are important to the user, in which case it will be a predictive model for current data to predict what will happen next. In this case, the model should allow the evaluation of actions. Steps to be taken to obtain optimal results.

Therefore, CM should represent the upper level of the description of a software product that performs the prediction of the "future" from known data, which we call the Predictor. Within the framework of the ML paradigm, we consider that the predictor is being trained, and then, based on his training, forms a forecast. Training is on some sequence, and the goal is to tune the parameters of the model implemented by this program to predict the future part of the (not yet observed) sequence. In practice, this may consist not only in tuning the parameters of one particular model, but also in choosing a set, predictor, and/or training samples from a certain region, and the goal is to predict new sequences from the same region as accurately as possible.

The usual approach for this scenario is to develop a theoretical model of the real process (random sequence or process) consisting of identified process steps and possible state transitions with their transition probabilities.

3.1 Requirement for Consistency of CM Components

In order to have a tool for describing the options for using these approaches to solving prediction problems, we define a conceptual model of prediction algorithms, which must be consistent with a mathematical (probabilistic!) model, i.e. provide a representation of all the necessary elements of the subject area of a given prediction problem in a mathematical model of the predictor and it allows a priori assessment of its effectiveness (one or another measure of prediction error, for example). An example (and source) of possible "inconsistency" for the binary sequence predictor model is the fact that according to [14]:

for any predictor, there is a stationary and ergodic source such that the error

$|P(x_{t+1}|x_1 \ \ldots \ x_t) - P'(x_{t+1}|x_1 \ \ldots x_t)|$ does not go to 0 when the length of the observed sequence t goes to infinity (Here P() and P′( ) are the true probabilistic distribution and its estimate respectively).

This means that for any binary sequence $\{x_t\}$ that models this or that data, for any predictor of binary data, it may turn out that the predictor chosen for prediction does not guarantee that the error in estimating the probability distribution of the predicted value $x\ {*}_{t+1}$ with respect to the distribution of the true value x_{t+1} converges to 0 as the length of the training sequence increases.

In this sense, this predictor is inconsistent with the data model as a random sequence $\{x_t\}$ (say, Markovian), and to overcome this inconsistency, the description of the QM must include the concepts of describing the probabilistic model that explicitly or implicitly underlies the software implementation of the predictor. This means that, at a minimum, one should always have a set of predictors, in the hope that among them there will be one for which the specified property does not hold (the difference between the specified distributions will converge to zero relatively quickly), and the conceptual model should include a description of the choice (enumeration) predictors.

Another example of possible inconsistency between data models and predictor mathematical models is the presence of a non-linear relationship between past and future values among the data (for example, self-similarity [15]). If this circumstance, i.e., a possible mathematical model of the data, is not represented in the conceptual model of the proposed prediction method, then there is an obvious uncertainty in the choice of an effective algorithm used in the available toolbox.

So, the conceptual model in our understanding should be the top level description of a software product that performs the prediction of the "future" from known data, which we call the Predictor. Within the framework of the ML paradigm, we consider that the predictor is being trained, and then, based on his training, forms a forecast. Training is on some sequence, and the goal is to tune the parameters of the model implemented by this program to predict the future part of the (not yet observed) sequence. In practice, this may consist not only in tuning the parameters of one particular model, but also in choosing a set, predictor, and/or training samples from a certain region, and the goal is to predict new sequences from the same region as accurately as possible.

3.2 Generalized Representation of the Conceptual Model of the Prediction Problem

We will consider the predictor p of values of the sequence $x_1, \ldots, x_t$ as a function $b_{t+1} = f_p (x_{t\text{-}m}, \ldots, x_t)$, which calculates the value of the random sequence predicted by the predictor f_p at time t, m is the number of terms of the sequence preceding by the time t, which is used in this predictor p to obtain a prediction (b_{t+1} means exactly the estimate of the true value of x_{t+1} and may not coincide with it).

A sequence of random data $x_{t\text{-}m}, \ldots, x_t$ (or a segment of the implementation of a continuous process) can be considered either as a training sample (its part) and / or as input data for some regression model (for example, ARIMA), i.e. as a regression x_{t+1} over $x_{t\text{-}M}, \ldots, x_t$, where $M \leq m$, and for $M < m$ values between $x_{t\text{-}m}$ and $x_{t\text{-}M}$ are used for the training.

Functions f_p belong to one or another family of algorithms (for example, neural networks, gradient, etc.).

We will call the prediction is "probabilistic" if the values at the next moment of time are predicted only with a certain probability (Note that the prediction algorithm, i.e. the method of calculating from known (observed) values) can be either deterministic or probabilistic).

With that said, the conceptual model is presented as:

$$CM = \left\{ M^{\theta}{}_{D}\,(S), M_L, F_{LM} \right\} \tag{1}$$

where $M^{\theta}{}_{D}(S)$ is a prediction model defined by the data type D (binary, real), with a probability measure θ on D, and a structure S on D, which refers to the way data of type D is structured as predictor input variables (for example, splitting data for training and test choices, or scalar or vector data, etc. [9, 11, 50]),

M_L is a model of loss ("PENALTY") from the received prediction with loss function $L(X_S, Y_S)$, where X_S, $Y_S \in D$ observed (X_S) and Y_S predicted data with structure S given by some ratio on $D \times D$ (e.g., "success rate", the ratio of the number of successful predictions to the total number of predictions), related with a measure of predictability [8, 14, 21] and prediction efficiency,

$F_{LM} = \{f_1, .. f_m\}$ is a set of predictors based on the models $M^{\theta}{}_{D}(S)$ and M_L – consisting of predictors in the sense defined above, i.e. methods for estimating the future value with the conditional probability distribution of the predicted value.

We assume that the predictor f_p always corresponds to an admissible prediction model, i.e. all its input data and parameters can be uniquely determined within the framework of the model under consideration at the moment, and therefore are consistent in the sense indicated above (as it was introduced in the Sect. 3.1 for the concept of CM).

This construction (CM) allows representing classes of models according to the types of predicted data, used probability measures, loss functions when making decisions about the acceptability of the received forecast.

Example 1
Let θ be the probability Bernoulli measure on $D = \{0,1\}^n$ – the set of sequences of length $n = 1,2\ldots$ – the number that determines the length of the sequences under consideration. From the point of view (2), n is a parameter of the structure S, for the sequence $X^t = \{x_1, x_{2,}, x_i, \ldots, x_{t\text{-}1}, x_{t|}\}\in D$. Probability $P\ (x_i = 1) = \theta$.

For different structures, a model with a given probability measure will have its own peculiarity (see reasoning about [19] in the 4.1).

Let the loss function is "0/1loss", that is a measure of the loss of the predictor user from prediction errors of the form "zero is predicted instead of the true value" and vice versa, equal to the proportion of incorrect predictions for a sequence of length n, those:

$$e_n = E_p\left(\Sigma_{i=1,n}I(b_t \neq x_t)/n\right),$$

where I() – indicator function, b_t, x_t – predicted and true value, respectively, E_p means distribution averaging with parameter Θ.

One can consider the following predictor $b_t = f(x_{t\text{-}1})$, where $f \in F_{LM}$ and the loss function is defined, known [8] as "the optimal rule for predicting the criterion of minimum loss in measure e_n":

$$b_t = 1\ \text{if}\ \text{Prob}(1) > 1/2 \tag{2}$$

$$b_t = 0\ \text{if}\ \text{Prob}(1) < 1/2$$

and in the case, Prob(0) = Prob(1) = 1/2, the prediction is not performed.

In this case, as it is easy to see, the average losses are equal to the error probability $1\text{-}\Theta$, when $\Theta > 1/2$, and Θ, if 0 is considered "success", which can be expressed as:

$$L_p = \min(\Theta, 1 - \Theta) \tag{3}$$

Instead of a single predictor, we can consider the actions of some subset of F_{ML} predictors.

In this case, the conceptual model for binary data ($D = \{1,0\}^n$) can generate the well-known "multi-expert" prediction scheme, in which binary prediction is viewed as a game between the predictor and the environment, and predictors (whose goal is to minimize expertise loss) that fulfills the prediction.

Each expert F is a sequence of functions F_t: $\{0,1\}^{t\text{-}1} \rightarrow \{0,1\}$, $t \geq 1$, i.e. expertise is a way of setting a probability distribution on a set of sequences $\{0,1\}^{t\text{-}1}$. Each expert defines a forecast strategy as follows: when observing with the first t-1 bits $y_1,.\ y_{t\text{-}1}$ expert F predicts that the next bit of y is 1 with probability $F_t(y^{t\text{-}1})$ [16].

It is easy to see that the content of the conceptual model of this approach does not differ from the previously considered view of prediction as an estimate of the conditional distributions of certain events associated with a random binary sequence (with possible differences in the prediction efficiency estimates used).

Let us now consider what requirements should be met by mathematical models of data in the specific prediction problems, for example, traffic in telecommunication networks, so that it is possible to ensure and control the prediction efficiency in a given cycle (at a given step) of prediction, i.e. to ensure consistency with the properties of the input data and the criterion making a decision about the effectiveness of the considered algorithmic prediction model.

4 Requirements for Mathematical Models of Prediction

As it is easy to see from (1), the first issue that should be taken into account when choosing/developing mathematical models of predictors is the probabilistic measure used in a particular problem, and hence the probabilistic distribution model considered on the data set D, and the measure of prediction accuracy determined by the measure ML (2).

An obvious criterion for forecast accuracy is the distribution of the probability of prediction error (which determines the risk of incorrect prediction), which obviously depends on the probabilistic data model, and, formally, its assessment should be based on knowledge of the probability distributions of the data, and models of the relationship between the distribution of data and the probability of the correct forecast. But usually these distributions are unknown.

From a formal point of view, the unknown distribution of the predicted data means impossibility of a priori estimation of the accuracy (uncertainty) of the prediction with a given prediction algorithm leads either to the requirements for its assessment in the prediction process, or to the use of Bayesian methods [17] of the distribution of the desired conditional probability p(y|x) predicting x from known data y, which is also associated with technical difficulties.

This indicates the need for a certain theoretical model that allows traffic prediction problems to be based not on specific knowledge of distributions, but on knowledge. About general properties of distributions of certain class. However, the ability to overcome this problem depends on the data type D.

4.1 About Binary Data Prediction

This class of problems can be of interest both in itself and in solving problems of predicting the sign of increments of discrete or continuous random processes (see Sect. 5).

So, for discrete data from some finite alphabet A, one of the theoretical approaches to overcome the problem of the lack of exact knowledge of the data distribution is to use the concept of an individual sequence, which is considered as existing in a single instance, and not as a sample trajectory from the ensemble, as is customary in theory. Random processes. It is believed that it is generated by a random source, usually a stationary ergodic one.

A natural question is: can one predict the next values of an individual sequence by estimating a probability distribution based on the past, for example, a specified proportion of correct predictions up to time t, and then minimizing the expected loss in this assignment (i.e., acting as if future events actually happened with an estimated probability). The answer is yes, if you use the so-called. "universal scheme" predictor [14].

A well-known and used example of a universal measure for symbolic (binary, in particular) sequences, which expresses the conditional probability $\mathrm{Prob}(x_{t+1}|x_1, \ldots x_t)$ and is built on the basis of a universal code U, the redundancy of which is asymptotically minimal for classes of Bernoulli and Markov sources [14] (the most famous example is the code obtained as a result of applying the Lempel-Ziv (LZ) compression algorithms [14]).

According to the well-known result [14], let U be a universal code for some set of sources Ω generating letters from the alphabet A, and the measure μ_U for each word v in the alphabet A is given by the equality.

As shown in [14], under the assumption of equiprobable generation of binary strings from the set U by some (hypothetical) data source, the conditional probability $\mathrm{Prob}(x_{t+1}|x_1, \ldots x_t)$ can be expressed:

$$\mu_U(v) = 2^{-|v|} / \sum\nolimits_{u \in Av} 2^{-|U(u)|}$$

where v is the considered string $x_1, \ldots, x_t$ from a given set of binary strings in a given universal code (e.g., LZ), which provides compression as much as their entropy allows, A_v denotes the number of strings $A^{|v|}$.

Then the measure μ_U is universal on Ω, i.e., predicting the next value of any sequence from the set given by sources from Ω, about which there is reason to assume that this is a Markov sequence, can be predicted with some minimum error without knowledge the exact distribution parameters, as:

$$\mathrm{Prob}\,(x_{t+1}|x_1, \ldots x_t) = \mu_U\,(x_1, \ldots, x_{t-1}, x_t)\,/\mu_U\,(x_1, \ldots, x_{t-1})\,.$$

Note that the above LZ algorithm [14], which performs lossless character sequence compression, can at the same time act as a predictor by determining x_{t+1} as the corresponding leaf in the partial match tree [14] with the conditional probability induced by the stepwise algorithm parsing (Recall that one of the well-known criteria for the randomness of a binary sequence is the possibility of its compression (to a value corresponding to the unit of entropy per symbol, which is the randomness criterion [18])).

Let us now clarify the issue of prediction accuracy.

Since the predictor is considered as a conditional distribution, the question of using a universal predictor is reduced to estimating the closeness of the estimated distribution $P(x_{t+1} = a|x_1,..x_t)$ and the conditional distribution γ known at time t (by the predictor) at time t + 1. It is known [14] that an effective measure of the closeness of two distributions is the Kullback-Leibler measure:

$$\mathrm{KL}\left(P\middle\|\gamma\right) = \Sigma_{a\in A} P\left(x_{t+1} = a|x_1, x_2, ., x_t\right) \log\left(P\left(x_{t+1} = a|x_1, x_2, ., x_t\right) / \right.$$

$$\left.\gamma\left(x_{t+1}|x_1, x_2, ., x_t\right)\right).$$

where KL is the divergence measure of the Kullback-Leibler (KL) distributions.

A more practical and simpler example of a universal predictor is the Laplace Predictor $L(x_{t+1} = a|\ x_1\ \ldots x_t)$.

It estimates the conditional probabilities $P(x_{t+1} = a \in A|\ x_1, x_2\ \ldots, x_t)$ from the known values of $x_1, \ldots, x_t$ like:

$$\gamma\left(x_t = a|x_1, \ldots x_{t-1}\right) = \left(n_{x1,\ldots xt-1} + 1\right) / \left(|t| + |A|\right)\Big).$$

where $n_{x1,\ldots xt\text{-}1}$ is the number of occurrences of the letter *a* in the subsequences $x_1, \ldots x_{t\text{-}1.}$

If A = {0,1}:

$$\gamma\left(x_t = 1|x_1, \ldots x_{t-1}\right) = \left(n_1 + 1\right) / \left(M + 2\right),$$

$$\gamma\ \left(x_t = 0|x_1, \ldots x_{t-1}\right) = \left(n_0 + 1\right) / \left(M + 2\right).$$

n_1, n_0 are the number of ones and zeros in the sample of the size $|t| = M$.

It is important to note that the predicted probabilities cannot be equal to zero even through a certain letter did not occur in the word $x_{1,}\ \ldots, x_{t-1,}x_t$.

For example, for the sequence 01010, the Laplace predictor:

$$\gamma\left(1|01010\right) = 3/7 \text{ for } A = \{0, 1\}.$$

It was shown in [9] that for any source with probabilistic distribution P that generates independent and identically distributed symbols from the alphabet A, Laplace predictor error satisfies the inequality:

$$\mathrm{KL}\left(P, L\right) \leq \log\left(e\left(|A| - 1\right) / \left(t + 1\right)\right).$$

$$\left(e = 2.718\ldots \text{ is the Euler number}\right).$$

As it is not hard to see, Laplace predictor is universal as its considers prediction as a set of estimations of unknown (conditional) probabilities, and the average error of the Laplace predictor (estimated either by the KL divergence or the variation distance) goes to zero for any unknown i.i.d. source, when the sample size t grows. Moreover, it can be easily shown that the error (and the corresponding variation distance) goes to zero with probability 1, when t goes to infinity. Obviously, such a property is very desirable for any predictor and for larger classes of sources, like Markov, stationary and ergodic, etc.

We see that the error of the Laplace predictor tends to zero for any source that generates independent and equally distributed symbols (i.e. does not depend on the probability distributions of symbols (the probability of occurrence of symbols), but requires certain (theoretically) probabilistically-conditioned constraints).

Unfortunately, there is no predictor that has this property for any stationary ergodic source [6, 8]). But the universal predictor γ, for any stationary ergodic source ω generating letters from some finite alphabet A, ensures that another measure of error tends to zero, namely, the Cesàro mean of errors:

$$\lim_{t\to\infty}\left(\Sigma_{t=1,s}\mathrm{KL}(\omega,\gamma)_t\right)/t=0.$$

In other words, we can talk about the universality according to Cesare with such a measure of error, the predictor γ is universal for the set of sources Ω if for it this expression = 0 for any $\omega \in \Omega$. That is, if we know to which of the wide class of (stationary ergodic) sources (generating data) can belong data, then we can, given this equality to conclude that for any source $\omega \in \Omega$, the value of the predictor $\gamma(x_1, \ldots, x_t)$ approaches the probability $\omega(x_1 \ldots x_t)$, which means that the predictor is universal (in the sense of averaging over the divergence KL on the interval of length t).

From this equality, we can conclude that, in a certain sense, the universal measure μ is a nonparametric estimate for an unknown conditional probability distribution P.

Thus, if the KL-Cesaro estimate is relevant to the prediction problem under consideration (for example, there is a monotonic relationship between $\gamma()$ and some predictive quality criterion, for example, in terms of the loss function), then θ in the representation $\{M^{\theta}{}_{D}(S), M_L, F_{LM}\}$, we can consider as an arbitrary stationary ergodic measure.

As it shown in [14], whatever the actual data generation mechanism, using a universal approach (more precisely, estimating the quality of a prediction based on it) does not perform much worse than any other possible forecasting method that uses knowledge of the probability distributions of data.

Thus, using the results of the universal predictor theory, one can abstract from the parameters of specific distributions in the course of prediction.

To better understand the implications of this result, let's ask, does additional information about the probabilistic properties of the data always improve the prediction score?

A negative answer follows, for example, from the results of [19]. In [19] was proved that for any finite i.i.d. a sequence of binary data in which each outcome “success “or failure” (0 or 1) the conditional expectation of the proportion of successes among the results that immediately follow a series of consecutive successes will be strictly less than the corresponding conditional probability of success (from which one determines expectation – depends on the occurrence of at least one series of k consecutive successes within the first n – 1 trials, where n > 3 and 1 < k < n – 1).

Thus, the choice of the structure S, so that it includes k previous values of the binary sequence, in this case, can lead to a worse prediction. It all depends on the prediction model.

Correspondingly, attempts to use knowledge about the probabilistic properties of binary sequences do not always lead to better forecasts, and the resulting forecast depends on the accepted criterion (“optimal Bernoulli”, in the considered case (2)).

This suggests that additional information does not always lead to an increase in the a priori probability of a correct forecast; here we are talking about a priori probability, because explicitly a posteriori probability when choosing a prediction method (or predictor) is not present – it can only be estimated indirectly, based on the results of past work.

Let us consider how criteria are manipulated in the modern practice of MO-based predictions to ensure independence from the knowledge of distributions.

4.2 Performance Criterion Not Related to Data Distributions

As the analysis of the literature shows, another way to avoid the need to evaluate the exact laws of distributions is to use the criterion of empirical risk minimization (ERM) [20, 51].

The main idea behind ERM is that we cannot know the true “risk” when running on a particular dataset because we don’t know the true distribution of the data the algorithm will operate on, but instead we can measure its performance on an already known dataset training data (“empirical” risk), and minimize it. Cloud computing traffic prediction papers [20] show that the principle of risk minimization used by time series prediction algorithms affects the accuracy of algorithms in different ways in environments with different traffic models. ERM is rated as:

$$b_t = \arg_{h\in H}\ \min\left(R_{emp}\,(h)\right)$$

$$R_{emp}\,(h) = \left(\Sigma_{i=1,n} L\left(h\,(x)\,,\,x'\right)\right)/n$$

where L(h(x),x’) is a loss function that measures how much the prediction value x’ = b_t under hypothesis h∈H differs from the true value x = x_t.

In other words, R_{emp} is an estimate of the average loss calculated from n previous observations. As shown in [21] the proportion of correct predictions averaged over the number of observations is the estimate of the predictor $\gamma()$, and (as it is easy to show) this estimate coincides with R_{emp} for the specified "0/1loss" with Hamming-type loss function L(a, b):

$$L(a, b) = 1, \text{ if } a \neq b, \text{ and } L(a, b) = 0 \text{ in the case of } a = b,$$

where a, b are true and predicted values respectively.

However, no assumptions are made about the distribution of the predicted data.

So, for symbolic/binary random sequences that could be considered as stationary ergodic, there are no characteristics other than ordinary frequencies (Laplace, Bernoulli optimal predictor (2)) that significantly affect the prediction efficiency.

4.3 Influence of Probabilistic Properties of Random Time Series on Prediction Efficiency

The concept of probabilistic causality, introduced in the context of random processes homogeneous in time, can be used to determine the similarity relation on stochastic processes.

Examples of Concepts in the Theory of Random Processes with Real Values Affecting the Efficiency of Prediction

Let we have two random processes X(t) and Y(t) defined on an ordered set real domain R with associated probability functions P and Q on the same result set. We say that the two processes are causally similar [52] if:

$$P(x(t)\,|x(t-a)) = Q(x(t)\,|x(t-a)), \forall t \text{ and } \forall a > 0.$$

where "|" as previously means conditional probability.

It is obvious that all processes that are homogeneous in time and transferred in time are causally determined.

Intuitively, this is true for a closed physical system, but not for an open system, since in this second case other external variables may influence dynamic evolution. If two processes are homogeneous in time, Markovian and jointly of the same transition matrix, it is also easy to show that they are causally similar.

Finally, if two Markov processes (not necessarily time-homogeneous) share the same transition matrix at the same time step, then they are also causally similar.

From the point of view of the prediction problem, this means that the conditional distribution of the predictor γ () for a particular data set can be determined by different mathematical models, which will be considered below.

Other important concepts in prediction theory are dependency models of past and future values of data (linear or non-linear models, correlation structures) and stationarity of data in the sense that some (usually hypothetical) laws governing the change in data over time of observation (and associated change in events) remain unchanged, at least during the collection of the history, on which the forecast must be made.

For example, for highly correlated time series (including the case of binary sequences), linear regression models (ARIMA, such as [3]) give a better prediction than if they are weakly correlated.

In other words, if, when considering the model properties of the time series, we restrict ourselves only to the correlation properties of the sequence, and the conditional forecast probability $\gamma(x_{t+1}|x_1,x_2,.,x_t)$ considered above is clearly related to correlation (as is the case, for example, for Gaussian distributions, when correlation is equivalent to independence), then according to the degree of correlation we can expect one or another quality of prediction by a linear regression predictor.

However, as a study of the literature has shown, many examples of data, for example, telecommunication network traffic, considered as a random sequence (time series), in general, have more complex and subtle properties from the point of view of the theory of random processes, and, accordingly, they have more complex of prediction models.

For example, the traffic values at the predicted moment may depend on events significantly remote in time (LRD – Long-range dependence, see below) and it is natural to assume that LRD traffic prediction will be quite effective (since information about the past clearly affects the predicted future). Moreover, the trajectories (realizations) random processes with such properties have "self-similarity", i.e. repeatability of patterns on different time scales [23].

4.4 *Self-Similarity as a Nonlinear Relationship Between the Past and the Future*

In fact, the self-similarity property reflects the nonlinear relationship between the past and the future.

The fact is that, for example, the self-similar behavior of traffic, and its long-term correlation, although used as the basis for forecasting] [, but on the condition that the predictor should not respond to short-term traffic changes. However, there are situations where this short-term data is important, for example, at the beginning of a DDoS attack [24].

At the same time, self-similarity means a nonlinear relationship between the present and the future, since the repeatability of a form on different scales can not be expressed by a linear transformation, since the transformation in time is associated with the appearance of new harmonics in the spectral representation of corresponding random process.

If in the discrete spectrum of a random process (time series) x(t) there is only one spectral frequency harmonic ω, and at some future moment x(t + k) 2ω appears (a "faster" component), then this can only be expressed with using non-linear transformation of values x(t + k − m), ... x (t + k − 1), m < k. This circumstance can also make it inefficient to use a very large volume of observations.

From a more general point of view, self-similarity is a term from fractal theory [25], which describes objects that visually look the same regardless of scale, which is expressed in the fact that local signal patterns are repeated many times in a whole time series on different (usually small) time scales, so the original set can be reproduced from its smaller portion at suitable magnification.(Intuitively, this property means the presence of a past-future connection structure that, from an intuitive point of view, could increase the possibility of correct prediction). For example, in a number of methods [24], self-similarity destruction is a predictive feature – the methods are based on the observation that the presence of a DDoS attack reduces the degree of self-similarity of normal traffic, since DDoS tools do not generate self-similar traffic, and this is reflected in the traffic.

In the practice of telecommunications networks, self-similarity can be caused by so-called "Elephant" connections [27], generating a continuous stream with an extremely large total size of bytes created by, for example, a TCP stream or other network channel protocols. These streams can use extremal share of the total bandwidth over a period of time. Thus, it is they who will determine the state of traffic during this period of time (say, the volume of transmitted packets per second, or packet delay), and it is this period that affects the structure of traffic in this time.

Note that self-similarity is also called scale invariant under the transformation x = bx, y = ay, if a curve F(x) is scale invariant under the transformation [28, 45]:

$$\mathrm{F}\,(\mathrm{bx}) = \mathrm{aF}\,(\mathrm{x}) \equiv \mathrm{b}^{\mathrm{H}}\mathrm{F}\,(\mathrm{x})\,.$$

where the exponent H = log(a)/log(b) is called the Hurst exponent (also known as Hurst parameter or simply H). The Hurst exponent H determines the time separating correlated (for example, stronger than a certain threshold value) of random process (e.g., traffic) samples from each other (more details below).

Hurst exponent:

$$\mathrm{E}\,(\mathrm{R}\,(\mathrm{n})\,/\mathrm{S}\,(\mathrm{n})) = \mathrm{Cn}^{\mathrm{H}}, \mathrm{n} \to \infty.$$

R(n) is the range of the first n cumulative deviations from the mean S(n) is the series (sum) of the first n standard deviations E(n) is the expected value n is the time span of the observation (number of data points in a time series), C is a constant.

Practically, in nature, there is no limit to time, and thus H is non-deterministic as it may only be estimated based on the observed data. The calculation of H is considered in detail in [28].

This value shows the degree of presence of self-similarity in the time series, since its value is the greater, the smaller the time shift between the same pairs of values in this time series.

(The greater the delay between two identical pairs of values in the time series, the smaller the Hurst coefficient.)

The presence of self-similarity properties raises the question of the possibility of using it to increase the efficiency of predictors.

Therefore, let us briefly consider well-known approaches to mathematical models of time series with self-similarity.

1. Poisson models

Speaking about the possibility of using the Poisson flow model as a means of modeling self-similarity, we note that in [29] it is shown that, for example, modeling network traffic with the Poisson model, assuming that the packet length will tend to smooth out by averaging over a long time scale, may be incorrect, precisely because network traffic exhibits a long-term dependence. At the same time, uniform sampling from heavy-tailed distributions can produce poor estimates, since a relatively small number of samples can seriously affect the final estimates.

The slow decay of the variance of a random process (e.g., in the number of arriving packets) as the scale of self-similar traffic increases is in stark contrast to the mathematical structure provided by the Poisson simulation, in which the variance of the arrival process decays as the square root of the scale measure.

Therefore, less traditional models and properties of random process consider for the series with self-similarity.

2. Long Range Distance model

If a random process, for example, network traffic, can have the property of a long-range (extended in time) dependence [23], in which the correlation is preserved between events sufficiently remote in time corresponding to changes, then one speaks of a Long Range Distance (LRD) model.

From the point of view of correlations, the time series X_t, $t \in Z$, is called long-range dependent if its covariance function [25, 26, 27, 28]:

$$\gamma\ (t) = E\,(X_0 - EX_0)\,(X_t - EX_t) \sim c\ |\ t\ |^{2-2H}{}_{,}\ t \to \infty,$$

$H \in [1/2,1]$ – the Hurst exponent.

In spectral terms, the presence of LRD properties can be represented as:

$$f(\lambda) \sim c|\lambda|^{2H\text{-}1}\ \text{при}\ \lambda \rightarrow 0$$

где $f(\lambda)$ is the spectral density X_t,

$c > 0$ is a constant.

The larger H, the stronger the time dependence, because the covariance function decays more slowly at infinity (i.e., the correlation of events separated in time is preserved at large H), i.e. the decay of the values of the autocorrelation function is much slower than the exponential convergence typical of (short-range) classical models such as autoregressive moving averages (i.e. if the data is represented by a process with autocorrelation properties).

In the case of processes without LRD, there is no dependence ($H = 0.5$) and its autocorrelation $r_k = 0$ for the lags $k \geq 1$.

The time series describing the traffic becomes self-similar due to the simultaneous influence of many sources (for example, as in the above [24]), and the aggregate multi-level source traffic (throughput traffic) with a distribution of latency with heavy tails for the time interval in which the source is active or inactive, can be approximated by Fractional Brownian Motion (FBM) as shown in [25, 27, 45]. Therefore, FBM is considered as a natural tool for modeling the phenomenon of self-similarity.

3. Fractional Brownian motion model

Fractional Brownian Motion (FBM) with the Hurst parameter H is a continuous Gaussian process with an autocorrelation function. At the same time, for H > ½, which, formally, means the LRD-property. But in contrast to the classical Brownian motion, the increments FBM (of the difference process) do not have to be independent, and knowledge of the law of this dependence (for example, the conditional distributions indicated above) in some cases could improve the prediction (as prediction of a process with known prediction) with continuous time on [0, T] that has zero (conditional!) expectation for all t in [0, T] (see the possible use of this property for prediction in Sect. 5 of this paper).

Note that in terms of using process properties for prediction, a Brownian Motion (BM) without "fractionality" is a motion in which the process value changes with random increments over time, and the prediction is determined by this randomness.

At the same in this process there is a certain "memory", which means the dependence between the past and the future.

From a formal point of view, BM is the integral of white noise. These motions define paths that are random but (statistically) self-similar, i.e. the approximate trajectory (section) of the particle's motion ("outlining" the implementation of a random process) resembles the entire path, and this is an intuitively understandable possibility of prediction. But this requires a model that contains one or another description of the structure of fractals, and not just a probabilistic model of increments.

Section 5 will show how the measurable properties of the autocorrelation function can be used to make a forecast.

So, the Hurst exponent can clearly indicate whether the time series has the property of a pure random walk, or has some correlation structure [22, 25, 29]. For example, in the field of Internet of Things [31] the prediction model may include a parameter for network traffic as a priori knowledge. Since the self-similarity property is well interpreted in terms of describing traffic by IT specialists, its use in conjunction with the features of the deep network increases the interpretability of the model, namely, the ability to understand how data properties, in particular self-similarity, affect the quality of the forecast.

4.5 *Stationarity as a Property of a Random Process Affecting Predictability*

Let us now consider another important property of the data modeled by a random process, namely, the stationarity.

It is intuitively clear that the stationarity of a random process in the narrow sense is more favorable for forecasting than non-stationarity, if only because the joint probability distribution $W(x_{t+1,}x_1, \ldots x_t)$, which determines the conditional probability of the predictor $\gamma(x_{t+1|}x_1, \ldots x_t)$, does not depend on time.

Most of the considered real processes, however, are not strictly speaking stationary.

For example, this in most cases concerns network traffic, where non-stationarity can be associated with its hopping over a wide range of time scales, and, many publications show [25, 27] that network traffic does not in principle have stationary behavior, also due to the presence of time cycles (daily, weekly) and can be easily affected by network reconfigurations (for example, manual reconfiguration, dynamic routing changes), communication failures and the deployment of new machines and applications.

Obviously, for similar reasons, time series describing other real processes, such as electricity consumption, etc., can also behave.

In other words, one can speak confidently about stationarity only to a certain extent.

To estimate this “confidence degree”, statistical tests for stationarity are well known [32], however, it is difficult to include their results in the prediction model, since apart from intuitive qualitative arguments that stationary processes are better predicted than non-stationary ones, nothing can be used in the model.

In [31], predictors are compared based on short-term correlations, which is typical for problems of estimation, classification, and prediction of non-stationary processes, and it is investigated whether it is useful to include a long-range dependence in the prediction model, which, as noted, is associated with the self-similarity property. The conclusion is that, first of all, short-term correlations dominate the contribution to predictor performance, i.e. time to calculate satisfactory, in terms of the applied criterion, predicted values. As a consequence, linear prediction with a relatively short correlation structure is sufficient for prediction applications, rather than the long-term correlations that exist between the future value of the time series (e.g., traffic level and remembered history)). This is the case, for example, when there are very many short connections, sometimes called ‘mice’ flows [27].

In [28] it is shown that relatively short observations can be considered as stationary, at least in variance, because under self-similarity, the variance of the sample mean decreases more slowly than the reciprocal of the sample $X^{(m)}$ size m (slowly decay variances) $Var(X^{(m)}) \sim a_2 m^{-\beta}$ with $() < \beta < 1$, a_2 is a positive constant.

It is also significant that autocorrelations decay as hyperbolic functions, and not exponentially fast.

Accordingly, with a certain degree of conventionality, one can also speak of stationarity in terms of autocorrelation functions.

Hurst Exponent as a Measure of Stationarity In [31, 33, 34] it is shown that the Hurst exponent H can act as a stationarity criterion for the self-similar process.

Let us point out its connection with the possible conventional characteristics of mathematical statistics and the theory of random processes, and, accordingly, with predictability.

Values $H > 1$ indicate non-stationarity.

For a stationary self-similar process $H \in (0.5.1)$. The closer the value of the Hurst parameter is to 1, the slower the dispersion decays as the time scale increases, and the traffic is said to become more pulsating, and therefore non-stationary.

Let us consider how these properties of LRD (self-similar) series affect the technique for solving various prediction problems. If the task is to predict the following values according to the root-mean-square criterion, and there is reason to consider the series stationary according to H), then it makes sense to consider (compare with) the Kolmogorov-Wiener optimal prediction criterion.

First of all, it is known that the root-mean-square error of linear prediction for the simplest known linear forecast for a stationary time series x(t) is [35]:

$$\varepsilon = \left(1 - \rho^2\right) \mathrm{var}(x)\,,$$

where the correlation coefficient is $\rho = ACF(1)$,

ACF(1) means the autocorrelation function in the lag 1,

where the error corresponds to the minimum mean square error (Mean Square Error-MSE):

$$\sigma^2 = E(x\,(t + m) - X_e)^2$$

where X_e is the predicted value, for example, by the Kolmogorov linear predictor.

$$X_e = \Sigma_{i=1,n}\, a_i x\,(t - i)\,,$$

coefficients a_i are the objects of the Kolmogorov-Wiener optimal linear prediction task.

Therefore, given that for the LRD process the autocorrelation function [22, 45]:

$$AC_H\,(k) = 1/2\left((k + 1)^{2H} - 2k^{2H} + (k - 1)^{2H}\right)$$

where k is the lag number, there is a monotonic relationship between the root-mean-square error of linear prediction epsilon and the Hurst exponent in the interval [0.5–1].

This analysis can be useful when using predictive software tools such as the ARMA moving average autoregressive model, the AR autoregressive model, the moving average MA model, and the ARIMA autoregressive integrated moving average; see, for example in [36].

In the case of H > 1, prediction methods for stationary series become inefficient, unexpected bursts with a magnitude much larger than expected from traditional models, which is consistent with non-stationarity.

It is important, however, that when the initial process B is nonstationary, for the increment, the fractional Brownian motion has stationary and dependent increments. The last expression shows that the increments are positively correlated if $H \in (1/2, 1)$, uncorrelated if $H = 1/2$, and negatively correlated if $H \in (0, 1/2)$.

This property turns out to be important for predicting the sign of increments, and will be discussed in Sect. 5.

Note, that most statistical forecasting methods are based on the assumption that time series can be made approximately stationary (i.e. “stationary”) by going over the difference in data over time, so that instead of directly considering index, we calculate the difference between successive time steps.

However, the ability to predict data with a time difference, rather than the data directly, is a much more significant indicator of the model’s predictive power.

Indeed, the prediction of a pure random walk is impossible in principle, but success rate SR > 0.5 may appear simply due to a slight change in neighboring values and an accuracy criterion that is insensitive to these changes (in a large percentage of observed cases).

For a stationary process, the MSE linear prediction theory based on the Kolmogorov-Wiener model assumes that the time series has a finite mean and variance.

However, for time series with LRD properties, this cannot always be done. The fact is that many real data streams formed as a result of overlaying data from several sources have self-similarity and LRD-property, and at the same time are distributed according to Pareto [24].

4.6 Influence of Probability Distributions of Processes with LRD on Prediction

For systems with $H \geq 0.5$, a Gaussian probability distribution function cannot be used as a characteristic pdf. One should therefore look for an alternative distribution function to characterize these systems. Numerous studies of LRD series have shown that their distributions are close to Pareto [37]:

$$P_{Par}\ (X) = ab^a / x^{a+1}$$

where $x \geq a$. The mean and variance of x_t that follows are, respectively, given by:

$$\mu_{\text{Par}}\ (\text{x}) = \text{ab}/\ (\text{a}-1)$$

$$\text{Var}_{\text{Par}}\ (\text{x}) = \text{ab}^2/(a\text{–}1)^2\ \ (a\text{–}2)$$

$$\text{x} \geq \text{a} > 0, \text{b} > 0$$

It can be easily seen that μ_{Par} and Var_{Par} do not exist if $a = 1, 2$.

If it obeys the Pareto law, then the above expression approaches infinity for a = 2, no matter how large the estimation interval is. Therefore, the use of Wiener-Kolmogorov predictors to conclude on the predictability of the LRD series relative to the usual MSE is unacceptable.

Reference [37] shows the relationship between H and the parameters of the Pareto distribution, which can provide finite values of expectation and variance, and hence the possibility of linear prediction by the MSE criterion

For an LRD series with finite variance, the covariance slowly decreases to 0 as a power function. Such time series can be called LRD time series with finite variance.

In this case, the distributions of the LRD values of the series can have heavy tails [23]. shows that if the heavy-tailed distribution of a time series distribution with LRD does not allow an acceptable estimate of mean and variance, the MSE generalization can be used with Kolmogorov-Wiener predictor in a linear combination of past observed values.

There are examples where a stochastic process shows heavy tails in the domain of a probability distribution, [38], but the tail parameters can be used to represent correlation functions of LRD processes with infinite variance.

Let us consider how these properties of LRD (self-similar) traffic affect the technique for solving various prediction problems. If the task is to predict the next values by the root-mean-square criterion, then it makes sense to compare with the Kolmogorov-Wiener criterion as the basis of most linear regression methods.

Note that considering that for many problems of traffic prediction, the prediction of increments can be a practically important task, the fact of their stationarity can be effectively used, without searching for the optimal root-mean-square solution (Sect. 5).

Let us now briefly consider the specifics of using nonlinear prediction models with regard for time series with the considered properties.

4.7 Nonlinear Models and Neural Algorithms

A non-linear prediction model in modern IPs can be implemented either in non-linear regression models or in artificial neural networks [39], for example, in the deep learning models such as convolutional neural networks (CNNs), Graph Convolutional Network (GCN) [40] and recurrent neural networks (RNN) [41], in addition to machine learning algorithms such as Support Vector Regression (SVR) [41, 42].

However, the neural network can have problems due to overfitting [4], when the model explains well only the examples from the training set, adapting to the training examples, instead of learning to classify the examples that did not participate in the training. It is easy to see that self-similarity can contribute to this phenomenon. At the same time, such a common way to reduce overfitting as Dropout – turning off some neurons with a certain probability on some data interval from the training process may not work due to the fact that training will be similar to the previous one.

Also, possible sudden changes in cloud traffic can be easily confused by a neural network with traffic anomalies, which leads to training inefficiency.

These sudden changes can be interpreted as non-stationarity by calculating the values of H.

Another non-linear time series forecasting technique that is being tried for traffic forecasting is support vector regression (SVR) [39], which is based on structural risk minimization. However, the choice of suitable kernel functions and optimal parameters is very difficult [43]. Examples are briefly discussed in [44].

From the above analysis of the dependence of the efficiency of using prediction models on specific properties of time series, it follows that the natural way to take into account the dependence of predictor properties on data behavior (stationarity, nonlinearity of dependence) is their online selection in the process of solving the network control problem.

This is confirmed in many publications. For example, in [47] it was concluded that for predicting the response time and throughput of cloud services, at different stages of resource use, artificial neural network and linear regression algorithms have different efficiency. In other words, the overall prediction accuracy can be improved by combining different prediction algorithms. However, due to the computational complexity of prediction algorithms, the computational complexity of choosing search tree prediction tools may be unacceptable for the problems of using prediction in online network management (for examples when using popular predictors, see Sect. 5).

It is important, that the data represented by time series is different from other data models in the sense that it gives us additional information about the time of occurrence of events that can be used when building a machine learning model. For example, if the time series is highly correlated in time, then its value at time “t + 1” is likely to be close to the value at time “t”, and the model actually does, one that when predicting the value at time “t + 1”, it simply uses the value at time “t” as its prediction.

However, if we are talking about predicting a change in the sign of the increment, then this closeness does not give anything significant.

Therefore, let us consider separately the problem of predicting the sign of the increments of random time series and processes.

5 Properties of Probability Distributions as Characteristics of the Predictability of the Sign of Increments of Random Time Series and Processes

An example of the usefulness of predicting the sign of traffic increments is, for example, when attacks on a network are carried out in a way that cannot be detected by antivirus software, and analysts have to rely on analyzing changes in network traffic (traffic volume), or the direction of changes in file write intensities [46].

In some cases, when analyzing traffic, trend analysis is used not only because of the usefulness of this characteristic itself, but also due to the fact that in this case the forecast is more accurate than for the absolute values of the predicted traffic characteristics (for example, traffic volume per unit time) [47].

Consider the problem of predicting the trend sign of the time series $\{x_i\}$ by the incremental sequence $\Delta_{t+1} = x_{t+1} - x_t$. In this case, Δ_{t+1} is a centered value, with a sample mean close to zero for finite segments x. As is known from the theory of random processes, the correlation between successive values of Δ_{t+1} turns out to be much weaker [22] than in the original sequence $\{x_t\}$. At first glance, this may indicate a worse predictability of increments compared to the predictability of the original x_t values. This, however, concerns the values of the increments, not their sign. Moreover, since the sign of the increment means the direction of change, the sign of the residual autocorrelation of neighboring values is an important characteristic, since from a practical point of view, a negative sign of autocorrelation ("negative correlation") means a tendency to a multidirectional trend in neighboring sections of the sequence, which, obviously, can be used to predict the sign of the change. Further, we will show that these intuitive assumptions have a certain mathematical justification in the theory of random processes [48].

Definition 1 The change in the sign of the increment of values in random data is predictable (and the sign of the increment is predictable) if $E(\text{sign}(\Delta_{t+1})/F_t) \neq E(\text{sign}(\Delta_{t+1}))$, where E is the sign of the expectation (according to the distribution of values, in this case Δ_t), F_t is a part of the probability space on which the random process is considered, which corresponds to the observed values x_t, or $x_{t\text{-}k}, \ldots, x_t$, or, say, some sample estimate of the conditional mean x_{t+1}, etc. (from a formal point of view, F_t corresponds to the so-called "filtering" in the theory of random processes [30]).

In other words, a sign change is predictable if the probability of its current value ("−" or "+") changes when certain previous events F_t change.

In terms of the forecasting approach adopted in mathematical statistics, this means that we can express, say, the probability of a positive change as:

$$\Pr(\Delta_{t+1} > 0|F_t) = E(I(\Delta_{t+1} > 0)|F_t),$$

where I() is the indicator function.

It is easy to show [48] if the conditional distribution $D(\Delta_{t+1}|F_t)$ can be approximated by the normal distribution $N(a, \sigma^2(\Delta_{t+1}|t)$, where:

$$\sigma^2\,(\Delta_{t+1}|F_t) = E(\Delta - E\,(\Delta))^2,$$

$$a = E\,(\Delta)\,,$$

$\Delta = \{\Delta_{1,..}\ \Delta_t,\ \Delta_{t+1}\}$ are increments over the entire area of consideration of the random sequence $X = \{x_{t\text{-}k}, \ldots, x_{t,}..\}$,

$\sigma^2(\Delta_{t+1}|F_t) = E(\Delta - E(\Delta))^2$ is conditional variance of increments in time in which events occur, in particular, over the entire time interval $(t - k, \ldots, t)$ in which values $x_{t\text{-}k}, \ldots, x_t$, then the probability of a positive sign:

$$\Pr\,(\Delta_{t+1} > 0) = \Phi\,(a, \sigma\,(\Delta_{t+1}|F_t))\,,$$

where Φ is the standard normal distribution function.

It is clear from the above that a zero mean would make the sign unpredictable (at least for a normal increment distribution) in the sense of the definition given above, since variations "up" and "down" with any variance about zero mean for a symmetric distribution occur equiprobably.

For an arbitrary distribution D, it can be obtained that the minimum estimate of the prediction error relative to the true sign of Δ_{t+1} of the assumed "loss function":

Loss $(\Delta_{t+1},\ \Delta'_{t+1}) = E_t(I(\Delta_{t+1} > 0) - \Delta'_{t+1|t})^2$ is achieved by estimating:

$$\Delta'_{t+1|t} = E\,(I\,(\Delta_{t+1} > 0|\Omega_t)) = P\,(\Delta_{t+1} > 0|\Omega_t) = 1 - F\left(-a_{t+1|t}/\sigma_{t+1|t}\right)$$

where, we repeat, the conditional expression $(\ldots|t)$ means the calculation of the observed values up to the moment t inclusive.

Therefore, the dynamics of changes in variations ("volatility", to use the terminology of financial mathematics) will affect the sign forecast in all cases when the conditional mean is not equal to zero. Then the sign of the increment is predictable, even if the conditional mean is unpredictable at zero mean.

If the distribution of F is skewed, then the sign can be predicted even if the mean is zero: in this case, the time-varying skewness can be a determining factor in predicting the sign.

But for a non-zero conditional mean $(a > 0)$, even if the distribution is symmetric about the conditional mean and the conditional mean is constant by assumption (unlike the t-dependent variation of $\sigma(t + 1|t)$, the sign of the increment is predictable by the above definition.

So, conclusions about the predictability of a feature can only be based on knowledge of the symmetry of the law of distribution of time series, modeled by a random process with this law, and on the distribution parameters, namely, the mean, variance, skewness, etc.

5.1 Binary Prediction Model of Sign

The considered approach to determining the predictability of the sign of increments has a certain connection with another sign prediction paradigm based on the consideration of a binary sequence corresponding to the signs. For example, it is natural to assume that the value of the new binary sequence is 0 if x_t-x_{t-1} $\leq$ 0 and 1 in the case of a positive increment, and use this binary sequence to predict the corresponding statistics of the sign change sequence. At the same time, it is obvious that this statistic corresponds to the statistic of the change in signs of the increment.

In the most general form, the concept of predictability for binary sequences can be represented as follows.

Definition 2 A binary sequence x_1,x_2,... from some distribution Z is predictable for a predictor implementing some polynomial algorithm A if for each 1 < i < n and any polynomial algorithm for estimating the next value rejects any statistical test for the inequality [32]:

$$| \mathrm{Prob}(A(x_1, x_2, ..x_{i-1})) = x_i) - 1/2 | \leq O(\nu(n))$$

where A($s^{i-1}{}_1$) is an event consisting in observing a segment of the sequence up to the moment i which consists in observing a segment of the sequence up to the moment i, is the predicted value, O(ν(n)) denotes a function decreasing faster than any polynomial in n.

It is easy to see that in this conceptual model, rule (2) with loss function (3) can easily be used as a predictor [21].

5.2 On the Connection Between the Properties of Autocorrelation Functions of Random Processes and the Probabilities of Changing Signs of Increments

Consider the approach to sign prediction as a nonlinear short-term traffic prediction.

It is shown in [49] that the sign of the difference between neighboring observations x_1,x_2,..., can be correctly predicted with a probability >1/2 despite the well-known fact of decorrelation of independent increments (Sect. 2). Consider a sequence of centered random variables y = {y_i = x_i – (Ex_i)}, i = 1, . . . ,t.. with zero mean and probability density P(y).

If we assume that the increments y_i are independent, then the conditional expectation of its increments y_{i+1}− y_i at the last observed value of y_i is:

$$E\,(y_{i+1} - y_i|y_i) = E\,(y_{i+1}|y_i) - E\,(y_i\ |y_i) = -y_i$$

which follows from the fact that $E(y_i|y_i) = y_i$, $E(y_{i+1}|yi) = 0$ with zero expectation, and all increments $\{y_i\}$ are independent.

Accordingly, considering the conditional mean as a sign predictor, the following rule is proposed for predicting the sign of the difference $y_{i+1} - y_i$:

$$\text{sign}\,(y_{i+1} - y_i) = \quad -\text{sign}\,(y_i) \tag{4}$$

In [49], it is stated that uncorrelated increments are sufficient to fulfill the indicated sign relation. Although uncorrelated does not always mean independence, for network traffic this assumption can be accepted. Indeed, in real processes, the dependence of time-sequential values often takes place due to a non-stationary trend, which is eliminated by centering $y_i \equiv x_i - Ex_i$, and in addition, most real data, such as changes in traffic volumes, changes in the number of requested IP addresses, etc. It somehow can be connected with a random change (decrease-increase) of some external factors [50], and can be represented by random walk processes close to processes with independent increments [22, 30].

Regarding the prediction accuracy according to the rule (4) (called in the [44] as SC ("Sign criterion-based") predictor), we can give more subtle mathematical reasoning [42].

The share of successful predictions according to (4) for sufficiently long sequences is proposed to be estimated as [49]:

$$R = \frac{1}{2} + (1 - F(0))\,F(0)$$

where:

$$\mathrm{F}\,(\mathrm{x}) = \int_{-\infty}^{x} dy \mathrm{P}\,(\mathrm{y})$$

$$y = y_{i+1} - y_{i,};$$

P() is a probability distribution function (pdf), F(0) obviously represents the probability of increments being negative and 1-F(0) positive.

It is easy to see that R reaches its maximum at F(0) =1/2, i.e. with a symmetrical distribution of the increment (difference) of the initial random time series.

These formulas mathematically express, at first glance, a paradoxical result, that when guessing the sign of the increment of independent random variables from the previous value of the centered process, you can get the probability of success more significantly more than ½!

This fact can also be given some elementary probabilistic justification.

Indeed, the truth of (4) depends on a combination of three conditions for the ratios between y_{t+1}, y_t (i.e., we consider all possible obvious relations ">0" "0<" "y_{t+1}<> y_t"). In total, $2^3 = 8$ such conditions can be formally written down (zero values are excluded by measurement practice). Moreover, in 4 cases (4) are performed correctly, two cases are contradictory (and therefore impossible), and with two combinations there will be a deliberately false solution, namely, in the cases $y_{t+1} > 0$, $y_t > 0$, $y_{t+1} > y_t$, $y_{t+1} < 0$, $y_t < 0$, $y_{t+1} < y_t$.

Assuming that all combinations are equally probable, we obtain the a priori probability of the correct execution of (1) equal to 2/3.

At the same time, there is no reason to believe that two combinations leading to incorrect predictions will occur on a certain interval several times more often than those leading to correct ones.

5.3 Joint Use of SC Predictor and 1/0 Predictor

One of the signs of the possibility of using (4) as a predictor is the uncorrelatedness (or extremely weak correlation) of successive differences in the values of the considered time series (process) with a tendency to negative correlation of the values y_{i+1}-y_i and y_i, since the “anti-correlation” of two random variables means a tendency to change in the opposite direction [22, 49].

The point is that if within a time interval the data is positively correlated, then changes in a given direction will tend to future changes in the same direction, and the path will be smoother than the normal Brownian motion process. If the data is negatively correlated, then there will be a positive change more likely than a negative one.

The proposed technique of joint use of the continuous and binary sign prediction models presented here is applied in [44] (called as “SC-0/1 procedure”) and compared with MLP, XGB, LSTM predictions widely used in modern practice. Experiments show the high efficiency of the proposed approach.

It is essential that the rule (4) does not require the use of a long sequence of past observations, which may be inefficient in case of strong non-stationarity and nonlinearity of the predicted data.

6 Conclusion

As the analysis of the literature shows, many modern prediction tools based on the principles of MO do not work effectively due to the pronounced nonlinearity of traffic changes and non-stationarity, the possible inadequacy of assumptions about the need for a large amount of previous observations. This paper is an attempt at some ordering and categorization of a huge stream of publications on modern methods, techniques and models of forecasting data of various nature, which should to a certain extent simplify the search and analysis of some results of the theory of random processes, allowing a quick assessment of the predictability of both absolute data values and signs of their change.

For this, the concept of a conceptual model of algorithms for predicting the state of systems in the subject area, widely used in modern Big Date and Software

Engineering, is adapted to represent specific probabilistic models of random sequences and processes of various nature (symbolic, binary, real). Under such a conceptual model, the paper refers to a formalized description of the relationships between elements, objects (both real and model) and goals of mathematical models and prediction algorithms, as well as the relationships between them, expressed in probabilistic terms.

Such a construction allows one to represent classes of models according to the types of predicted data, the probability measures used, and loss functions when making decisions about the acceptability of the received forecast.

Guided by this CM, such sensitive elements of models as the specific manifestations of the nonlinearity of past and future relations, the degree of stationarity, the characteristics of autocorrelation functions, the specificity of distribution laws, and the criteria for prediction accuracy are singled out.

The proposed procedure for using a preliminary assessment of the indicated characteristics of probabilistic models makes it possible to assess the presence of significant non-stationarity in data flows, when it is impossible to consider long sequences for predictor training. For highly non-stationary time series, the learning process is associated with enumeration of parameters (for example, neural network coefficients) if necessary, retraining of models caused by the above non-stationarity, which can take a lot of time even on the most expensive computing systems (GPU, clusters).

Among the tasks of forecasting, the task of predicting signs of increments (direction of change) of the time series process is singled out separately. The previously proposed prediction procedure [44] implemented as a simple heuristic rule for predicting the increment of two neighboring values of a random sequence is considered. The connection of this approach for time series with known approaches for predicting binary sequences is shown.

At the same time, since the efficiency of prediction algorithms theoretically depends on the volume of previous observations, which can be unreliable due to the non-linear and non-stationary nature of the time series (simulating, for example, the traffic of telecommunication networks, which is what rule (4) corresponds to).

References

1. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995)
2. Predict Traffic of LTE Network, [Online] https://www.kaggle.com/naebolo/predict-traffic-of-lte-network. Accessed on 26 Oct 2020
3. Chen, A., Law, J., Aibin, M.: A survey on traffic prediction techniques using artificial intelligence for communication networks. Telecom. ***2***(4), 518–535 (2021)
4. Zhang, J., Tan, D., Han, Zhu, H.: From machine learning to deep learning: Progress in machine intelligence for rational drugv discovery. Drug Discov. Today. **22**(11), 1680–1685 (2017)
5. Rooba, R., Vallimayil, V.: Semantic aware future page prediction based on domain. Int J. Pure Appl. Math. **118**(9), 911–919 (2018)
6. Ryabko, B.: Compression-based methods for nonparametric prediction and estimation of some characteristics of time series. IEEE Trans. on Inf. Theory. **55**(9), 4309–4315 (2009)

7. Brovelli, M., Sanso, F., Venuti, G.: A discussion on the Wiener–Kolmogorov prediction principle with easy-to compute and robust variants. J. Geod. **76**, 673–683 (2003)
8. Feder, M., Merhav, N.: Universal prediction. IEEE T. Inform. Theory. **44**(6), 2124–2147 (1998)
9. Sharma, S..: Activation functions in neural networks (2019). Retrieved from https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6
10. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer Science + Business Media, New York (2009)
11. Fettke, P.: Conceptual Modelling and Artificial Intelligence, Joint Proceedings of Modellierung Short. Workshop and Tools & Demo Papers Workshop on Models in AI (2020)
12. Introduction to Unified Modeling Language (UML) 3rd INSPIRATION Training, GFA (December 4–5, 2012)
13. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: verification of probabilistic real-time systems. In: In, Proc. 23rd International Conference on Computer Aided Verification (CAV'11), pp. 585–591. Vol. 6806 of LNCS, Springer (2011)
14. Ryabko, B.: Prediction of random sequences and universal coding. Probl. Inf. Transm. **24**(2), 3–14 (1988)
15. Ikharo, A.B., Anyachebelu, K.T., Blamah, N.V., Abanihi, V.: Optimising self-similarity network traffic for better performance. Int J Sci Technol Res, Int J Sci Technol. Print ISSN: 2395-6011. https://doi.org/10.32628/IJSRST207413164
16. Cesa-Bianchi, N., Lugosi, G.: On prediction of individual sequences. Ann. Stat. **27**(6), 1865–1895 (1999)
17. Yu, P., Kuo, K.-S., Rilee, M.L., Yu, H.: Assessing Deep Neural Networks as Probability Estimatorsar. Xiv:2111.08239v1 [cs.LG] (2021)
18. Ryabko, B., Monarev, V.: Using information theory approach to randomness testing. J. Stat. Plan. Inference. **133**, 95–110 (2005)
19. Miller, J.B., Sanjurjo, A.: Surprised by the Hot Hand Fallacy? A Truth in the Law of Small Numbers. arXiv:1902.01265v1 [econ.GN] (2019)
20. Nikravesh, A., Ajila, S.A., Lung, C.-H.: An autonomic prediction suite for cloud resource provisioning. J. Cloud Comput. Adv. Syst. Appl. **6**, 3 (2017). https://doi.org/10.1186/s13677-017-0073-4
21. Frenkel, S.: Theoretical aspects of a priori on-line assessment of data predictability in applied tasks 5th International Symposium on Cyber Security Cryptology and Machine Learning CSCML 2021. LNCS. **12716**, 187–195 (2021)
22. Buket Coskun, B., Vardar-Acar, C., Demirtas, H.: A Generalized Correlated,: Random Walk, Converging to Fractional Brownian Motion. arXiv:1903.05424v3 (2019)
23. Ming, L., Jia-Yue, L.: On the Predictability of Long-Range Dependent Series. Mathematical Problems in Engineering Volume (2010). https://doi.org/10.1155/2010/397454
24. Brignoli, D.: DDOS detection based on traffic self-similarity (n.d.). https://ir.canterbury.ac.nz/bitstream/handle/10092/2105/Thesis_fulltext.pdf;sequence=2
25. Graf, S.: Statistically self-similar fractals. Prob. Th. Rel. Fields. **74**, 357–392 (1987)
26. Park, R., Hernández-Campos, F., Le, L., Marron, J., Park, J., Pipiras, V., Smith, F., Smith, L., Trovero, M., Zhu, Z.: Long-range dependence analysis of internet traffic. J. Appl. Stat. **38**(7), 1407–1433 (2011)
27. Megues, P., Molnar, S.: Analysis of Elephant Users in Broadband Network Traffic. 19th EUNICE Workshop on Advances in Communication Networking (2013). https://doi.org/10.1007/978-3-642-40552-5_4
28. Leland, W.E., et al.: On the Self-Similar Nature of Ethernet Traffic (Extended Version), pp. 1–15. IEEE Press, Piscataway (1994)
29. Becchi M., From Poisson Processes to Self-Similarity: a Survey of Network Traffic Models. 2008., https://www.cse.wustl.edu/~jain/cse567-06/ftp/traffic_models1/index.html
30. Bosq, D., Nguyen, H.: A Course in Stochastic Processes. Stochastic Models and Statistical Inference, Kluwer, Dordrecht (1996)

31. Pan, C., Wang, Y., Shi, H., Shi, J., Cai, R.: Network traffic prediction incorporating prior knowledge for an intelligent network. Sensors. **22**(7), 2674 (2022)
32. Lavasani, A., A., Eghlidos, T.: Practical next bit test for evaluating pseudorandom sequences. Comput. Sci. Eng. Electric. Eng. **16**(1), 19–33 (2009)
33. Park, C., Hernandez, F., Le, L., Marron, J.S., Park, J., Pipiras, V., Smith, F.D., Smith, R.L., Trovero, M., Zhu, Z.: Long range dependence analysis of Internet traffic. Journal of Applied Statistics. **38**, 1407–1433 (2004)
34. He, H.: 1 Shitao Cheng, 1 and Xiaofu Zhang, signal nonstationary degree evaluation method based on moving statistics theory. Shock. Vib. **2021**., Article ID 5562110, 18 (2021). https://doi.org/10.1155/2021/5562110
35. Box, G.E.P., Jenkins, G.M., Reinsel, G.C.: Time Series Analysis: Forecasting and Control. Wiley, New York (2008)
36. Lyman, J., Edmonson, W., McCullough, C., Rao, M.: The predictability of continuous-time band limited processes. IEEE Trans. Signal Process. **48**(2), 311–316 (2000)
37. Song, W., Duan, S., Chen, D., Zio, E., Yan, W., Cai, F.: Finite iterative forecasting model based on fractional generalized Pareto motion. Fractal Fract. **6**, 471 (2022)
38. Loiseau, P., Gonçalves, P., Dewaele, G., Borgnat, P., Abry, P., Primet, P.: Investigating self-similarity and heavy-tailed distributions on a large-scale experimental facility. IEEE/ACM Trans. netw. **18**, 1261–1274 (2010)
39. Chen, A., Law, J., Aibin, M.: A survey on traffic prediction techniques using artificial intelligence for communication networks. Telecom. **2**, 517–536 (2021)
40. Vinchoff, C., Chung, N., Gordon, T., Lyford, L., Aibin, M.: Traffic Prediction in optical networks using graph convolutional generative adversarial networks. In: In Proceedings of the International Conference on Transparent Optical Networks, pp. 3–6. Bari, Italy (2020)
41. Aibin, M.: Deep Learning for Cloud Resources Allocation: Long-Short Term Memory in EONs. In Proceedings of the International Conference on Transparent Optical Networks, Angers, France, 9–13 July 2019; pp. 8–11
42. Yin, F., Wang, J., Guo, C. (eds.): A Boosting-Based Framework for Self-Similar and Non-linear Internet Traffic Prediction ISNN 2004, pp. 931–936. LNCS 3174 (2004)
43. Shi, Y., Fernando, B., Hartley, R.: Action Anticipation with RBF Kernelized Feature Mapping RNN. arXiv:1911.07806v3 [cs.CV] 11 Jul (2021)
44. Frenkel, S.: Predicting the direction of changes in the values of time Series for relatively small training samples. In: 6th International Symposium on Cyber Security Cryptology and Machine Learning CSCML 2021CSCML, Beer-Sheva, Israel, pp. 118–134. Proceedings, Lecture Notes in Computer Science (13301) (2022)
45. The Influence of Long-Range Dependence on Traffic Prediction Sven A. M. Östring, H. Sirisena Published 11 June 2001 Computer Science ICC 2001. IEEE International Conference on Communications. Conference Record (Cat. No.01CH37240)
46. Nikravesh, A.Y., Ajila, S.A., Lung, C.-H.: An autonomic prediction suite for cloud. J. Cloud Comput. Adv. Syst. Applic. **6**, 3 (2017). https://doi.org/10.1186/s13677-017-0073-4
47. Zhao, A., Liu, Y.: Application of Nonlinear Combination Prediction Model for Network Traffic. 2nd International Conference on Electronic & Mechanical Engineering and Information Technology (EMEIT-2012). Proceedings, 2337–2340 (2012)
48. Christoffersen, P., Diebold, F.: Financial asset returns, direction-of-change forecasting, and volatility dynamics. Manag. Sci. **52**(8), 1273–1287 (2006)
49. Sornette, D., Andersen, J.: Increments of uncorrelated time series can be predicted with a universal 75% probability of success. Int. J. Mod. Phys. **11**(4), 713–720 (2000)
50. Cloud, B.L., Dalmazo, L., Vilela, M.: Performance analysis of network traffic predictors. J. Netw Syst Manage. **25**, 290–320 (2017)
51. Aryan, M.: Efficient Methods for Large-Scale Empirical Risk Minimization. A Doctoral Thesis, Philadelphia, PA (2017)
52. Kleeman, R.: Information theory and dynamical system predictability. Entropy. **13**, 612–649 (2011)

A Three-Step Method for Audience Extension in Internet Advertising Using an Industrial Taxonomy

Dmitry Frolov and Zina Taran

1 Introduction

Modern technologies transform many areas of human life and modify industries. The era of Big Data brought on remarkable possibilities for extracting meaningful insight from the data. In particular, the growth of the digital advertising, whereby an increasing share of advertisement moves from traditional formats (such as TV, radio, out-door) to the Internet both accelerates the production of raw data and creates the demand for insight generated from it. Digital advertising brings new ways to investigate potential audiences, create and control advertisements, and evaluate results. Companies wish to predict consumer preferences, determine relevant customer segments for particular products or services, and target marketing offers based on the data. The amount and diversity of data on one hand, and the rapidly increasing sophistication of methods on the other allow for such predictions to be carried out with ever increasing accuracy. For example, they allow marketers to expand on their time-tested ideas of targeting consumers. Whereas in the past simple regression methods allowed a company to identify a group of people that were very much like their existing customers, nowadays it becomes possible to go beyond that to what is now referred to as an audience extension. In general, ability to derive better, more accurate insight and to extract more information from the data becomes essential for any business that wishes to remain competitive. The objective of this paper is to propose a better method of extracting useful user insight from the

D. Frolov (✉)
Department of Data Analysis and Artificial Intelligence, HSE University, Moscow, Russia

Z. Taran
Division of Management, Marketing, and Business Administration, Delta State University, Cleveland, MS, USA

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_8

data that the organization is likely already collecting without the need to gather new information.

2 Programmatic Targeting in Internet Advertising

Nowadays, digital advertising approaches allow an advertiser to choose appropriate user audiences for their campaigns. The most popular approach is programmatic, a recently emerged technique. Programmatic brings real-time bidding (RTB) to allow an advertiser to make their purchasing decisions separately for every contact with every user [6, 17]. A real-time bidding system provides an auction for the advertisement impression between multiple advertisers, and each advertiser has a possibility to buy relevant audience only [7, 13, 16]. This technology relies on assigning users with their interest profiles containing estimates of levels of their interest in this or that market segment. Consider a conventional, currently much popular, approach to programmatic selection of targeted audiences [2, 14]. This approach (CAS) requires to pre-specify a threshold t (usually, $t = 0.3$ or $t = 0.4$) and formulate a condition: Given a set of market segments provided by an advertiser and a user's profile, check whether the profile has one or more of the advertiser's market segments. If yes, the user is selected as part of the audience if at least one of these segments has its fuzzy weights greater than t, according to the CAS rule.

An issue with CAS is that the number of users satisfying the condition at threshold t may be less than the number specified in the advertisement order; say, they want to show their advert to a million users, but only three hundred thousand of those under observation satisfy the condition. In this case, a conventional strategy is to have CAS extended (CASE) by lessening t to t', $t' < t$, so that more users satisfy the condition at t' than at t. This may increase the number of users exposed to the advert indeed, but usually the efficiency lessens, because the added users have a weaker tendency to be impressed by the advert. A similar diminishing response rates can be caused by popular the so-called look-alike techniques [8].

To overcome this issue without decreasing the threshold, we propose using segments profiles 'generalized' over an industrial taxonomy rather than the original ones. This method utilizes a novel algorithm for most adequate generalization in taxonomies [1] to extend user segments by parsimoniously lifting them over IAB content taxonomy into a higher rank 'head subject'. This algorithm was proposed as an intelligent information retrieval tool [1]. Here it is applied to a very different task of targeted advertisement.

An approach for user profiling to involve an industrial ontology, to which we follow, was originated by [9] and further advanced by various authors such as [2]. The approach involves the following blocks: (a) an industrial ontology in the format of a rooted tree taxonomy, to provide for marketing user behavior segments, (b) a system for tracking users' web surfing histories, and (c) a device to convert the users' histories into user profiles. A user profile assigns weights to taxonomy segments according to user's interests in them. Advertising system has to detect users belonging to chosen segments and provide advertisement impressions for them. To apply this approach, we use industrial taxonomy IAB (see https://www.iab.com/ [3]). As a device converting user surfing histories into user profiles, we use a random forest classifier based on paper [15].

The paper is organized as follows. Section 3 is devoted to a model and a method of computation generalization. Section 4 describes our combined method, Development and Lifting of User Segments Profile (DLUSP), to extend audiences of targeted advertising. Section 5 describes results of several real-world experiments comparing a popular conventional approach and DLUSP: our method does increase the number of successful matches between user segments and campaign segments 2.5-3-fold without losing in the targeting quality. Section 6 draws a conclusion and lists directions for future works.

3 Parsimoniously Generalization a Fuzzy Thematic Subset in Taxonomy

Let us consider the main definitions related to generalization in taxonomies, according to [1], to describe the principles of computational generalization in detail.

Mathematically, a taxonomy is a rooted tree whose nodes are annotated by taxonomy topics. We consider the following problem. Given a fuzzy set S of taxonomy leaves, find a node $t(S)$ of higher rank in the taxonomy, that covers the set S in a most specific way. Such a “generalization”, or “lifting” problem is a mathematical explication of the human facility for generalization, that is, “the process of forming a conceptual form” of a phenomenon represented, in this case, by a fuzzy leaf subset.

Consider, for the sake of simplicity, a hard set S shown with five black leaf boxes on a fragment of a tree in Fig. 1. Figure 2 illustrates the situation at which the set of black boxes is lifted to the root, which is shown by blackening the root box, and its offspring, too. If we accept that set S may be generalized by the root, this would lead to a number, four, white boxes to be covered by the root and, thus, in this way, falling in the same concept as S even as they do not belong in S. Such a situation will be referred to as a gap. Lifting with gaps should be penalized. Altogether, the number of conceptual elements introduced to generalize S here is 1 head subject, that is, the root to which we have assigned S, and the 4 gaps occurred just because of the topology of the tree, which imposes this penalty. Another lifting decision is illustrated in Fig. 3: here the set is lifted just to the root of the left branch of the tree. We can see that the number of gaps has drastically decreased, to just 1. However, another oddity emerged: a black box on the right, belonging to S but not covered by the root of the left branch at which the set S is mapped. This type of error will be referred to as an offshoot. At this lifting, three new items emerge: one head subject, one offshoot, and one gap. This is less than the number of items emerged at lifting the set to the root (one head subject and four gaps, that is, five), which makes it

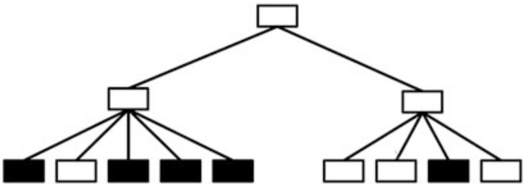


Fig. 1 A crisp query set, shown by black boxes, to be conceptualized in the taxonomy

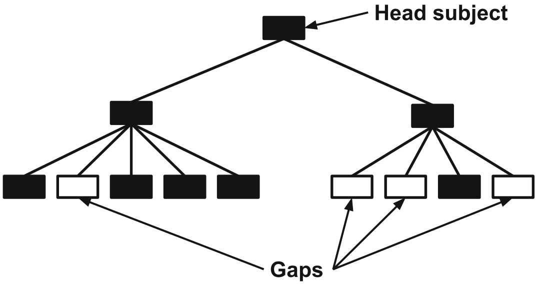


Fig. 2 Generalization of the query set from Fig. 1 by mapping it to the root, with the price of four gaps emerged at the lift

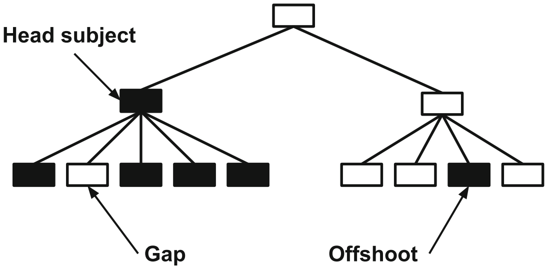


Fig. 3 Generalization of the query set from Fig. 1 by mapping it to the root of the left branch, with the price of one gap and one offshoot emerged at this lift

more preferable. Of course, this conclusion holds only if the relative weight of an offshoot is less than the total relative weight of three gaps.

We are interested to see whether a fuzzy set S can be generalized by a node t from higher ranks of the taxonomy, so that S can be thought of as falling within the framework covered by the node t. The goal of finding an interpretable pigeon-hole for S within the taxonomy can be formalized as that of finding one or more "head subjects" t to cover S with the minimum number of all the elements introduced at the generalization: head subjects, gaps, and offshoots. This goal realizes the principle of Maximum Parsimony (MP).

Consider a rooted tree T representing a hierarchical taxonomy so that its nodes are annotated with key phrases signifying various concepts. We denote the set of all its *leaves* by I. The relationship between nodes in the hierarchy is conventionally expressed using genealogical terms: each node $t \in T$ is said to be the *parent* of the nodes immediately descending from t in T, its *children*. We use $\chi(t)$ to denote the set of children of t. Each *interior* node $t \in T - I$ is assumed to correspond to a concept that generalizes the topics corresponding to the leaves $I(t)$ descending from

t, viz. the leaves of the subtree $T(t)$ rooted at t, which is conventionally referred to as the *leaf cluster of* t.

A *fuzzy set* on I is a mapping u of I to the non-negative real numbers that assigns a membership value, or support, $u(i) \geq 0$ to each $i \in I$. We refer to the set $S_u \subset I$, where $S_u = \{i \in I : u(i) > 0\}$, as the *base* of u. In general, no other assumptions are made about the function u, other than, for convenience, commonly limiting it to not exceed unity. Conventional, or *crisp*, sets correspond to binary membership functions u such that $u(i) = 1$ if $i \in S_u$ and $u(i) = 0$ otherwise.

Given a fuzzy set u defined on the leaves I of the tree T, one can consider u to be a (possibly noisy) projection of a higher rank concept, u's "head subject", onto the corresponding leaf cluster. Under this assumption, there should exist a head subject node h among the interior nodes of the tree T such that its leaf cluster $I(h)$ more or less coincides (up to small errors) with S_u. This head subject is the generalization of u to be found. The two types of possible errors associated with the head subject, if it does not cover the base precisely, are false positives and false negatives, referred to in this paper, as *gaps* and *offshoots*, respectively. They are illustrated in Figs. 2 and 3. Given a head subject node h, a gap is a node t covered by h but not belonging to u, so that $u(t) = 0$. In contrast, an offshoot is a node t belonging to u so that $u(t) > 0$ but not covered by h. Altogether, the total number of head subjects, gaps, and offshoots has to be as small as possible. To this end, we introduce a penalty for each of these elements. Assuming for the sake of simplicity, that the black box leaves on Fig. 1 have membership function values equal to unity, one can easily see that the total penalty at the head subject raised to the root (Fig. 2) is equal to $1 + 4\gamma$ where 1 is the penalty for a head subject and γ, the penalty for a gap, since the lift on Fig. 2 involves one head subject, the root, and four gaps, the blank box leaves. Similarly, the penalty for the lift on Fig. 3 to the root of the left-side subtree is equal to $1 + \gamma + \lambda$ where λ is the penalty for an offshoot, as there is one copy of each, head subject, gap, and offshoot, in Fig. 3. Therefore, depending on the relationship between γ and λ either lift on Fig. 2 or lift on Fig. 3 is to be chosen. That will be the former, if $3\gamma < \lambda$, or the latter, if otherwise.

Consider a candidate node h in T and its meaning relative to fuzzy set u. An *h-gap* is a node g of $T(h)$, other than h, at which a *loss* of the meaning has occurred, that is, g is a maximal u-irrelevant node in the sense that its parent is not u-irrelevant. Conversely, establishing a node h as a head subject can be considered as a *gain* of the meaning of u at the node. The set of all h-gaps will be denoted by $G(h)$. A node $t \in T$ is referred to as *u-irrelevant* if its leaf-cluster $I(t)$ is disjoint from the base S_u. Obviously, if a node is u-irrelevant, all of its descendants are also u-irrelevant.

An *h-offshoot* is a leaf $i \in S_u$ which is not covered by h, i.e., $i \notin I(h)$. The set of all h-offshoots is $S_u - I(h)$. Given a fuzzy topic set u over I, a set of nodes H will be referred to as a *u-cover* if: (a) H covers S_u, that is, $S_u \subseteq \bigcup_{h \in H} I(h)$, and (b) the nodes in H are unrelated, i.e. $I(h) \cap I(h') = \emptyset$ for all $h, h' \in H$ such that $h \neq h'$. The interior nodes of H will be referred to as *head subjects* and the leaf nodes as *offshoots*, so the set of offshoots in H is $H \cap I$. The set of *gaps* in H is the union of $G(h)$ over all head subjects $h \in H - I$.

We define the penalty function $p(H)$ for a u-cover H as:

$$p(H) = \sum_{h \in H-I} u(h) + + \sum_{h \in H-I} \sum_{g \in G(h)} \lambda v(g) + \sum_{h \in H \cap I} \gamma u(h). \tag{1}$$

The problem we address is to find a u-cover H that globally minimizes the penalty $p(H)$. Such a u-cover is the parsimonious generalization of the set u.

Before applying an algorithm to minimize the total penalty, one needs to execute a preliminary transformation of the tree by pruning it from all the non-maximal u-irrelevant nodes, i.e. descendants of gaps. Simultaneously, the sets of gaps $G(t)$ and the internal summary gap importance $V(t) = \sum_{g \in G(t)} v(g)$ in Eq. (1) can be computed for each interior node t. We note that the elements of S_u are in the leaf set of the pruned tree, and the other leaves of the pruned tree are precisely the gaps. After this, our lifting algorithm ParGenFS applies. For each node t, the algorithm ParGenFS computes two sets, $H(t)$ and $L(t)$, containing those nodes in $T(t)$ at which respectively gains and losses of head subjects occur (including offshoots). The associated penalty $p(t)$ is computed too.

An assumption of the algorithm is that no gain can happen after a loss. Therefore, $H(t)$ and $L(t)$ are defined assuming that the head subject has not been gained (nor therefore lost) at any of t's ancestors. The algorithm ParGenFS recursively computes $H(t)$, $L(t)$ and $p(t)$ from the corresponding values for the child nodes in $\chi(t)$.

Specifically, for each leaf node that is not in S_u, we set both $L(\cdot)$ and $H(\cdot)$ to be empty and the penalty to be zero. For each leaf node that is in S_u, $L(\cdot)$ is set to be empty, whereas $H(\cdot)$, to contain just the leaf node, and the penalty is defined as its membership value multiplied by the offshoot penalty weight γ. To compute $L(t)$ and $H(t)$ for any interior node t, we analyze two possible cases: (a) when the head subject has been gained at t and (b) when the head subject has not been gained at t.

In case (a), the sets $H(\cdot)$ and $L(\cdot)$ at its children are not needed. In this case, $H(t)$, $L(t)$ and $p(t)$ are defined by:

$$\begin{aligned} H(t) &= \{t\} \\ L(t) &= G(t) \\ p(t) &= u(t) + \lambda V(t). \end{aligned} \tag{2}$$

In case (b), the sets $H(t)$ and $L(t)$ are just the unions of those of its children, and $p(t)$ is the sum of their penalties:

$$\begin{aligned} H(t) &= \bigcup_{w \in \chi(t)} H(w) \\ L(t) &= \bigcup_{w \in \chi(t)} L(w) \\ p(t) &= \sum_{w \in \chi(t)} p(w). \end{aligned} \tag{3}$$

To obtain a parsimonious lift, whichever case gives the smaller value of $p(t)$ is chosen.

When both cases give the same values for $p(t)$, we may choose, say, (a). The output of the algorithm consists of the values at the root, namely, H – the set of head subjects and offshoots, L – the set of gaps, and p – the associated penalty.

ParGenFS Algorithm
INPUT: u, T
OUTPUT: $H = H(Root), L = L(Root), p = p(Root)$

```
I   Base case: for each leaf of the $T$.
        for each leaf $i \in I$
            $L(i) = \emptyset$
            if $u(i) > 0$
                $H(i) = \{i\}$
                $p(i) = \gamma u(i)$
            else
                $H(i) = \emptyset$
                $p(i) = 0$
II  Recursion: for each internal node of the $T$, down up to the Root.
            $p_{gain} = u(t) + \lambda V(t)$
            $p_{nogain} = \sum_{w \in \chi(t)} p(w)$
            if $p_{gain} \leq p_{nogain}$
                $H(t) = \{t\}$
                $L(t) = G(t)$
                $p(t) = p_{gain}$
            else
                $H(t) = \bigcup_{w \in \chi(t)} H(w)$
                $L(t) = \bigcup_{w \in \chi(t)} L(w)$
                $p(t) = p_{nogain}$
III Return the sets $H = H(Root), L = L(Root), p = p(Root)$.
```

It was mathematically proven that the algorithm ParGenFS leads to an optimal lifting indeed [1].

4 A 3-Step Method, Developing and Lifting of User Segments Profiles

Intuitively, the method consists of three steps: (1) computing membership values for the interest segments for a user by a classifier; (2) performing generalization of those sets and obtaining high-ranked segments, which is a core part of the method; (3) obtaining a set of advertising campaigns for a user.

There are three inputs to our method, Developing and Lifting of User Segment Profile (DLUSP). These are: (i) a large set of internet users from which the audience

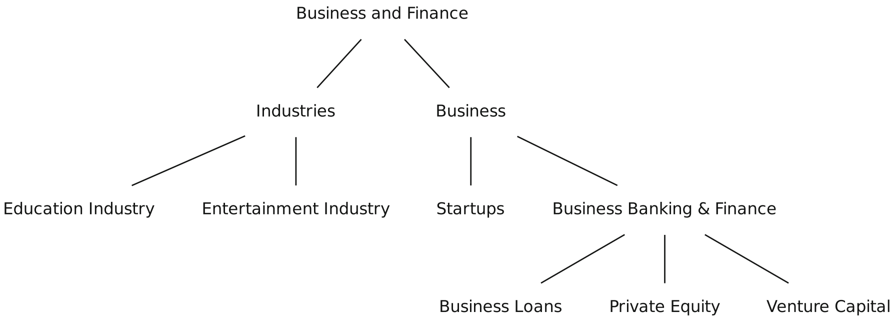


Fig. 4 IAB Contents taxonomy fragment

is recruited for an advertisement; (ii) an industrial taxonomy of goods and services; (iii) a set of taxonomy segments relevant to an advertisement under consideration.

The user set in (i) is maintained by a special service storing information of millions individual users visiting popular sites such as amazon.com or ozon.ru for making purchases, getting services, etc., for native advertising. Such is Data Management Platform (DMP) developed in a small company, start-up Natimatica, Ltd. (see https://natimatica.com).

The industrial taxonomy in (ii) is exemplified by the IAB Content taxonomy [3], which is a 4-layer rooted tree of taxonomy topics. We focus on its leaf segments. A fragment from Business and Finance branch of the IAB taxonomy can be seen in Fig. 4.

The set of taxonomy segments in (iii) comes from a chat between a specially assigned company employee and the advertiser of the contents of their advertisement.

Our method DLUSP includes two technically loaded components: Development of user segment profile (DUSP) and Lifting of user segment profile (LUSP).

The DUSP works within the DMP; it assigns any individual user with a fuzzy set of IAB leaf segments relevant to their visits. The visits are reflected in texts from the visited pages. These texts are transformed into a numerical format [5], which are further transformed into fuzzy membership values for leaf segments by a special Random Forest classifier [15]. The resulting fuzzy set is referred to as the user's profile.

The LUSP works at a given a user segments profile. It finds a higher-rank taxonomy node generalizing the profile. We developed an algorithm in [1] to 'lift' the profile to its 'head subject', a node in the higher ranks of the taxonomy tree. The head subject tightly covers the profile, usually bringing in some errors, 'gaps' and 'offshoots'. A gap is a taxonomy tree node covered by the head subject but not being in the profile. An offshoot is a node in in the profile, but not covered by the head subject. Our algorithm globally minimizes a penalty function combining the numbers of head subjects and gaps and offshoots, as well as fuzzy membership values. In this, we use experimentally chosen penalties for gaps ($\lambda = 0.2$) and

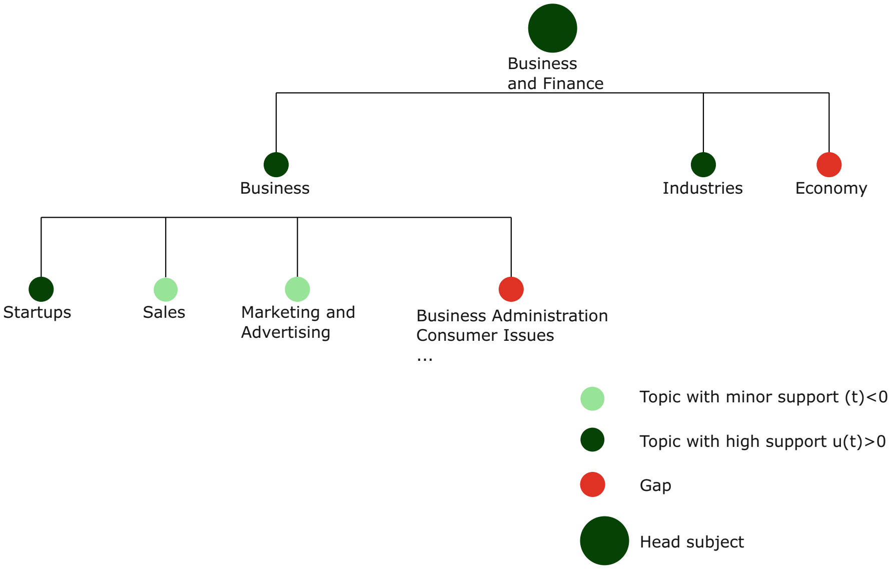


Fig. 5 An example of user segments profile lifted in IAB taxonomy fragment

offshoots ($\gamma = 0.9$). An example of an optimally lifted segments profile is presented in Fig. 5.

The developed method consists of the following steps applied at a set of advertising campaigns requested by advertisers.

DLUSP Algorithm

INPUT: a user identifier U; threshold t for user selection; set of active advertising campaigns C

OUTPUT : C_F – a set of targeted campaigns for the user identifier U

In a loop A over elements of C:

I Perform the DUSP to obtain a user interest profile S for the given identifier U from the user data storage system (DMP).

II Obtain a set of head subjects H_S using the ParGenFS algorithm to generalize the user profile S ($\lambda = 0.2$, $\gamma = 0.9$).

III For the set of active advertising campaigns $C = \{c_1, \ldots, c_K\}$, extract the sets of segments assigned to them: $\{p_{c_1}, \ldots, p_{c_K}\}$. Compare the obtained head subjects H_S with each of the sets of segments p_{c_i}. Obtain C_F – the set advertising campaigns that should be shown to U. To do this, check the intersection of p_{c_i} and the set $\chi(H_S)$ formed by the union of H_S and all its descendants in the taxonomy tree:

(a) If $p_{c_i} \cap \chi(H_S) \neq \varnothing$, add the c_i campaign to the set C_F;
(b) if $p_{c_i} \cap \chi(H_S) = \varnothing$, do not add the campaign c_i to the set C_F.

End of loop A

IV Return the set C_F.

Table 1 Examples of lifting of user segments profiles

User	Segments assigned by a classifier (with membership values)	Segments after applying LUSP
1	{Cloud Computing (0.596), Web Development (0.481), Internet for Beginners (0.432), IT and Internet Support (0.356), Social Networking (0.312)}	{Internet (1.0)}
2	{Men's Jewelry and Watches (0.662), Men's Business Wear (0.514), Men's Casual Wear (0.443), Men's Outerwear (0.320)}	{Men's Fashion (1.0)}
3	{3-D Graphics (0.678), Video Software (0.571), Graphics Software (0.570), Operating Systems (0.308), Business Accounting and Finance (0.351) }	{Computer Software and Applications (0.902), Business Accounting and Finance (0.351)}

Note that the DLUSP method does not eliminate the need for targeting that is specified outside the terms of the user's interests, for example, geographical (user regions) or demographics (gender, user age) criteria.

Three examples of applying LUSP to user segments profiles are presented in Table 1.

5 Experiments and Results

Table 2 presents comparative results of testing DLUSP method at real life advertising campaigns in Natimatica, Ltd., involving the three targeting methods under consideration:

1. Conventional programmatic targeting based on matching segments (CAS);
2. Audience extending by decreasing thresholds (CASE);
3. Audience extending by lifting user segments profiles (LUSP).

Our comparison criteria were the following: (a) numbers of advertising impressions obtained, (b) numbers of clicks, and (3) click-through rates (CTR, (b)/(a)). The last metric is especially important, because it usually characterizes the quality of the audience impressed by the advertisement. At CASE method, the lessened threshold values were chosen to have audience sizes approximately equal to those emerged at the DLUSP method.

Out of two methods for expanding the audience under comparison, CASE and LUSP, the latter is the clear winner. Indeed, in all the three cases presented, the minimum increase of the number of clicks by LUSP was 74%, whereas the maximum increase by CASE was 44%.

Table 2 Advertising campaign results at different targeting methods

Campaign	IAB segments selected	Metric	Method CAS	 CASE	 LUSP
Software for parental control of children. Duration: 10 days	Internet Safety, Anti-virus Software, Daycare and Pre-School, Parenting Children Aged 4–11, Parenting Teens	Impressions	378933	1017598 (+168.5%)	942104 (+148.6%)
		Clicks	1061	1526 (+43.8%)	2544 (+139.8%)
		CTR,%	0.28	0.15 (−46.4%)	0.27 (−3.6%)
Frame houses for villages. Duration: 4 days	Houses, Outdoor Decorating, Gardening, Remodeling & Construction, Landscaping	Impressions	87599	160204 (+82.9%)	153032 (+74.7%)
		Clicks	201	288 (+43.3%)	367 (+82.6%)
		CTR,%	0.24	0.18 (−25.0%)	0.24 (+0.0%)
Mortgage at a major Russian bank. Duration: 10 days	Home Financing, Personal Loans	Impressions	159342	275035 (+72.6%)	289308 (+81.6%)
		Clicks	749	853 (+13.9%)	1302 (+73.9%)
		CTR,%	0.47	0.31 (−34.0%)	0.45 (−4.3%)

6 Conclusion and Further Work

One can clearly see the efficiency of the proposed method within the advertising approach based on user profiling. This method is based on lifting of user segments within an industrial taxonomy. It appears that the found general head subjects are as informative of customer interests as the most specific segments in their surfing histories. This finding allows us to significantly expand the audience for an advert without much loss in the click-through rate.

The observation that transit to head subjects does not much change the customer clicking attitudes accords with intuition. An additional advantage of our method is that it involves no external audiences.

Directions for future research lie in further expanding of tested advertising campaigns. Furthermore, we plan to compare DLUSP method with audience proximity look-alike methods [8]. Also, our plans include developing a strategy for parameters fitting in the lifting algorithm, λ and γ.

References

1. Frolov, D., Nascimento, S., Fenner, T., Mirkin, B.: Parsimonious generalization of fuzzy thematic sets in taxonomies applied to the analysis of tendencies of research in data science. Inf. Sci. **512**, 595–615 (2020)

2. Hoppe, A., Roxin, A., Nicolle, C.: Customizing semantic profiling for digital advertising. In: OTM Confederated International Conferences, On the Move to Meaningful Internet Systems, pp. 469–478. Springer, Berlin (2014).
3. IAB Tex Lab Content Taxonomy. https://www.iab.com/guidelines/iab-quality-assurance-guidelines-qag-taxonomy/. Last Accessed 31 Jan 2021
4. Jauvion, G., et al.: Optimization of a SSP's header bidding strategy using Thompson sampling. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 425–432. ACM, New York (2018)
5. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: Proceedings of the 31st International Conference on Machine Learning (ICML), pp. 1188–1196 (2014)
6. Li, X., Guan, D.: Programmatic buying bidding strategies with win rate and winning price estimation in real time mobile advertising. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 447–460. Springer, Cham (2014)
7. OpenRTB Protocol. https://www.iab.com/guidelines/real-time-bidding-rtb-project/. Last Accessed 31 Jan 2021
8. Popov, A., Iakovleva, D.: Adaptive look-alike targeting in social networks advertising. Proc. Comput. Sci. **136**, 255–264 (2018)
9. Pretschner, A., Gauch, S.: Ontology based personalized search. In: Proceedings of the 11th IEEE International Conference on Tools with AI, pp. 391–398 (1999)
10. Punjabi, S., Bhatt, P.: Robust factorization machines for user response prediction. In: Proceedings of the 2018 World Wide Web Conference on World Wide Web, pp. 669–678 (2018)
11. Rodgers, S., Thorson, E.: Digital Advertising. Routledge, New York (2017)
12. She, X., Wang, S.: Research on advertising click-through rate prediction based on CNN-FM hybrid model. In: 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), pp. 56–59. IEEE, 2 (2018)
13. Wang, J., Yuan, S., Zhang, W.: Real-time bidding based display advertising: mechanisms and algorithms. In: European Conference on Information Retrieval (ECIR) 2016, pp. 897–901. Springer, Cham (2016)
14. Wiener, D.A., Hsu, J.K., Papa, S.J., Hilal, S.W., Chen, K.M., Hui, V.W.N., Hekster, B., Connelly, J.P.: Extending audience reach in messaging campaigns using probabilistic id linking. U.S. Patent Application No. 16/745,115 (2020)
15. Xu, B., Guo, X., Ye, Y., Cheng, J.: An improved random forest classifier for text categorization. JCP **12**(7), 2913–2920 (2012)
16. Yuan, Y., Wang, F., Li, J., Qin, R.: A survey on real time bidding advertising. In: International Conference on Service Operations and Logistics, and Informatics (SOLI), pp. 418–423. IEEE (2014)
17. Zhang, S., Wakefield, R., Huang, J., Li, X.: Exploring determinants of consumers' attitudes toward real-time bidding (RTB) advertising. Inf. Technol. People **34**(2), 496–525 (2021)

From Prebase in Automata Theory to Data Analysis: Boris Mirkin’s Way

Boris Goldengorin

1 Mirkin’s Prebase in Abstract Automata Theory: A Warm Up

Let me review some of Boris’ results.

One cannot miss a groundbreaking discovery of simple relations between regular expressions and abstract automata made by Boris Mirkin and his PhD supervisor Mark Spivak in 1960s in the University of Saratov (Russia), a faraway outpost of the Computer Science developments, to never ever reappear on the map after both left the place. It should be said that these results were overlapping those by J. Brzozowski in the University of Toronto related to the introduced by him the concept of event derivative. Nevertheless, Prof. J. Brzozowski made a good use of his Polish roots: He noted the work by Boris published in Russia, in Russian, and described it in several synopses in the Journal of Symbolic Logic in 1969–1971 (see [1]) – these pieces paved the way to lasting recognition of Boris’s work in the concept of Mirkin’s prebase. What is curious about this – the fact that Boris did not know anything of these publications, because of the power of the Soviet “iron curtain” effectively preventing the Russian scientists from any international contacts under pre-text of the “class struggle”. He learnt of J. Brzozowski’s activities in 40 years, in the dusk of their careers. Here is an extract from B. Mirkin’s letter to J. Brzozowski

B. Goldengorin (✉)
Department of Mathematics, New Uzbekistan University, Tashkent, Uzbekistan

The Scientific and Educational Mathematical Center «Sofia Kovalevskaya Northwestern Center for Mathematical Research», Pskov State University, Pskov, Russia

Department of Discrete Mathematics, Phystech School of Applied Mathematics and Informatics, Moscow Institute of Physics and Technology, Dolgoprudny, Moscow region, Russia
e-mail: b.goldengorin@newuu.uz; goldengorin.bi@mipt.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_9

of 2012: “If I had known of this, then my decision to walk off the automata field might have been more difficult for me – or never happen!”

2 Jump to Group Choice and Data Analysis

In the end of 1960s B. Mirkin moved to work in the Institute of Economics, Siberian Branch of the USSR Academy of Sciences, Novosibirsk, Russia.

There, he started research on consensus among binary relations. He was motivated by the idea that since socio-economic decisions are mostly not quantitative (at least, this was so in the USSR), then corresponding mathematical models should be based on non-quantitative data too. In Boris’ own words, he “proceeded to extend rankings to unordered partitions as embodiment of nominal features. The world had moved on by then and was becoming more receptive to minority rights. For example: everybody knows that most people are “early birds”, whereas some are “owls” preferring working long evenings and waking up late; thus, the normative rule that “the early bird catches the worm” was changing to a less restrictive motto: “no matter which, early bird or owl, just behave accordingly” [2].

This idea underlay Boris’ move from rankings to partitions including development of what later was called Mirkin’s distance between partitions, inspired by analogous development of the distance between rankings by J. Kemeny [3]. Also, this research produced various extensions of the celebrated Arrow’s theorem of impossibility of democratic choice [4], culminating in characterization of Arrow’ monotonicity and alternative independence axioms with those consensus choice functions which he called federation consensus choice rules [5]. Supplemented with analyses of real-world data on expert judgement and voting behavior, this led to publication of Mirkin’s first monograph, on mathematics of group choice and related issues [6]. Boris began working on that by following an advice from Misha Braverman, a founding father of Russian data science efforts. Misha said: “This subject is getting popular, and it would be a good idea to show to our ‘cognoscenti’ that there is no point in reinventing the wheel [2].” This book not only propelled B. Mirkin into top ranks of the Soviet mathematics-economics research community, but it also opened way to do research and get published on this subject by scientists in several Soviet satellite countries such as Bulgaria. The authorities in those countries did not permit publications on subjective preferences because no work in that area had been published in the Soviet Union.

Developing further approximation models for binary relation data, Boris came to the analysis of structure of pair-wise similarity, or pair-wise interaction, matrices. Encouraged by his bosses who instituted a lab for data analysis managed by Boris, he, together with his collaborators such as V. Kupershtokh, V. Trofimov, and P. Rostovtsev, developed a machinery, both methods and codes, for finding approximate individual clusters and partitions in similarity data [7], as well as more weird structures such as “structured partition” [8] or “chain order partition” [9]. The latter, chain order partition, is a unique development arguably having no analogues in the literature, a structure to model processes of change [9]. Unfortunately, currently no research is conducted over these two concepts.

The former, structured partition, is a partition with a network of links between the parts to be found, the concept akin to that of block-model emerged in social psychology [10]. This was applied, under Boris' supervision, to (a) the analysis of organization structures of big industrial enterprises leading to improvements of those when two structures in big industrial enterprises, the material flow and control ones, were co-analyzed [11], and (b) analysis of genetic structures based on genetic experiment results [12].

Motivated by the idea that partition is a structure to properly represent categorical features, B. Mirkin extended his thinking to what he referred to as "categorical factor analysis" [13, 14]. He represented the structure to be found by an "ideal" similarity matrix, being zero-one binary relation matrix assigned with real alpha and beta for one and zero, respectively. After such a structure has been found to approximate a given similarity matrix, the one-by-one extraction approach of the Principal Component Analysis can be applied to compute a residual similarity matrix so that a next approximate structure can be looked for (this is partially described in a later review [14], including an independent discovery of what is referred to as additive clustering). This approach was successfully applied by B. Mirkin and his collaborators for extracting from similarity data such structures as partitions [14], single clusters [15], bi- and tri-clusters [16], fuzzy clusters [17], and communities in feature-rich networks [18].

Then B. Mirkin moved to cluster-modeling conventional object-to-feature data matrices. He was the first to develop a matrix factorization model with a least-squares fitting criterion to underlie the celebrated k-means clustering [19]. He has referred to this view as a "data recovery approach" [20, 21], which predates the very popular "encoder-decoder data reconstruction" approach in deep learning. In clustering, this brings forth a celebrated Pythagorean decomposition of the square data scatter in the sum of two items, the k-means square error criterion, the unexplained part, and a complementary criterion, the explained part. The explained part sheds a really new light on such issues as the "real" goal of k-means clustering (finding big anomalous clusters, according to B. Mirkin) and the contributions of nominal features (appearing to coincide with various measures of deviation from the statistical independence, including the celebrated Pearson's chi-squared, which relates, rather unexpectedly, to the data normalization scaling utilised) [21]. This alone, in my view, should suffice to bring forward a sound mathematical theory for combinatorial clustering to embrace such aspects as mixed scale data, data normalization options, clustering criteria and algorithms at different data formats (contingency tables, entity-to-feature data tables, similarity data), etc.

B. Mirkin masterly used these equivalencies, first discovered in the concept of matrix correlation between features developed by him and his collaborators (see, for example, in [22]), for grouping of massive sociology survey datasets together with P.S. Rostovtsev. At that time, back in 1970s, a "massive dataset" was to embrace some several dozen thousand objects, so that it could not fit into a single computer memory and, thus, had to be processed over an externally linked magnet tape at which the dataset had to be stored. Such was a dataset collected by a renown Russian social hygienist T. Shanin in mid-1970s over various respiratory diseases among more than 60,000 residents of Akademgorodok, a research center at which the Institute of Economics is located. Using an approximation criterion expressed in terms of bivariate association indexes, Rostovtsev and Mirkin were able to classify the respondents in 14 "respiratory disease" clusters and, further on, test the hypotheses of main factors underlying this structure. The organizer of the survey suspected, with a degree of certainty, that the factors were human habits of "smoking" and "drinking". To their surprise, the computer scientists failed to demonstrate that. On the contrary, both "smoking" and "drinking" appeared almost statistically independent of their clustering results. Instead, two different factors have been found: "bad housing" and "presence of the disease in the family". At that time, under the Soviet rule, such a result was absolutely a "no-go", as contradicting to both medical views of that time and the soviet mentality, so that the report was never published. Of course, currently, in 50 years, it seems obvious that the official point of view stressed on individuals themselves as those responsible for their diseases by smoking and drinking, whereas the researchers demonstrated that the diseases were socially conditioned rather than individually: It is the "socialist state" which is responsible for the need in both improvement of housing and medical services.

B. Mirkin was not satisfied with the developed methods, he extended this approximation model to related issues such as clustering with explicit feature weights [23], hierarchical clustering [24], fuzzy clustering [25], and multicriteria linear stratification [26].

Among his results in this perspective, let me mention a few amazing, even perhaps amusing, but rather unique empirical facts:

(a) At extending k-means to weighted features and Minkowski exponent, at the Iris dataset, much popular in data science, the number of errors reduces to just 5 from conventional 15–17 errors admitted by popular fuzzy and crisp clustering methods [23] – this is comparable to the record found with supervised machine learning algorithms;
(b) At a representative sample of 30 international data scientists, Boris' automatic stratification method over three criteria: (i) citation level, (ii) merit, and (iii) taxonomic rank of the results, gave zero weight to the celebrated Hirsch index and manifested no correlation between the criterion of the scientist's taxonomic rank (iii) and each, (i) citation, and (ii) merit levels [26];
(c) The set of 40 partitions produced at various Minkowski exponent p values (at p running through a sequence of values from $p = 1$ to $p = 4$: $p = 1.0, 1.05, 1.1, \ldots, 3.95, 4.0$) forms an environment to reflect the ground truth partition, which is supposed to be unrelated; this observation is supported by the analysis of synthetic datasets, as well as celebrated Irvine repository datasets [27].

B. Mirkin has described his views on data analysis, in general, and clustering, specifically, in two recent monographs [20, 21(c)].

3 Last Universal Cellular Ancestor (LUCA)

My account of B. Mirkin's main subjects would be far from complete if I fail to mention his results in genomics, as well as a follow-up development. That all started with early work by Boris on interval orders, for which he found a "global" characterization [28] simultaneously with P. Fishburn in the USA who found a "local" characterization [29]. Boris' diploma student participated in a volleyball contest, at which he and his nearest teammate got friendly and had a chat of their respective research projects, which, to their mutual amazement, appeared to be quite similar – one in mathematics, the other in genetics. S. Rodin, the student in genetics, was interested in interval graphs because those were models of procaryote genomes bombarded by mutation-causing DNA products. Joint work by B. Mirkin and S. Rodin generated effective methods [30(a)], as well as successful analyses of genetic data on structure, semantics and evolution of genomic systems described in their monograph [30(b)]. Unfortunately, this work did not bring much harvest to Boris, except for a few questions raised such as: "How come this guy may work for Genomics while being on payroll in Economics?" However, later-on, it is the interval graphs that led Boris to meet Prof. Fred Roberts, then Associate Director of DIMACS, a National Center for Discrete Mathematics and Computer Sciences at the Rutgers University NJ, who also had worked on interval graphs and who helped Boris to obtain two grants from the Office of Naval Research USA (1993–1998). Boris came to the USA because his boss in Moscow had said to him over telephone: "There is a real muddle here in Russia; stay out there as long as you can."

While in DIMACS, Ilya Muchnik, a friend and roommate of Boris, asked his help in interpretation of an algorithm for comparison of different evolutionary trees over the same organisms [31]. Boris said: "This I cannot do because I see no biologically sound idea behind the algorithm. What I can do is to develop a graph-theoretic model for the conventional "gene duplication-independent existence-loss of one copy" explanation of the differences in evolutionary trees for different gene families". He made a ring to the main author of paper [31] asking for explanation of the biological meaning of the algorithm, and after receiving no explanation, went ahead with his own thinking [32(a)]. Later, prompted by Prof.-Dr. M. Vingron (currently Head of Department in Max Plank Institute for Molecular Genetics, Berlin, FRG) who happened to visit DIMACS at that very time, he proceeded to establish equivalence between three different ways of comparison between different gene and species trees [32(b)], including that from paper [31]. This work was noticed by Dr. E. Koonin from NCBI NIH USA who had developed his own approach to the issue, supported by massive genomic data. At that time Boris already was teaching in Birkbeck University of London. Jointly with his colleague from Birkbeck, Prof. T. Fenner, B. Mirkin developed a model and maximum parsimony algorithm for Koonin's approach to gene history reconstruction, which allowed them to reconstruct and interpret the contents of the genome of the very first living organism (572 genes altogether) [33(a)], as well as the ancestral lactic (milk) acid bacteria [33(b)]. Here is the Dr. Eugene Koonin evaluation: "During the 20 years elapsed since the publication of this work, several attempts to reconstruct the gene set of the LUCA using more sophisticated algorithms and many more genomes of prokaryotes have been undertaken, but the results of Mirkin and colleagues do not appear to have been superseded.", see the Book of Abstracts. **International conference Data Analysis, Optimization and their Applications on the occasion of Boris Mirkin's 80th birthday**. MIPT, 2023. Page 23.

Currently Boris, together with T. Fenner and others, works to convert his biologically motivated constructions into modeling such elusive AI concepts as cognitive abstraction/generalization and interpretation. Their first successful results on generalization in taxonomies are published in [34]. Another his big successful project, just completed, is modeling of an oceanic phenomenon, "upwelling", much important for fishing industries, by using invented by him concept of core-shell cluster, jointly with his collaborators from Portugal [35].

4 Who Is Boris Mirkin?

Turning to personal characteristics of B. Mirkin, I would like to cite from an insightful fragment by Dr. Igor Mandel who points out that Boris is always (a) working on issues of his own choice, (b) making fun of joys and sorrows of life, and (c) being tolerant to whoever and whatever occurs [36].

"The best way to learn something about someone's personality is to observe what he or she is doing when circumstances are changing. Most people follow the mainstream, with all its twists. Yet some follow their goals regardless of these fluctuations. As long as I know Boris, he belongs to the minority. He has changed a dozen positions in five or six countries in the turbulent for Russia times from the end of the 1980s, but *one* thing remained *constant*—he continued working and reflected his work in his writing. From the first one, "Group Choice" (1974, in Russian), which elevated him to the top of the analytical community and triggered our meeting, to the latest one "Core Data Analysis" (2019)—he always wrote books of his results interwoven with other international results within corresponding fields. Of course, he has written many articles as well, but his passion to write books seems to me unprecedented. I vividly remember how much it cost me to write just one and can imagine what it is to have produced ten, on different topics, of the very high quality, highly original, and almost all as a single author. One may expect that a person capable of doing that is a kind of gloomy scientist thinking only about writing mandatory number of pages per day and will be way off.

In fact, all our talks started and ended with jokes and laughing, which seems to be the *second constant* element in Boris' life. He has not only permanently produced jokes himself, but vividly reacted to those of others. It was the main reason why most of our scientific discussions quickly went in an unpredictable direction, and ultimately the original topic could disappear entirely (but new ones emerged). As a result, we published only one joint work, while another one is still buried under a pile of jokes for the past 5 years.

The *third*, and the most surprising *constant*, is Boris' tolerance. Since we both are living in what is referred to as "interesting times", what the Chinese would wish to their enemies, I had expected to hear extreme opinions and complaints, from the left, the right, the top, and from the bottom—and I did hear much of those indeed. But never from Boris. His tolerance was not only towards the politics, but in fact towards everything; his belief in good side of all, in general, and of human nature, in particular, I'm sure, is a key in helping him to overcome many troubles and to keep his first and second constants (i.e., writing and laughing) alive. A wonderful painting by Alexander Makhov hangs on the wall in Boris' Moscow apartment—a big fish is spasmodically bent at the beach in the attempt to get off the hook from the fishing line. I definitely saw it as a symbol of tragedy and torture—but Boris suddenly said that he purchased it because he sees there an unshakable will to fight and survive. And I agreed that this interpretation was also possible—actually, might be the only one reasonable."

I would illustrate this third feature of tolerance – should I say "kindness"? – by a story of my own.

A few years ago, I arrived with my wife and three English-speaking children aged 11, 13, and 15 at Moscow to an apartment rented by phone. To place children in a Moscow school, at least temporary registration with police in Moscow of some of the parents is required. I talked to both my acquaintances and long-term friends, each of whom gave me a big NO, "absolutely impossible", followed by a set of useless advices. As a result, the children did not study yet, and I continued to turn to everyone I encountered to request a Moscow registration.

Passing by the office of Prof. Mirkin at HSE NRU, I decided to stop by to say hello. We have known each other for a long time, yet our relations remained rather superficial. Anyway, I walked into the celebrated professor's office with apologetic concern about my registration needs. Boris did not hesitate and despite all my warnings that all my friends, good acquaintances, countrymen told me NO, he said: Ok, I'll think about it and give you my answer tomorrow. The next day, when I called Boris, I received a firm Yes, which was – still is – to me beyond any words of gratitude.

Acknowledgments I would like to thank Prof. Boris Mirkin for sharing with me his memories related to his research career including that in data analysis. Boris Goldengorin's research was supported by Russian Science Foundation project No. 21-71-30005 and Scientific and Educational Mathematical Center «Sofia Kovalevskaya Northwestern Center for Mathematical Research» (agreement № 075-02-2023-937, 16.02.2023)

References

1. Brzozowski, J.A.: BG Mirkin. Novyj algoritm postroéniá bazisa v ázyké régularnyh vyražénij. Izvéstiá Akadémii Nauk SSSR, Téchničeskaá Kibérnétika, no. 5 (1966), pp. 113–119.- BG Mirkin. An algorithm for constructing a base in a language of regular expressions. English translation of the preceding. Engineering Cybernetics, no. 5 (Sept.–Oct. 1966), pp. 110–116. J. Symb. Log. **36**(4), 694–694 (1971)

2. Mirkin, B., Fenner, T.I.: Distance and consensus for preference relations corresponding to ordered partitions. J. Classif. **36**(2), 350–367 (2019)
3. Mirkin, B.G., Cherny, L.B.: On a distance measure between partitions of a finite set. Autom. Remote. Control. **31**(5), 91–98 (1970); Kemeny, J.G.: Mathematics without numbers. Daedalus **88**(4), 577–591 (1959); Kemeny, J.G., Snell, L.J.: Preference ranking: an axiomatic approach. In: Mathematical Models in the Social Sciences, pp. 9–23 (1962)
4. Arrow, K.J.: Social Choice and Individual Values, vol. 12. Yale University Press (2012)
5. Mirkin, B.: Federations and transitive consensus rules. In: Gavrilets, Y. (ed.) Models for Social and Economic Processes and Social Planning. Nauka Publishers, Moscow (1979) (in Russian). See synopsis in English: Mirkin, B.G. Federations and transitive group choice. Math. Soc. Sci. **2**(1), 35–38(1982)
6. Mirkin, B.: Group Choice Issues. Nauka Publishers, Moscow, 256 p. (1974) (in Russian, Проблема группового выбора, Физматгиз, Москва, 1974). English translation: Mirkin, B. Group Choice, P. Fishburn (ed.) Winston and Sons, Washington D.C., 1979, Distributed by Wiley, 252 p
7. Mirkin, B.G. (ed.): Methods for Analysis of Multivariate Economics Data. Nauka Publishers: Siberian branch, 206 p. (1981) (in Russian) (Методы анализа многомерной экономической информации)
8. Kupershtokh, V.L., Trofimov, V.A.: Algorithm for the analysis of the structure of connection matrices. Avtomatika i Telemekhanika. **11**, 170–180 (1975) (in Russian)
9. Vyssotskaya, N., Kupershtokh, V., Polishchuk, L., Trofimov, V., Cherny, L.: Scales of chain ordering. In: Mirkin, B. (ed.) Modeling in Economics Research. Institute of Economics, Novosibirsk, pp. 109–113, in Russian (1978). (Высоцкая, Н. В., Куперштох, В. Л., Полищук, Л. И., Трофимов, В. А., & Черный, Л. Б., Миркин, Б. Г. Шкалы упорядочения. Моделирование в экономических исследованиях)
10. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications. Cambridge University Press, Cambridge (1994)
11. Grenback, G.V., Basareva, V.G., Kupershtokh, V.A., Silchenko, T.A.: Analysis and design of organizational structure for industrial enterprise. In: Aganbegian, A.G. (ed.). Nauka Publishers. Novosibirsk division, 1983, 183 p. (in Russian, Гренбэк Г.В., Басарева В.Г., Куперштох В.Л., Сильченко Т.А. Анализ и формирование организационной структуры промышленного предприятия (вопросы методологии и методики) /отв. ред. А.Г. Аганбегян)
12. Миркин, Б.Г., Родин, С.Н.: Графы и гены, Наука, ФизМатГиз, М. (1977). English translation: Mirkin, B., Rodin, S. Graphs and Genes, Springer (1984)
13. Mirkin, B.G.: Approximation problems in the space of relations and analysis of nonnumerical features. Avtomatika i Telemekhanika. **9**(53–61), 110–118 (1974) (in Russian); (a) Kupershtokh, V.L., Mirkin, B.G., Trofimov, T.A.: Least squares approach in the analysis of categorical features. In: Mirkin B. (ed.) Issues in the Analysis of Discrete Data, Part 2, Novosibirsk, pp. 64–76 (in Russian) (1976). (Куперштох, В. Л., Б. Г. Миркин, and В. А. Трофимов. Метод наименьших квадратов в анализе качественных признаков. Проблемы анализа дискретной информации, часть 2)
14. Mirkin, B.G.: Additive clustering and qualitative factor analysis methods for similarity matrices. J. Classif. **4**(1), 7–31 (1987)
15. Nascimento, S., Casca, S., Mirkin, B.: A seed expanding cluster algorithm for deriving upwelling areas on sea surface temperature images. Comput. Geosci. **85**, 74–85 (2015)
16. (a) Mirkin, B.G., Rostovtsev, P.S.: Linked feature sets, Section 6.3 in Mirkin B.G. Groupings in Socio-economic Research, Finansy I Statistika Publishers, Moscow, 1985, 146–151. (in Russian); (b) Mirkin B.G., Kramarenko A.V.: Approximate bi-cluster and tri-cluster boxes in the analysis of binary data//International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing. Springer, Berlin/Heidelberg, pp. 248–256 (2011)
17. Mirkin, B., Nascimento, S.: Additive spectral method for fuzzy cluster analysis of similarity data including community structure and affinity matrices. Inf. Sci. **183**(1), 16–34 (2012)

18. Shalileh, S., Mirkin, B.: Summable and nonsummable data-driven models for community detection in feature-rich networks. Soc. Netw. Anal. Min. **11**(1), 1–23 (2021)
19. Mirkin, B.G.: A sequential fitting procedure for linear data analysis models. J Classif (Springer). **7**(2), 167–195 (1990)
20. Mirkin, B.: Clustering: A Data Recovery Approach. Taylor and Francis, 1st edition 2005, 2nd edition 2012
21. (a) Mirkin, B.: Braverman's spectrum and matrix diagonalization versus iK-means: a unified framework for clustering. In: Braverman Readings in Machine Learning. Key Ideas from Inception to Current State. LNCS 1100. Springer, Cham, pp. 32–51 (2018); (b) Taran, Z., Mirkin, B.: Exploring patterns of corporate social responsibility using a complementary K-means clustering criterion. Bus. Res. **13**(2), 513–540 (2020); (c) Mirkin, B.: Cluster interpretation aids in mixed scales, In: Mirkin, B (ed.) Core Data Analysis: Summarization, Correlation, and Visualization. Springer, Cham. pp. 338–342 (2019)
22. Mirkin, B. (ed.): Models for Aggregating Socio-economic Data. Novosibirsk, Institute of Economics, 173 p. (1978) (In Russian, Модели агрегирования социально-экономической информации)
23. De Amorim, R.C., Mirkin, B.: Minkowski metric, feature weighting and anomalous cluster initializing in K-means clustering. Pattern Recogn. **45**(3), 1061–1075 (2012)
24. Kovaleva, E.V., Mirkin, B.G.: Bisecting K-means and 1D projection divisive clustering: a unified framework and experimental comparison. J. Classif. **32**(3), 414–442 (2015)
25. Nascimento, S., Mirkin, B., Moura-Pires, F.: Modeling proportional membership in fuzzy clustering. IEEE Trans. Fuzzy Syst. **11**(2), 173–186 (2003)
26. (a) Mirkin, B., Orlov, M.: Three aspects of the research impact by a scientist: measurement methods and an empirical evaluation. In: Optimization, Control, and Applications in the Information Age. Springer, Cham, pp. 233–259 (2015); (b) Murtagh, Fionn, Michael Orlov, and Boris Mirkin. Qualitative judgement of research impact: domain taxonomy as a fundamental framework for judgement of the quality of research. J. Classif. **35**(1) 5–28 (2018)
27. de Amorim, R.C., Shestakov, A., Makarenkov, V., Mirkin, B.: The Minkowski central partition as a pointer to a suitable distance exponent and consensus partitioning. Pattern Recogn. **67**, 62–72 (2017)
28. (a) Mirkin, B.G.: An axiom of mathematical utility theory. Cybernetics **6**(6), 776–779 (in Russian) (1970); (b) Mirkin, B.G.: Description of some relations on the set of real-line intervals. J. Math. Psychol. **9**(2), 243–252 (1972)
29. Fishburn, P.C.: Intransitive indifference with unequal indifference intervals. J. Math. Psychol. **7**(1), 144–149 (1970)
30. (a) Mirkin, B.G., Rodin, S.N.: Analysis of Boolean matrices related to the solution of certain genetic problems. Cybernetics **10**(2), 318–324 (1974); (b) Mirkin, B.G., Rodin, S.N.: Graphs and Genes, M., Nauka Publishers, 1977, 237 p. (in Russian). English translation: Graphs and Genes, Springer, Heidelberg (1984)
31. Goodman, M., Czelusniak, J., Moore, G.W., Romero-Herrera, A.E., Matsuda, G.: Fitting the gene lineage into its species lineage, a parsimony strategy illustrated by cladograms constructed from globin sequences. Syst. Biol. **28**(2), 132–163 (1979)
32. (a) Mirkin, B, Muchnik, I, Smith, T.F.: A biologically consistent model for comparing molecular phylogenies. J. Comput. Biol. **2**(4), 493–507 (1995); (b) Eulenstein, O., Mirkin, B., Vingron, M.: Comparison of annotating duplication, tree mapping, and copying as methods to compare gene trees with species. In: Mathematical Hierarchies and Biology: DIMACS Workshop, November 13–15, 1996. Vol. 37. American Mathematical Soc. (1997)
33. (a) Mirkin, B.G., Fenner, T.I., Galperin, M.Y., Koonin, E.V.: Algorithms for computing parsimonious evolutionary scenarios for genome evolution, the last universal common ancestor and dominance of horizontal gene transfer in the evolution of prokaryotes. BMC Evol. Biol. **3**(1), 1–34 (2003); (b) Makarova, K., Slesarev, A., Wolf, Y., Sorokin, A., Mirkin, B., Koonin, E., ... 43 co-authors ... & Mills, D.: Comparative genomics of the lactic acid bacteria. Proc. Natl. Acad. Sci. **103**(42), 15611–15616 (2006)

34. (a) Mirkin, B., Nascimento, S., Pereira, L.M.: Cluster-lift method for mapping research activities over a concept tree. In: Advances in Machine Learning II, pp. 245–257. Springer, Berlin/Heidelberg; (b) Frolov, D., Nascimento, S., Fenner, T., Mirkin, B.: Parsimonious generalization of fuzzy thematic sets in taxonomies applied to the analysis of tendencies of research in data science. Inf. Sci. **512**, 595–615 (2020)
35. Nascimento, S., Martins, A., Relvas, P., Luís, J.F., Mirkin, B.: Novel cluster modeling for the spatiotemporal analysis of coastal upwelling. In: EPIA conference on artificial intelligence, pp. 563–574. Springer, Cham (2022)
36. Mandel, I.: Three and one questions to Dr. B. Mirkin about complexity statistics. In: Clusters, Orders, and Trees: Methods and Applications, pp. 1–9. Springer, New York (2014)

Manipulability of Aggregation Procedures for the Case of Large Numbers of Voters

Alexander Ivanov

1 Introduction

Manipulability occurs when during a voting an agent misrepresents his/her preferences and gets a better outcome of the procedure.

It was proven in Gibbard [10] and Satterthwaite [19] that for the case of single-valued social choice every non-dictatorial aggregation procedure is manipulable. Then, Duggan and Schwartz [8] showed the same result for the case of multi-valued choice when ties between alternatives are possible. Thus, a question arises: which aggregation procedure is the least manipulable one?

Since then, various papers studying manipulability have been published, a non-exhaustive list includes [2, 6, 7, 9, 16, 18, 20, 21].

Two approaches have been used in most papers. The first one is to look for an analytical solution, i.e., to find a formula for a certain manipulability index for a certain aggregation procedure [9, 16]. The main difficulty is that analytical solutions are usually found only for most popular positional rules, for example, Plurality rule, Inverse Plurality rule, Borda's rule. However, it is known that such rules are more manipulable [2], and the less manipulable aggregation procedures, for example, Hare's procedure, Nanson's procedure [3] are more complicated to find an analytical formula for them.

That is why there is the second approach: computer modeling. One of the first papers in this area [2] used computer modeling to generate all possible profiles for small cases: 3 . . . 5 alternatives and 3 . . . 10 agents. Later, researchers started generating a large random number of profiles (one million) to get approximate

A. Ivanov (✉)
National Research University Higher School of Economics (NRU HSE), Moscow, Russia

Institute of Control Sciences of Russian Academy of Sciences (ICS RAS), Moscow, Russia

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_10

estimations of manipulability indices. Such an approach allowed to estimate the degree of manipulability of many aggregation procedures without deriving any analytical formulae for the cases of 3 . . . 100 agents ([1, 3]; Aleskerov et al. [4]).

The results from computer modeling showed that the least manipulable aggregation procedures in most cases are Inverse Borda's rule, Nanson's procedure and Hare's procedure [3].

However, some questions have arisen from the results for the case of 3 . . . 100 agents. For example, if there is no least manipulable aggregation procedure for the case of small number of agents, maybe there is one for the case of large number of agents? It was shown in Aleskerov and Kurbanov [2] and Aleskerov et al. [3] that the manipulability indices for even and odd numbers of agents may vary 3–5 times. Does that result hold for larger number of agents?

In this paper, we address these and other questions by comparing the previously known results in literature from computer modeling for the case of 3 . . . 100 agents with our new results from computer modeling for the cases of 3 . . . 10,000 agents for the case of Impartial Culture and 3 alternatives.

2 Main Notions

2.1 Aggregation Procedures

We use similar notation as in Aleskerov and Kurbanov [2] and Aleskerov et al. [3]. We denote the number of agents as n, and the number of alternatives as m. Each agent has a preference (linear order) over the set of alternatives. For the case of n agents there are $m!$ possible preferences. The set of agents with their preferences over the set of alternatives comprise a profile, P. An aggregation procedure $C(P)$ determines the winner taking a profile P as an input.

We consider nine scoring aggregation procedures and use the same definitions as in Aleskerov and Kurbanov [2] and Aleskerov et al. [3]:

1. Plurality rule

   The alternative which is the first best for the largest number of agents is chosen.
2. q-Approval rule with q = 2

   The alternative which is the first best or the second best for the largest number of agents is chosen.
3. Borda's rule

   For each alternative Borda's count is calculated: for each agent for whom it is first best, the alternative gets $m - 1$ points, for each agent for whom it is second best, the alternative gets $m - 2$ points . . . for each agent for whom it is m-th best, the alternative gets 0 points. The result of the aggregation procedure is the alternative with the highest Borda's count, i.e., the alternative(s) with the highest sum of points.

4. Black's procedure

   The procedure chooses the unique Condorcet winner if it exists, and uses Borda's rule otherwise.
5. Threshold rule

   The alternative which is the worst for the smallest number of agents is chosen. If there are more than one such alternatives, the number of agents for whom they are second-worst alternatives are compared, etc., until the winner is found.
6. Hare's procedure

   If there is an alternative which is the first best for the simple majority of agents, it is the winner. Otherwise, the alternative with the least number of first-best votes is eliminated, and the procedure repeats.
7. Inverse Borda's rule

   Borda's count is calculated for all alternatives. The alternative with the lowest Borda's count is eliminated, and the procedure repeats until the winner is found.
8. Nanson's procedure

   Borda's count is calculated for each alternative. Then, the average Borda's count among alternatives is calculated. The alternatives with Borda's count less than the average are eliminated, and the procedure repeats.
9. Coombs' procedure

   The alternative which is the worst for the largest number of agents is eliminated. Procedure repeats until the winner is found.

In Aleskerov et al. [3] 10 aggregation procedures are studied. In addition to 9 aggregation procedures described above, Inverse Plurality rule is included. However, for the case of 3 alternatives which we consider in this paper, Inverse Plurality rule has the same definition as q-Approval rule with q = 2.

2.2 Manipulation and Manipulability Index (NK-Index)

If P is a profile where all agents cast their sincere preferences, and $P^{'}$ is a profile, where an agent misrepresents her preferences, then manipulation happens if $C(P^{'})$ $\succ C(P)$ for the manipulating agent.

Every profile is marked either as manipulable or non-manipulable. A profile is called manipulable, if there exists at least one way of successful manipulation by at least one agent. If there are no ways for any agent to misrepresent her preferences and to get a better outcome, then the profile is non-manipulable.

The most widely used index to estimate the degree of manipulability of an aggregation procedure is Nitzan-Kelly index (NK-index) which was introduced in Nitzan [17], Kelly [14, 15]. NK-index stands for the share of manipulable profiles in the total number of profiles:

$$NK = \frac{number\ of\ manipulable\ profiles}{total\ number\ of\ \ profiles}$$

2.3 Extended Preferences

What happens in the case of a tie between two or more alternatives in the aggregation procedure? One approach is to use an additional tie-breaking rule, for example, alphabetical tie-breaking rule. This approach, however, is not neutral to the set of alternatives. For this reason, we use another approach known in the literature, namely, the so-called extended preferences. Such an approach allows ties between alternatives: if there is a tie, then all such alternatives constitute the social choice. For example, if both alternatives $\{a\}$ and $\{c\}$ under Plurality rule have equal number of votes, then the result of the procedure is $\{a, c\}$. It means that the result of an aggregation procedure $C(P)$ is a non-empty subset of the set of alternatives (multi-valued choice). Then, extended preferences are used to allow an agent to compare two multi-valued choices (the results of the aggregation procedure before and after a manipulation attempt). Detailed research regarding preferences extension axioms and extended preferences can be found in Barberà et al. [7].

We consider four ways of constructing extended preferences (EP) for the case of three alternatives: Leximin, Leximax, Risk-lover, Risk-averse. Their definitions are the same as in Aleskerov et al. [3].

Suppose that an agent has the sincere preferences $a \succ b \succ c$. Then, we consider four EPs:

1. Leximin: multi-valued choices are compared alphabetically, and the choice where the worst alternative is better is more preferred. EP under Leximin (three multi-valued choices where the ordering is different among four EPs are underlined):

$$\{a\} \succ \{a, b\} \succ \underline{\{b\} \succ \{a, c\} \succ \{a, b, c\}} \succ \{b, c\} \succ \{c\}$$

2. Leximax: multi-valued choices are compared alphabetically, and the choice where the best alternative is better is more preferred. EP under Leximax:

$$\{a\} \succ \{a, b\} \succ \underline{\{a, b, c\} \succ \{a, c\} \succ \{b\}} \succ \{b, c\} \succ \{c\}$$

3. Risk-averse ("PWorst" in Aleskerov et al. [3]: multi-valued choices are compared by the probability of the worst alternative, and the choice where the probability of the worst alternative alphabetically, and the choice, where probability of the worst alternative is lower, is better. EP under Risk-averse:

$$\{a\} \succ \{a, b\} \succ \underline{\{b\} \succ \{a, b, c\} \succ \{a, c\}} \succ \{b, c\} \succ \{c\}$$

4. Risk-lover ("PBest" in Aleskerov et al. [3]: multi-valued choices are compared by the probability of the best alternative, and the choice where the probability of the best alternative, and the choice, where probability of the best alternative is higher, is better. EP under Risk-lover:

$$\{a\} \succ \{a, b\} \succ \underline{\{a, c\} \succ \{a, b, c\} \succ \{b\}} \succ \{b, c\} \succ \{c\}$$

For example, if in some profile an agent has sincere prefenrece $a \succ b \succ c$, the result with sincere preferences is $\{a, c\}$ and the result with insincere preferences is $\{b\}$, then it is manipulation for Leximax EP and Risk-lover EP, and there is no manipulation for Leximin EP and Risk-averse EP.

2.4 Impartial Culture and Computer Modeling Scheme

What is the probability of a certain profile? We use Impartial Culture (IC) probabilistic model which was first studied in Guilbaud [11]. Under IC all preferences are equally likely. If each of n agents has one of $m!$ preferences, the total number of profiles under IC is equal to $(m!)^n$.

It can be noticed, that the number of profiles is growing very fast. Even a modern supercomputer cannot generate all possible profiles for cases of $n > 30$. That is why computer modelling papers use random profile generation. Instead of generating all possible profiles which is impossible for large n, 1,000,000 profiles are generated randomly. As it was shown in Karabekyan [13], such an approach allows to estimate the degree of manipulability of aggregation procedures with the precision of 0.001 in manipulability indices. We use this approach for estimating manipulability indices.

As a result, the computer modelling scheme to estimate the degree of manipulability of a given aggregation procedure for a given number of agents and for a given number of alternatives usually consists of the following steps:

1. Random 1,000,000 profiles are generated
2. For each profile, the sincere choice is calculated
3. For each profile each possible manipulation attempt by each agent is generated, i.e. for each of n agents $m\,! - 1$ ways of misrepresenting preferences are considered
4. If there is at least one successful manipulation attempt for at least one agent, the profile is marked as manipulable. If all possible manipulation attempts lead to the same or worse choice for manipulating agents, the profile is marked as non-manipulable.
5. After all profiles are analyzed, manipulability indices are calculated.

It can be noticed that the computational complexity of the scheme is high. For each of the randomly generated 1,000,000 profiles all possible $n_* (m\,! - 1)$ manipulation attempts should be checked, meaning we should calculate the new result of the aggregation procedure $C(P^{'})$ for each attempt. Finally, the algorithm repeats for every n. Detailed estimations of the complexity of the algorithms of computer modelling for calculating manipulability indices can be found in Ivanov [12]. The estimations for the cases of up to 10,000 agents took 3 weeks on one personal computer.

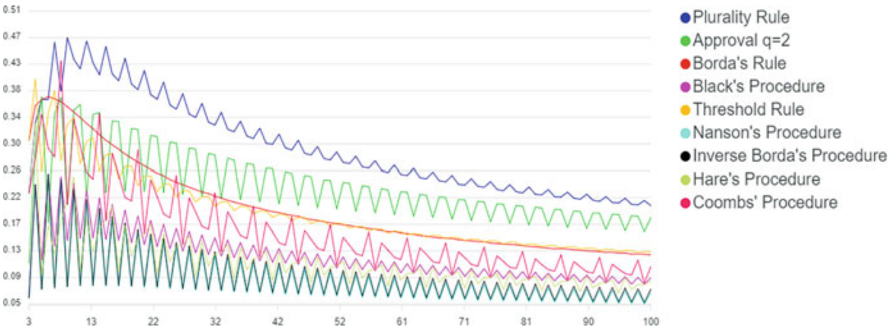
2.5 *Results*

We obtained the results for the case of $m = 3$ alternatives and for $n = 3 \ldots 10{,}000$ agents for the cases of Leximin, Leximax, Risk-averse and Risk-lover EP for Impartial Culture.

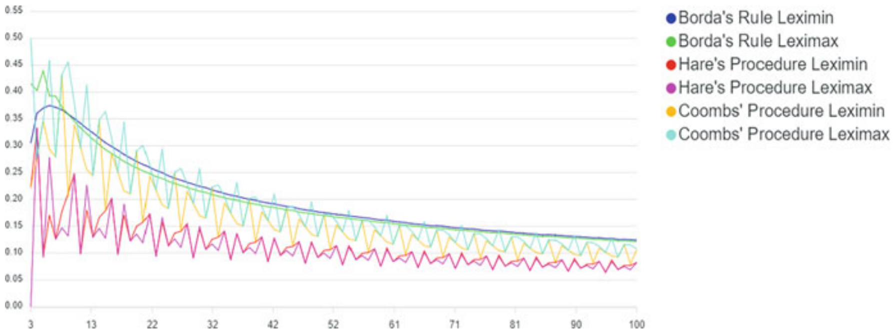
First, we provide the results for the case of $n = 3 \ldots 100$ agents. In Aleskerov et al. [3] the results for similar cases are provided, but with gaps: they calculated cases of n = 3 . . . 25, then n = 29, 30, 39, 40, etc. Picture 1 fills the missing picture for the cases of 31 . . . 38, 41 . . . 48, etc. agents.

The shapes of the lines suggest that there are certain periods in the values of NK-index. The idea that there are such periods in the values of NK-index was previously discussed in the literature, for example, in Aleskerov et al. [3] it is suggested that the period for most of the 9 discussed procedures is either 2 or 3. Having the chart for all n from 3 to 100, we can confirm that the periods for Black's procedure, Nanson's procedure, Inverse Borda's procedure are equal to 2, for Plurality rule, q-Approval q = 2 rule and Threshold rule periods are equal to 3, but we can suggest that the periods for Coomb's procedure and Hare's procedure are equal to 6, while NK-index for Borda's rule does not have any visible periods. This conclusion holds not only for Leximin EP, but also for other extended preferences as well. For example, Picture 2 illustrates the three rules (Borda's rule with no period, Coomb's rule and Hare's procedure with the period of 6) for both Leximin and Leximax EP.

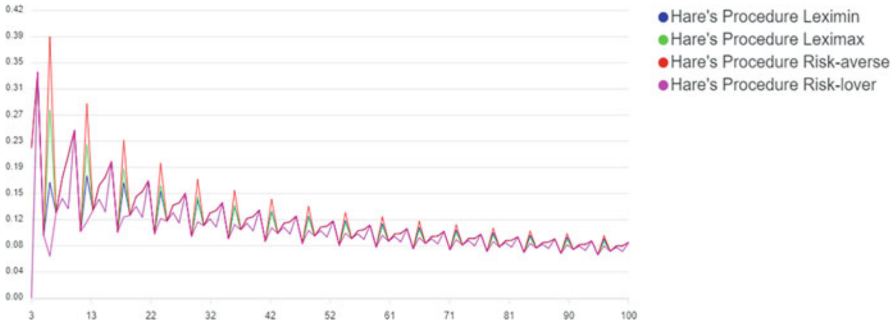
Next, we would like to verify the hypothesis from Aleskerov et al. [3] that both Leximin EP and Risk-averse EP show the same values of NK-index, and both Leximax and Risk-lover EP show the same values of NK-index for most rules. Such a result was presented in Aleskerov et al. [3] for the cases of $n \leq 100$ agents. We studied the cases of $100 < n \leq 10{,}000$ agents, and found out that for 5 procedures (Plurality, q-Approval with q = 2, Borda, Threshold and Coombs' procedures) it is true, but for other 4 procedures (Nanson, Inverse Borda, Black and Hare's procedures) there are differences in the values of NK-index between Leximin EP and Risk-averse EP, Leximax EP and Risk-lover EP.



Picture 1 NK-index, Leximin, 3–100 agents



Picture 2 Leximin and Leximax EP for Borda's rule, Coombs and Hare's procedures



Picture 3 NK-index for Hare's procedure for four types of extended preferences

For example, Picture 3 shows the results for Hare's procedure for all four types of extended preferences, and it can be noticed, that, for example, for the case of 77 agents NK-index is the same, but for 78 agents NK-index is different for all four extended preferences.

The next idea is to evaluate the difference in NK-index for four extended preferences. For small numbers of agents we can notice that the difference in NK-index may be large, for example, NK-index for Hare's procedure for 18 agents for Risk-lover EP is equal to 0.12, while for Risk-averse EP NK = 0.23, almost twice higher. Let us compare the differences for larger numbers of agents. For n = 90: NK = 0.0788 for Risk-lover and NK = 0.0954 for Risk-averse (21% higher); for n = 990: NK = 0.0253 for Risk-lover and NK = 0.0265 for Risk-averse (4.7% higher); for n = 9990: NK = 0.008043 for Risk-lover and NK = 0.008136 for Risk-averse (1.1% higher). The difference between values of the NK-index for different extended preferences decreases with growing number of agents, but still exists.

From Aleskerov et al. [3] and Aleskerov et al. [4] we know that there are 3 aggregation procedures which are the least manipulable for most cases for $n = 3 \dots 100$ agents: Hare's procedure, Nanson's procedure and Inverse Borda's procedure. Let us compare the values of NK-index for these rules for the cases of large number of agents (Picture 4)

It can be noticed, that for large number of agents Nanson’s procedure shows the smallest values of NK-index, i.e., is the least manipulable. Inverse Borda’s procedure is more manipable than Nanson’s procedure, but less manipulable than Hare’s procedure.

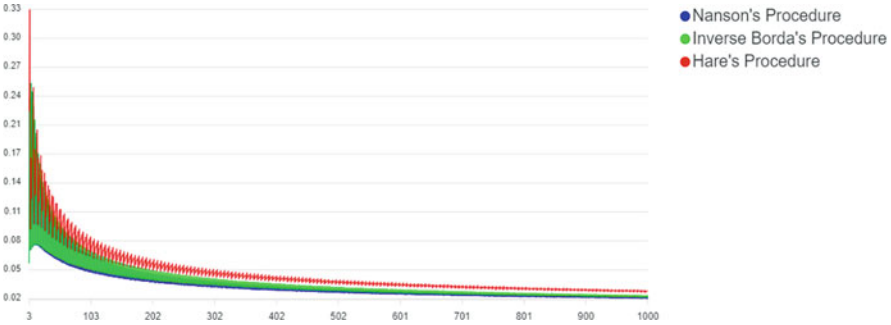
In order to have a better look at the difference between manipulability of the aggregation procedure, we suggest the following chart (Picture 5)

Horizontal axis stands for the number of agents, while the vertical axis stands for the ratio between a certain rule and the least manipulable rule (Nanson’s procedure).

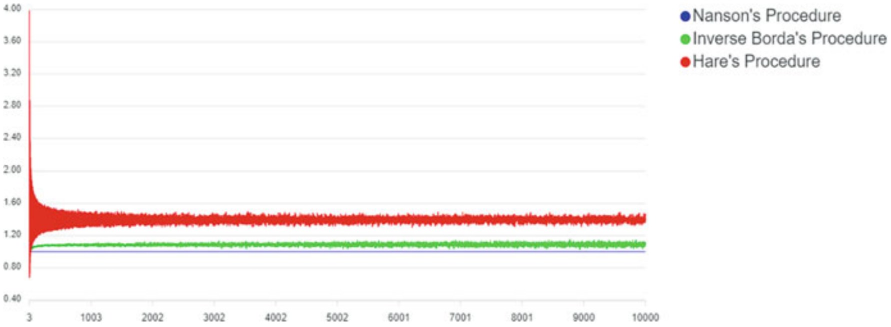
The line for the Nanson’s procedure (blue line) is given as a reference line (NK-index for Nanson’s procedure divided by NK-index for Nanson’s procedure is always equal to 1), because it allows to see that other procedures (e.g., Hare’s procedure) may be less manipulable than Nanson’s procedure only for small numbers of agents. According to our calculations, for the cases of $n \leq 26$ agents Hare’s procedure is sometimes less manipulable than Nanson’s procedure, but for all cases of $n > 26$ agents Nanson’s procedure is the least manipulable for Leximin EP.

Next, we analyze the results for Leximax EP (Picture 6)

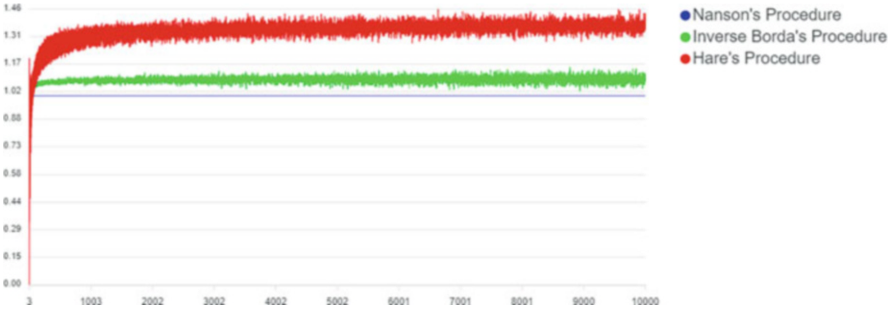
For the case of Leximax EP, for small numbers of agents Hare’s procedure is often less manipulable than Nanson’s procedure. For example, for the cases of



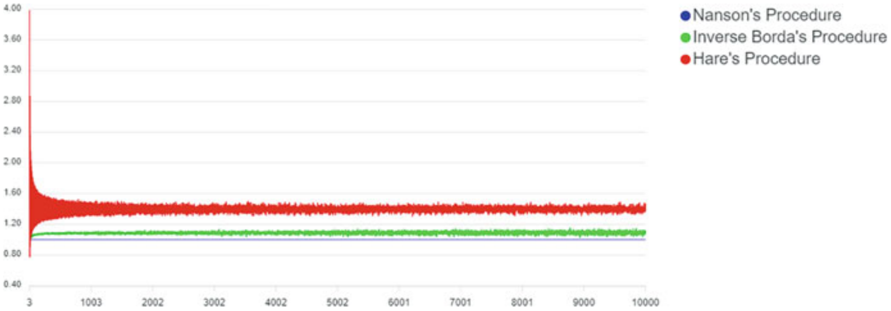
Picture 4 Nanson’s, Inverse Borda’s and Hare’s procedures for Leximin EP



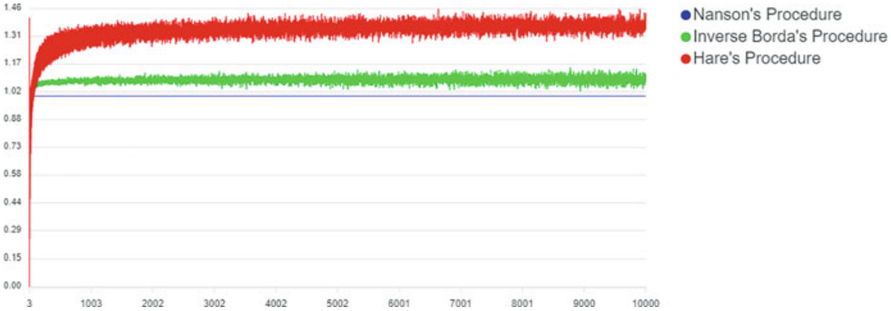
Picture 5 NK-index ratios to Nanson’s procedure NK-index, Leximin EP



Picture 6 NK-index ratios to Nanson's procedure NK-index, Leximax EP



Picture 7 NK-index ratios to Nanson's procedure NK-index, Risk-averse EP



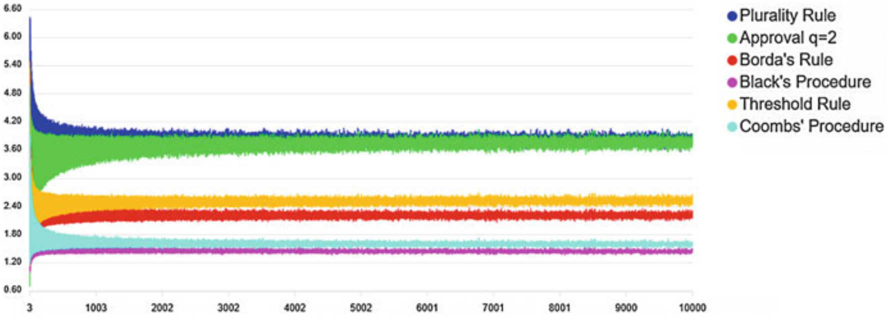
Picture 8 NK-index ratios to Nanson's procedure NK-index, Risk-lover EP

3 . . . 68 agents Hare's procedure is less manipulable in 34 out of 66 cases. However, Nanson's procedure is the least manipulable for all cases of $n > 68$ agents.

Picture 7 shows the case of Risk-averse EP.

Again, for small n Hare's procedure sometimes is the least manipulable, but Nanson's procedure is the least manipulable for all cases of $n > 20$ agents.

Picture 8 shows the case of Risk-lover EP.



Picture 9 NK-index ratios to Nanson's procedure NK-index, Leximin EP (remaining 6 procedures)

Table 1 Ratios between NK-index for a given procedure to Nanson's procedure's NK-index

Procedure	Leximin	Leximax	Risk-averse	Risk-lover
Plurality rule	3.81	3.69	3.81	3.69
q-approval, q = 2	3.70	3.65	3.70	3.65
Borda's rule	2.22	2.14	2.22	2.14
Black's procedure	1.45	1.4	1.45	1.40
Threshold rule	2.50	2.43	2.50	2.43
Inverse Borda's rule	1.09	1.08	1.09	1.08
Hare's procedure	1.39	1.35	1.40	1.34
Coomb's procedure	1.61	1.61	1.61	1.61

We see the same situation: for the cases of $n \leq 74$ Hare's procedure sometimes is the least manipulable, but for $n > 74$ Nanson's procedure is the least manipulable.

The next interesting result is that the ratio between the values of NK-index for aggregation procedures converges with growing n. In Pictures 5, 6, 7, and 8 we saw that result for Hare's procedure and Inverse Borda's procedure, Picture 9 shows the rest of the considered procedures (Plurality rule, Borda's rule, Black's procedure, Threshold rule, q-Approval rule with q = 2, Coombs procedure) in the same type of the chart: ratio of their NK-index to the NK-index of Nanson's procedure.

In Table 1 we provide summarized results as the average ratios between each procedure's NK index and Nanson's procedure's NK index.

For example, Plurality rule has in average 3.81 times larger values of NK-index than Nanson's procedure for the same n.

It can be noticed that the second least manipulable procedure is Inverse Borda's procedure which is in average 1.08–1.09 times more manipulable than Nanson's procedure. Such close results may be explained by similar mechanisms of the procedures: in both procedures Borda's count is calculated, and in Nanson's procedure alternatives with less than average count are eliminated, while in Inverse Borda's procedure only the alternative with the lowest Borda's count is eliminated.

Acknowledgments This work/article is an output of a research project implemented as part of the Basic Research Program at the National Research University Higher School of Economics (HSE University).

The author would like to thank F. Aleskerov and D. Karabekyan for their valuable feedback on this research.

References

1. Aleskerov, F., Ivanov, A., Karabekyan, D., Yakuba, V.: Manipulability of majority relation-based collective decision rules. In: Czarnowski, I., Howlett, R., Jain, L. (eds.) Intelligent Decision Technologies 2017. IDT 2017 Smart innovation, systems and technologies, vol. 72. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-59421-7_8
2. Aleskerov, F., Kurbanov, E.: Degree of manipulability of social choice procedures. In: Alkan, A., et al. (eds.) Current Trends in Economics Studies in economic theory, vol. 8, pp. 13–27. Springer, Berlin/Heidelberg (1999)
3. Aleskerov, F., Karabekyan, D., Sanver, R., Yakuba, V.: On manipulability of positional voting rules. Ser. J. Span. Econ. Assoc. **2**, 431–446 (2011)
4. Aleskerov, F.T., Ivanov, A., Karabekyan, D., Yakuba, V.I.: Manipulability of aggregation procedures in impartial anonymous culture. Procedia Computer Science. **55**, 1250–1257 (2015). https://doi.org/10.1016/j.procs.2015.07.133
5. Barberà, S., Bossert, W., Pattanaik, P.K.: Ranking sets of objects. In: Barberà, S., Hammond, P.J., Seidl, C. (eds.) Handbook of Utility Theory. Springer, Boston (2004). https://doi.org/10.1007/978-1-4020-7964-1_4
6. Chamberlin, J.R.: An investigation into the relative manipulability of four voting systems. Behav. Sci. **30**(4), 195–203 (1985)
7. Diss, M., Tsvelikhovskiy, B.: Manipulable outcomes within the class of scoring voting rules. Math. Soc. Sci. **111**, 11–18 (2021). https://doi.org/10.1016/j.mathsocsci.2021.02.002
8. Duggan, J., Schwartz, T.: Strategic manipulability without resoluteness or shared beliefs: Gibbard–Satterthwaite generalized. Soc. Choice Welf. **17**, 85–93 (2000)
9. Favardin, P., Lepelley, D.: Some further results on themanipulability of social choice rules. Soc. Choice Welf. **26**, 485–509 (2006)
10. Gibbard, A.: Manipulation of voting schemes. Econometrica. **41**, 587–601 (1973)
11. Guilbaud, G.T.: Les theories de l'intérêt general et le probleme logique de l'agregation. Econ. Appl. **5**, 501–584 (1952)
12. Ivanov, A.A.: On efficient schemes of estimating the degree of manipulability of aggregation procedures. J. Inf. Technol. Comput. Syst. **2**, 38–50 (2020) (in Russian)
13. Karabekyan, D.: The problem of manipulability in the social choice theory. PhD thesis (in Russian) (2012)
14. Kelly, J.: Almost all social choice rules are highly manipulable, but few aren't. Soc. Choice Welf. **10**, 161–175 (1993)
15. Kelly, J.S.: 4. Minimal manipulability and local strategy-proofness. Soc. Choice Welf. **5**, 81–85 (1988)
16. Lepelley, D., Valognes, F.: Voting rules, manipulability and social homogeneity. Public Choice. **116**(1/2), 165–184 (2003)
17. Nitzan, S.: The vulnerability of point-voting schemes to preference variation and strategic manipulation. Public Choice. **47**, 349–370 (1985)
18. Pritchard, G., Wilson, M.C.: Exact results on manipulability of positional voting rules. Soc. Choice Welf. **29**(3), 487–513 (2007)
19. Satterthwaite, M.: Strategy-proofness and Arrow's conditions: existence and correspondence theorems for voting procedures and social welfare functions. J. Econ. Theory. **10**, 187–217 (1975)

20. Slinko, A.: On asymptotic strategy-proofness of the plurality and the run-off rules. Soc. Choice Welf. **19**(2), 313–324 (2002)
21. Veselova, Y.: The difference between manipulability indices in the IC and IANC models. Soc. Choice Welf. **46**, 609–638 (2016). https://doi.org/10.1007/s00355-015-0930-3

Preferences over Mixed Manna

Alexander Karpov

1 Introduction

A preference model is an important building block of economic and political science theories. In the wide-spread Arrovian approach [1, 2, 20] each agent has a preference relation that is represented by a linear order over a set of alternatives.

A preference domain (subset of linear orders over a finite set) is a basic preference model. It is natural to assume that a society does not contain all possible preference orders. It is assumed that the set of alternatives is somehow structured, and this structure leads to a domain of structured preferences. The axis of alternatives that leads to single-peaked preferences is the most well-known such example. Single-peaked on a circle preferences [24], single-peaked on a tree preferences [7], Euclidean preferences [23], median spaces preferences [21], group-separable preferences [13] are examples of structured preferences that are specified by a structure of alternatives.

The literature on structured preferences (see surveys [9, 14, 28]) focuses almost exclusively on domains over the set of desirable alternatives. The most influential example of such a model is single-peaked preferences. All alternatives and agents' ideal points are placed on a line (axis). This line implies only an order, but not a distance between alternatives. Alternatives that are further from the agent's ideal point are worse for the agent. By this rule, we can compare alternatives only from one side of the ideal point. A domain of single-peaked preferences is a set of preference orders that are consistent with a given axis of alternatives.

A. Karpov (✉)
HSE University, Moscow, Russia

Institute of Control Sciences, Russian Academy of Sciences, Moscow, Russia
e-mail: akarpov@hse.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer
Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_11

Condorcet domains are sets of linear orders with the property that, whenever the preferences of all agents belong to this set, the majority relation induced by the preference profile with an odd number of voters has no cycles. The single-peaked domain is a Condorcet domain.

Considering all possible domains, Puppe [26] showed that the single-peaked domain is the only minimally rich and connected Condorcet domain that contains two completely reversed preference orders. Minimal richness condition requires that each alternative is a top alternative in at least one preference order from the domain. The minimal richness is a key property that reflects the desirability of alternatives.

In some cases, alternatives are undesirable outcomes, i.e. possible places for potentially polluting factories. In this case, single-dipped preferences is a suitable model. They are defined in the same fashion as the single-peaked preferences. There is a line (axis); the further from the agent's worst point the better the alternative is. For each alternative and each single-dipped domain, there is a preference order from the domain such that this alternative is a bottom alternative in this preference order.

Mixing goods and bads is a natural development of economic models (see e.g. [3]). In the case of preference models, this mixing leads to a weak minimal richness condition. It requires that each alternative is either the top, or bottom alternative in at least one preference order from the domain.

This paper describes the set of connected and weakly minimally rich Condorcet domains that contain two completely reversed preference orders. These preferences have the following common structure. There is a linear ordering of alternatives and a partition of alternatives on "inside" and "outside" alternatives. For each triple of alternatives, if the median of this triple, according to the linear ordering, is an inside alternative, then the restriction of the domain to this triple is single-peaked with a given axis. If the median of this triple, according to the linear ordering, is an outside alternative, then the restriction of the domain to this triple is single-dipped with a given axis. We call these domains GF-domains. The single-peaked domain, the single-dipped domain, and the Fishburn's domain [10] are examples of GF-domains. We have shown that all GF-domains are subsets of single-peaked on a circle domains. Single-peaked on a circle preferences [24] are straightforward generalization of single-peaked preferences. The main shortcoming of single-peaked on a circle preferences is that they do not guarantee the transitivity of collective preferences obtained by the majority rule. Restriction to GF-domain solves this problem. Each GF-domain is a Condorcet domain.

Single-peaked on a circle preferences are a finite population generalization of the circular city model [29]. This model is a basic spatial firm competition model [31]. The circular city model is also applied in political science [22]. In this case, far-left and far-right candidates are close to each other. Extremist candidates can be considered as outside alternatives for the centrist policy electorate. GF-domain is an appropriate model for the centrist policy electorate.

Another model that combines goods and bads is the structured dichotomous preferences model [8, 30]. In this case each agent partitions the set of alternatives in two subsets: approved alternatives (goods) and disapproved alternatives (bads).

The structure of the paper is as follows. Section 2 contains the main result concerning characterization of GF-domains. Section 3 discusses possible interpretations of GF-domains. Section 4 concludes.

2 Model

Let a finite set $X = \{1, \dots, m\}$ be the set of alternatives, and a finite set $N = \{1, \dots, n\}$ be the set of agents. Each agent $i \in N$ has a preference order P_i over X (each preference order is a linear order). Let $L(X)$ be the set of all linear orders over X. An n-tuple of preference orders is a preference profile $\mathcal{P} = (P_1, \dots, P_n) \in L(X)^n$. For brevity, we will write preference order as a string, e.g. $12 \dots m$, which means $1P2P3 \dots Pm$.

A subset of preference orders $D \subseteq L(X)$ is called a *domain* of preference orders. A domain D is a *Condorcet domain* if whenever the preferences of all agents belong to the domain, the majority relation of any preference profile with an odd number of agents is transitive. A Condorcet domain D is *maximal* if every Condorcet domain (on the same set of alternatives) that contains D as a subset coincides with D.

Each domain of single-peaked preferences is defined by an axis. An axis is a linear order over X. A preference profile P is *single-peaked* with respect to axis α, if agent i's upper-contour sets $U(P_i, x) = \{y \in X | y P_i x\}$ are connected according to the axis (this means that for any two elements from this set all elements between them according to the axis belong to this set). A preference profile P is *single-dipped* with respect to axis α, if agent i's the lower-contour sets $L(P_i, x) = \{y \in X | x P_i y\}$ are connected according to the axis.

A Condorcet domain D is *connected* if every two orders from the domain can be obtained from each other by a sequence of transpositions of neighboring alternatives such that the resulting order belongs to the domain at each step. A Condorcet domain D is *semi-connected* if it contains two completely reversed orders and an entire path connecting them. A Condorcet domain has *maximal width* if it contains a pair of completely reversed linear orders.

A Condorcet domain D is *minimally rich* if, for each alternative $x \in X$, there is an order $P \in D$ such that P has x as a top alternative. A Condorcet domain D is *weakly minimally rich* if, for each alternative $x \in X$, there is an order $P \in D$ such that P has x as either top or bottom alternative.

A domain D is a *peak-pit* domain if, for each triple of alternatives, the restriction of the domain to this triple is either single-peaked, or single-dipped.

A domain D is called a *Fishburn's domain* if it satisfies the *alternating scheme* [10]: there exists a linear ordering of alternatives $a_1, \dots, a_m$ such that for all i, j, k with $1 \leq i < j < k \leq m$ the restriction of the domain to the set $\{a_i, a_j, a_k\}$ is single-peaked with axis $a_i a_j a_k$ if j if is even (odd), and it is single-dipped with axis $a_i a_j a_k$ if j is odd (even). The reverse of Fishburn's domain is also a Fishburn's domain.

Having the natural ordering of alternatives we obtain the following Fishburn's domains in the case of four and five alternatives:

$F_4 = \{1234, 1243, 2134, 2143, 2413, 2431, 4213, 4231, 4321\}$,
$F_5 = \{12345, 12354, 13245, 13254, 13524, 13542, 31245, 31254, 31524, 31542, 35124, 35142, 35412, 35421, 53124, 53142, 53412, 53421, 54312, 54321\}$.

In the example with five alternatives, triples with medians 2 and 4 are single-dipped, triples with median 3 are single-peaked. 2 and 4 are never-top alternatives (in some orders, these alternatives are bottom alternatives), 3 is a never-bottom alternative (in some orders it is top alternative). A more detailed analysis of small Fishburn's domains can be found in [18].

Fishburn's domain has arrangements of pseudolines representation [12], rhombus tiling diagrams representation [6], median graph representation [27], weak order of a finite Coxeter group representation [17]. The number of linear orders in Fishburn's domains is found by [12]. If the number of alternatives does not exceed seven, then Fishburn's domains contain the highest number of linear orders among all Condorcet domains [11, 12]. For a higher number of alternatives, Fishburn's domain is the basis for an algorithmic construction of large Condorcet domains [6, 16].

A domain D is called a *GF-domain* if there exists a linear ordering of alternatives $a_1, \dots, a_m$ and a subset $A \subseteq X$ such that for all i, j, k with $1 \le i < j < k \le m$ the restriction of the domain to the set $\{a_i, a_j, a_k\}$ is single-peaked with axis $a_i a_j a_k$ if $a_j \in A$, and it is single-dipped with axis $a_i a_j a_k$ if $a_j \in X \setminus A$.

The GF-domain is a natural generalization of the Fishburn's domain. In contrast with Fishburn's domains, GF-domains are closed under removing candidates operation. The following proposition generalizes [26] characterization of the single-peaked domain of preferences by weakening minimal richness to weak minimal richness.

Proposition 1

(a) Each GF-domain is a connected and weakly minimally rich Condorcet domain with maximal width. (In particular, GF-domain is semi-connected.)

(b) Conversely, let domain D be a semi-connected and weakly minimally rich Condorcet domain. Then, domain D is a GF-domain.

Proof

(a) Each GF-domain D is a peak-pit domain that contains a pair of reversed orders $a_1, \dots, a_m, a_m, \dots, a_1$. From [6], each maximal peak-pit domain that contains a pair of reversed orders is semi-connected. From [25], each maximal semi-connected domain is connected.

Alternatives a_1, a_m are simultaneously top and bottom alternatives. Let us consider a triple of alternatives a_1, x, a_m. If $x \in A$, then it is never last in the restriction of D to this triple. If $x \in X \setminus A$, then it is never first in the restriction of D to this triple. Thus, alternatives from A occupy top alternatives, alternatives from $X \setminus A$ occupy bottom alternatives. From [26], each alternative from A is a top alternative in restriction of D to A and each alternative from $X \setminus A$ is a bottom alternative in restriction of D to $X \setminus A$. Thus, each alternative form X is either top, or bottom alternative in D.

(b) Suppose that domain D is a semi-connected and weakly minimally rich Condorcet domain. Because of semi-connectedness, the restriction of domain D to each triple of alternatives is semi-connected. Thus, this restriction is either single-peaked, or single-dipped. Because of semi-connectedness, domain D contains a pair of mutually reversed orders $a_1 \dots a_m$, $a_m \dots a_1$.

Alternatives a_1, a_m are simultaneously top and bottom alternatives in domain D. There is no third such alternative, otherwise the restriction of D to this triple is neither single-peaked, nor single-dipped.

Let us consider axis $a_1 \dots a_m$. For each triple of alternatives, we have either top alternative median, or bottom alternative median. If the median alternative is a top alternative, then the restriction of domain D to this triple has three different top alternatives and this restriction is single-peaked where the median alternative is a never-last alternative in this restriction. If the median alternative is bottom alternative, then the restriction of domain D to this triple has three different bottom alternatives and this restriction is single-dipped where the median alternative is a never-first alternative in this restriction. Thus, domain D is a GF-domain.

□

A preference profile $\mathcal{P}$ is *single-peaked on a circle* (SPOC) with respect to a circular permutation of alternatives C if agent i's upper contour sets $U(P_i, x) = \{y \in X | y P_i x\}$ are intervals according to the circular permutation. Lower contour sets also form intervals according to the circular permutation. Preferences single-peaked on a circle and preferences single-dipped on a circle are equivalent. SPOC domain is symmetric, i.e. it contains a reverse of each preference order, which belongs to the domain (see [5, 15] for studies of symmetric Condorcet domains). The following proposition presents the forbidden configurations characterization for the SPOC domain.

Proposition 2 ([24]) *A preference profile is SPOC if and only if it avoids three configurations (*$\{x, y\}$ *means, that alternatives are situated in any order)*

(i) there are two agents $i, j \in N$ *and five alternatives* $x, y, z, t, r \in X$ *such that*

$$\{x, y\} P_i z P_i \{t, r\},$$

$$\{x, t\} P_j z P_j \{y, r\};$$

(ii) there are three agents $i, j, k \in N$ *and four alternatives* $x, y, z, t \in X$ *such that*

$$\{x, y\} P_i \{z, t\},$$

$$\{x, z\} P_j \{y, t\},$$

$$\{x, t\}P_k\{y, z\};$$

(ii) there are three agents $i, j, k \in N$ and four alternatives $x, y, z, t \in X$ such that

$$\{y, z\}P_i\{x, t\},$$

$$\{x, z\}P_j\{y, t\},$$

$$\{x, y\}P_k\{z, t\}.$$

Applying forbidden configurations characterization of the SPOC domain we will prove that GF-domain is SPOC, but not maximal SPOC.

Proposition 3 *Each GF-domain is a subset of a SPOC domain.*

Proof We will prove that each GF-domain avoids configurations (i), (ii), (iii) from Proposition 2.

If we have a linear ordering of alternatives order $a_1 ... a_m$, then preference orders $a_1 ... a_m$ and $a_m ... a_1$ belong to the corresponding maximal GF-domain.

If a maximal GF-domain D contains suborders xyz and zyx, alternatives x, z are not never-top, and are not never-bottom in the restriction of D to set $\{x, y, z\}$. Thus, alternatives x, z are not median in triple x, y, z, and the corresponding ordering of alternatives has subordering xyz or zyx.

If a domain D has configuration (i): there are two agents $i, j \in N$, and five alternatives $x, y, z, t, r \in X$ such that $\{x, y\}P_i z P_i\{t, r\}$, $\{x, t\}P_j z P_j\{y, r\}$, then we have $yP_i zP_i t$, $tP_j zP_j y$. Thus, the restriction of the ordering of alternatives to the set $\{y, z, t\}$ is either yzt or tzy.

If alternatives x, r are situated on the same side of z in the ordering, then, without loss of generality, we have subordering $yz\{t, x, r\}$ and triples y, z, x and y, z, r have median z. Restrictions of D to these triples are either simultaneously single-peaked with never-bottom z or simultaneously single-dipped with never-top z. $\{x, y\}P_i z$ reveals single-dipped triple, $zP_j\{y, r\}$ reveal single-peaked triple. We get a contradiction.

If alternatives x and r are not situated on the same side of z in the ordering, then, without loss of generality, we have ordering $\{y, x\}z\{t, r\}$ and triples t, z, x and y, z, r have median z. It contradicts with being a GF-domain because restriction to t, z, x is single-peaked, and restriction to y, z, r is single-dipped.

If a domain D has configuration (ii): there are three agents $i, j, k \in N$, and four alternatives $x, y, z, t \in X$ such that $\{x, y\}P_i\{z, t\}$, and $\{x, z\}P_j\{y, t\}$, and $\{x, t\}P_k\{y, z\}$, then restriction to set $\{y, z, t\}$ is single-peaked (restriction of configuration is $yP_i\{z, t\}$, and , $zP_j\{y, t\}$, and $tP_k\{y, z\}$). For each renaming of alternatives we can rename agents in order to get initial configuration. All alternatives are permutable. Without loss of generality, alternative t is never-bottom in this triple. We have $yP_i tP_i z$ and $zP_j tP_j y$. Thus, the restriction of the ordering to the set $\{y, z, t\}$ is either ytz or zty.

Within each four-alternatives restriction of a GF-domain we have either for four three-alternatives restrictions of the same type or two three-alternatives restriction of one type and two three-alternatives restriction of another type.

If all restrictions are single-peaked, then we cannot have three upper contour sets with x: $\{x, y\}$, $\{x, z\}$, $\{x, t\}$. Thus, we have two single-peaked restrictions with never-bottom t and two single-dipped restrictions.

In restrictions to sets $\{x, z, t\}$ and $\{x, y, t\}$ alternative t is a bottom alternative in one suborder. Thus, restrictions to sets $\{x, z, t\}$ and $\{x, y, t\}$ are single-dipped, and restriction to set $\{x, y, z\}$ is single-peaked. The median of $\{x, y, z\}$ does not equal to t. We get a contradiction

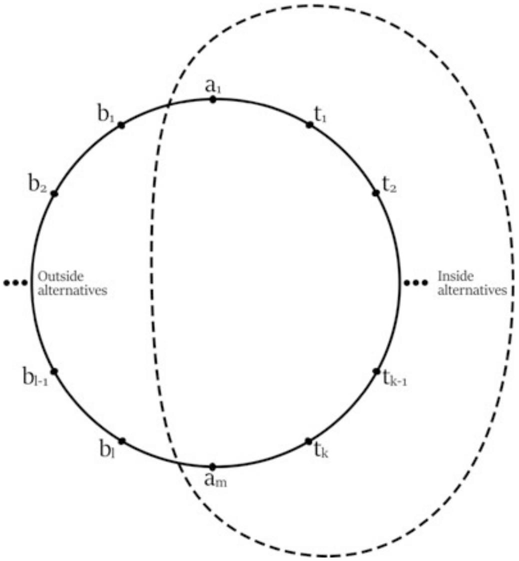
The argument for configuration (iii) is similar. □

3 Discussion

GF-domains receive clear SPOC interpretation and succeeds all algorithmic applications of SPOC domains from Peters and Lackner [24].

For GF-domain ordering $a_1 \ldots a_m$, there is a circular permutation of alternatives for the corresponding SPOC domain. Let $a_1 t_1 \ldots t_k a_m$ be the order of top alternatives and $a_1 b_1 \ldots b_l a_m$ be the order the bottom alternatives. These orders of alternatives are restrictions of the GF-domain ordering on the top and bottom alternatives correspondingly. The circular permutation $a_1 t_1 \ldots t_k a_m b_l \ldots b - 1$ is presented in Fig. 1. We propose the following interpretations of the picture.

Fig. 1 An example of circular permutation of alternatives with partition on inside and outside alternatives



There is a lake and a beach that occupies an interval of the lake coast. All agents take a rest on the beach. Alternatives are locations of ice-cream stands. Some of them are inside the beach (inside alternatives), others - outside. Some outside alternatives are better than some inside alternatives for some agents, but there is no agent which has an outside alternative as the first choice.

Another interpretation is time or calendar circle. 24-hours circle has a common working hours interval. A video conference session should proceed every day at exactly the same time slot. All agents prefer to do it within their working hours, but they can also compare outside alternatives. 365-days circle has a common school holidays interval. Students are going to have a trip. All agents prefer to do it within their holidays (inside alternatives), but can discuss other options.

Under political science interpretation [22], centrist policy candidates are inside alternatives. Extremist candidates from both sides complete the circle and constitute outside alternatives set.

[29] used the term "outside goods", which is not equivalent to our outside alternatives. A circular product differentiation model with an additional partition of products, e.g. domestic (inside), foreign (outside) leads to GF-domain interpretation. Consumers prefer domestic products, but can compare foreign products.

4 Conclusion

Empirical studies [19] show that single-peaked preferences are a theoretical concept without examples from real-world elections. Nearly single-peaked preferences [4] are aimed to approximate pure single-peaked preferences and enlarge the set of considered preference profiles. There are many ways to define nearly single-peaked preferences: preference profiles that become single-peaked after deleting k agents, or after deleting k alternatives, or after partitioning on k parts, or after executing k swaps of consecutive alternatives in preference orders, etc. All these generalizations are heuristic and do not have clear theoretical properties, e.g. all such domains are not Condorcet domains.

Despite nice mathematical properties of the Fishburn's domain, it has no applications in social choice theory. This paper axiomatically justifies a set of nearly single-peaked domains (GF-domains) and provides an interpretation for these preferences. Introducing GF-domains is a step towards applications of the new class of structured preferences, including the Fishburn's domain.

Forbidden configurations characterization of the GF-domain is an open problem. This characterization will lead to a deeper understanding of GF-domain structure and new applications in computational social choice theory.

Acknowledgments The author would like to thank Fuad Aleskerov for useful comments and Olga Skok for preparing figure 1. This article is an output of a research project implemented as part of the Basic Research Program at the National Research University Higher School of Economics (HSE University).

References

1. Aleskerov, F.: Arrovian Aggregation Models. Kluwer Academic Publishers, Dordrecht (1999)
2. Arrow, K.J.: Social Choice and Individual Values. Wiley, New York (1951)
3. Bogomolnaia, A., Moulin, H., Sandomirskiy, F.: Competitive division of a mixed manna. Econometrica **85**(6), 1847–1871 (2017)
4. Bredereck, R., Chen, J., Woeginger, G.J.: A characterization of the single-crossing domain. Soc. Choice Welf. **41**(4), 989–998 (2013)
5. Danilov, V.I., Koshevoy, G.A.: Maximal Condorcet domains. Order **30**(1), 181–194 (2013)
6. Danilov, V.I., Karzanov, A.V., Koshevoy, G.A.: Condorcet domains of tiling type. Discrete Appl. Math. **160**(7–8), 933–940 (2012)
7. Demange, G.: Single-peaked orders on a tree. Math. Soc. Sci. **3**(4), 389–396 (1982)
8. Elkind, E., Lackner, M.: Structure in dichotomous preferences. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence, pp. 2019–2025 (2015)
9. Elkind, E., Lackner, M., Peters, D.: Preference restrictions in computational social choice: a survey. arXiv:2205.09092v1 [cs.GT] (2022)
10. Fishburn, P.C.: Acyclic sets of linear orders. Soc. Choice Welf. **14**(1), 113–124 (1996)
11. Fishburn, P.C.: Acyclic sets of linear orders: a progress report. Soc. Choice Welf. **19**(2), 431–447 (2002)
12. Galambos, A., Reiner, V.: Acyclic sets of linear orders via the Bruhat orders. Soc. Choice Welf. **30**(2), 245–264 (2008)
13. Inada, K.: A note on the simple majority decision rule. Econometrica **32**, 525–531 (1964)
14. Karpov, A.: Structured preferences: a survey. Automat. Remote Control **83**(9), 1329–1354 (2022)
15. Karpov, A., Slinko, A.: Symmetric maximal Condorcet domains. Order (2022). https://doi.org/10.1007/s11083-022-09612-8
16. Karpov, A., Slinko, A.: Constructing large peak-pit Condorcet domains. Theory Decis. **94**(1), 97–120 (2023)
17. Labbé, J.-P., Lange, C.E.M.: Cambrian acyclic domains: counting c-singletons. Order **37**, 571–603 (2020)
18. Li, G., Puppe, C., Slinko, A.: Towards a classification of maximal peak-pit Condorcet domains. Math. Soc. Sci. **113**, 191–202 (2021)
19. Mattei, N., Walsh, T.: Preflib: a library of preference data. In: Proceedings of Third International Conference on Algorithmic Decision Theory (ADT 2013). Lecture Notes in Artificial Intelligence, pp. 259–270 (2013)
20. Mirkin, B.G.: Group Choice. V. H. Winston, Washington (1979)
21. Nehring, K., Puppe, C.: The structure of strategy-proof social choice — Part I: general characterization and possibility results on median spaces. J. Econ. Theory **135**(1), 269–305 (2007)
22. Peeters, R., Saran, R., Yüksel, A.M.: Strategic party formation on a circle and Durverger's Law. Soc. Choice Welfare **47**, 729–759 (2016)
23. Peters, D.: Recognising multidimensional Euclidean preferences. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence, pp. 642–648 (2017)
24. Peters, D., Lackner, M.: Preferences single-peaked on a circle. J. Artif. Intell. Res. **68**, 463–502 (2020)
25. Puppe, C.: The single-peaked domain revisited: a simple global characterization. Working Paper 97, KIT (2016)
26. Puppe, C.: The single-peaked domain revisited: a simple global characterization. J. Econ. Theory **176**, 55–80 (2018)
27. Puppe, C., Slinko, A.: Condorcet domains, median graphs and the single-crossing property. Econ. Theory **67**(1), 285–318 (2019)

28. Puppe, C., Slinko, A.: Maximal Condorcet domains. A further progress report (2022). SSRN: https://ssrn.com/abstract=4326244
29. Salop, S.C.: Monopolistic competition with outside goods. Bell J. Econ. **10**, 141–156 (1979)
30. Terzopoulou, Z., Karpov, A., Obraztsova, S.: Restricted domains of dichotomous preferences with possibly incomplete information. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence, pp. 5726–5733 (2021)
31. Tirole, J.: The Theory of Industrial Organization. MIT Press, Cambridge (1988)

About Some Clustering Algorithms in Evidence Theory

Alexander Lepskiy

1 Introduction

The Dempster–Shafer evidence theory [2, 15] considers data that is represented by a pair of objects $F = (\mathcal{A}, m)$, where $\mathcal{A}$ is the set of non-empty subsets (focal elements) of some base set X, m is a non-negative numerical function of sets (mass function) defined on the set of all subsets of the base set. The focal element $A \in \mathcal{A}$ describes the membership set of the true alternative $x \in A$ (for example, the air temperature forecast), and the mass $m(A)$ of this focal element A specifies the degree of belief that $x \in A$. Some set functions are put into one-to-one correspondence with the body of evidence. For example, there are such functions as belief and plausibility functions, which can be considered as the lower and upper bounds of the probability measure.

The tools for aggregating information presented by bodies of evidence, considering the reliability of information sources, their inconsistency and inaccuracy, are widely developed in the theory of evidence. However, many of the evidence body processing operations are computationally complex. In addition, it is required to reveal the enlarged structure of the set of focal elements in several problems, to analyze the degree of homogeneity of the body of evidence, its internal inconsistency, etc. Therefore, there is a need to approximate complex evidence bodies with many focal elements by simpler evidence bodies with a smaller number of focal elements. Both an approximation of a set function (for example, a belief function) corresponding to the body of evidence by another set function from a given class, and an approximation based on clustering of a set of focal elements are considered.

A. Lepskiy (✉)
National Research University Higher School of Economics, Moscow, Russia
e-mail: alepskiy@hse.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_12

Pignistic probability is an example of the first type of approximation [16]. Below we consider only an approximation based on the clustering of the set of focal elements.

Evidential data have their own frequency-set specifics. Therefore, direct analogues of the well-known clustering algorithms for 'point' data either need deep modernization or additional interpretability.

This article will analyze some modern methods for clustering bodies of evidence. The article is of an overview and methodological nature, but it will consider a new method, which is an analogue of the k-means method for evidence bodies.

2 Necessary Information from Evidence Theory

Let X be some finite (for simplicity) basic set, 2^X be the set of all subsets from X. Let us consider some subset of non-empty sets (focal elements) $\mathcal{A}$ from 2^X and a non-negative set function (mass function) $m : 2^X \to [0,1]$ that satisfies the conditions: $m(A) > 0 \Leftrightarrow A \in \mathcal{A}$, $\sum_{A\in\mathcal{A}} m(A) = 1$. A pair $F = (\mathcal{A}, m)$ is called a body of evidence. Let $\mathcal{F}(X)$ be the set of all evidence bodies on X.

There is a one-to-one correspondence between the body of evidence $F = (\mathcal{A}, m)$ and the belief function $Bel(A) = \sum_{B\subseteq A} m(B)$ or the plausibility function $Pl(A) = \sum_{B\cap A\neq\emptyset} m(B)$, which can be considered as lower and upper bounds for the probability $P(A)$, respectively. The following special cases of evidence bodies are distinguished:

(1) a categorical body of evidence of the form $F_A = (\{A\}, 1)$, i.e., a non-empty set A is the only focal element with unit mass;
(2) a vacuous body of evidence $F_X = (\{X\}, 1)$.

An arbitrary body of evidence $F = (\mathcal{A}, m)$ can be represented as $F = \sum_{A\in\mathcal{A}} m(A) F_A$.

The body of evidence of the type $F_A^\alpha = \alpha F_A + (1-\alpha) F_X$ is called simple.

The body of evidence $F = (\mathcal{A}, m)$ on X can be represented as a weighted hypergraph with a set of vertices X, a set of hyperedges $\mathcal{A}$ and their weights $m(A)$, $A \in \mathcal{A}$.

Example 1 Let we have $X = \{a, b, c, d, e\}$ and the body of evidence $F = 0.35F_{\{a\}} + 0.15F_{\{a,b\}} + 0.2F_{\{a,c\}} + 0.25F_{\{d,e\}} + 0.05F_{\{c,d,e\}}$ is given on X, i.e. $\mathcal{A} = \{\{a\}, \{a, b\}, \{a, c\}, \{d, e\}, \{c, d, e\}\}$. The hypergraph of the evidence body F is shown in Fig. 1. □

If two sources of information are represented by the bodies of evidence $F_1 = (\mathcal{A}_1, m_1)$ and $F_2 = (\mathcal{A}_2, m_2)$ on X, then the degree of conflict (contradiction) between these sources can be assessed using some functional (measure of external conflict) [10] $Con : \mathcal{F}(X) \times \mathcal{F}(X) \to [0, 1]$, which takes on greater values the more pairs of non-overlapping (or 'weakly over-lapping') focal elements of two evidence bodies with large masses exist. The classical measure of external conflict is [2]

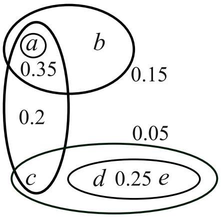


Fig. 1 Evidence body hypergraph

$$Con(F_1, F_2) = \sum_{A \cap B = \emptyset} m_1(A) m_2(B),$$

which we will use below. In addition to the measure of external conflict, the measure of internal conflict $Con_{in} : \mathcal{F}(X) \to [0, 1]$ of one body of evidence is also considered [11]. The ability to evaluate internal conflict is one possible application of evidence body clustering (see Remark 2 below).

3 Basic Approaches for Clustering Body of Evidence

The clustering of the body of evidence $F = (\mathcal{A}, m)$ is primarily related to the clustering of the set of its focal elements $\mathcal{A}$. There are two formulations of the problem of clustering a set of focal elements.

1. It is required to find such a subset of $\mathcal{A}' \subseteq 2^X$ that would be 'close' to $\mathcal{A}$ in some sense, but $\left|\mathcal{A}'\right| \ll |\mathcal{A}|$. The new mass function $m'(A)$, is found either by a local redistribution of the masses $m(B)$ of the sets B involved in the formation of a new focal element $A \in \mathcal{A}'$, or by a global redistribution that minimizes the discrepancy functional between $F = (\mathcal{A}, m)$ and $F' = \left(\mathcal{A}', m'\right)$.
2. It is required to find such a partition (or cover) of the set $\mathcal{A}$ of focal elements into subsets (clusters) $\{\mathcal{A}_1, \ldots, \mathcal{A}_l\}$ that would correspond in some sense to the structure of the set $\mathcal{A}$.

The first type of clustering is used to reduce the computational complexity of algorithms for processing evidence bodies or solving other approximation problems. The second type of clustering is used to identify the structure of a set of focal elements, to estimate the degree of heterogeneity, inconsistency, etc.

Next, we consider some implementations of clustering of these two types, namely:

(1) hierarchical clustering;
(2) clustering based on the density function of the distribution of conflict focal elements;
(3) clustering based on conflict optimization (including an analogue of the k-means method for evidence bodies).

3.1 Hierarchical Inner and Outer Clustering

The simplest approximation procedure by clustering was proposed in [12], where ‘close’ focal elements or focal elements with small masses were combined. In this case, the masses of the combined focal elements were summed up. A more complex clustering scheme, which is analogous to divisional-agglomerative algorithms [13] in some sense, has been proposed in [7] and [3, 14]. Two clusterings are the result of this algorithm. One of them is internal in the form of evidence body $F^- = \left(\mathcal{A}^-, m^-\right)$, the other is external in the form $F^+ = \left(\mathcal{A}^+, m^+\right)$. The set of focal elements $\mathcal{A}^-$ of internal clustering is the intersection of some sets from $\mathcal{A}$. While the set of focal elements $\mathcal{A}^+$ of external clustering is the union of some sets from $\mathcal{A}$. The masses of focal elements that are united in $\mathcal{A}^+$ or intersect in $\mathcal{A}^-$ are summarized: $m^-(B) = \sum_A m(A)$ if $B = \bigcap A \in \mathcal{A}^-$ and $m^+(C) = \sum_A m(A)$ if $C = \bigcup A \in \mathcal{A}^+$. In this case, such a pair (A, B) of focal elements is chosen for union/intersection, which delivers the minimum increment of the measure of imprecision [5] $f(F) = \sum_{A \in \mathcal{A}} m(A)\,|A|$.

The increments of this measure at the union/intersection of two sets and will be equal

$$\delta_{\cup}(C, D) = (m(C) + m(D))\,|C \cup D| - m(C)\,|C| - m(D)\,|D|$$

and

$$\delta_{\cap}(C, D) = m(C)\,|C| + m(D)\,|D| - (m(C) + m(D))\,|C \cap D|\,,$$

respectively. Therefore, the algorithm unites (intersects) those focal elements step by step, which deliver the minimum to the functional $\delta_{\cup}(C, D)$ ($\delta_{\cap}(C, D)$) at $C \neq D$. These procedures are repeated until a predetermined number $l < |\mathcal{A}|$ of focal elements remains, or some proximity condition between the original body of evidence $F = (\mathcal{A}, m)$ and its clustering is satisfied. As a result of such clustering, bodies of evidence F^- and F^+ are obtained, which in the theory of trust functions are called specialization and generalization of the body of evidence F, respectively [4]. Thus, the algorithm for hierarchical inner and outer clustering will be as follows.

Algorithm 1

Input data: body of evidence $F = (\mathcal{A}, m)$, the number of focal elements in clustering l.

Output data: bodies of evidence $F^- = \left(\mathcal{A}^-, m^-\right)$ and $F^+ = \left(\mathcal{A}^+, m^+\right)$.

1. Let $F^- = F^+ = F$.

2. Let’s find the pairs $(A^-, B^-) = \arg\min_{C \neq D} \delta_{\cap}(C, D)$ and $(A^+, B^+) = \arg\min_{C \neq D} \delta_{\cup}(C, D)$ in $\mathcal{A}^-$ and $\mathcal{A}^+$, respectively. Let’s replace a pair (A^-, B^-) with a set $A^- \cap B^-$ in $\mathcal{A}^-$, and a pair (A^+, B^+) with a set $A^+ \cup B^+$ in $\mathcal{A}^+$. We get new sets $\mathcal{A}^-$ and $\mathcal{A}^+$. Let’s recalculate the masses: $m^-(A^- \cap B^-) \leftarrow$

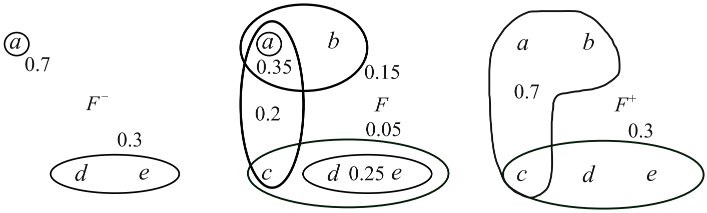


Fig. 2 Inner and outer clustering

$m^-(A^-) + m^-(B^-)$, $m^+(A^+ \cup B^+) \leftarrow m^+(A^+) + m^+(B^+)$, the masses of the remaining focal elements from $\mathcal{A}^-$ and $\mathcal{A}^+$ do not change.

3. Step 2 is repeated until $l < |\mathcal{A}^-| = |\mathcal{A}^+|$. □

Example 2 We have the following transformations of sets of focal elements for the outer and inner approximations of the body of evidence from Example 1 and $l = 2$, respectively (pairs of merged/intersected focal elements at each step are marked in bold):

$$\mathcal{A} = \{\{a\}, \{a, b\}, \{a, c\}, \{\boldsymbol{d}, \boldsymbol{e}\}, \{\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e}\}\} \to \{\{\boldsymbol{a}\}, \{\boldsymbol{a}, \boldsymbol{b}\}, \{a, c\}, \{c, d, e\}\} \to$$

$$\to \{\{\boldsymbol{a}, \boldsymbol{b}\}, \{\boldsymbol{a}, \boldsymbol{c}\}, \{c, d, e\}\} \to \{\{a, b, c\}, \{c, d, e\}\} = \mathcal{A}^+,$$

$$\mathcal{A} = \{\{a\}, \{a, b\}, \{a, c\}, \{\boldsymbol{d}, \boldsymbol{e}\}, \{\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{e}\}\} \to \{\{\boldsymbol{a}\}, \{a, b\}, \{\boldsymbol{a}, \boldsymbol{c}\}, \{d, e\}\} \to$$

$$\to \{\{\boldsymbol{a}\}, \{\boldsymbol{a}, \boldsymbol{b}\}, \{d, e\}\} \to \{\{a\}, \{d, e\}\} = \mathcal{A}^-.$$

We obtain the outer and inner approximations of the evidence body F, respectively $F^+ = 0.7F_{\{a,b,c\}} + 0.3F_{\{c,d,e\}}$ and $F^- = 0.7F_{\{a\}} + 0.3F_{\{d,e\}}$ (see Fig. 2). □

3.2 Clustering Based on Conflict Density Distribution

Another approach to clustering is to find a small (by cardinality) subset $\mathcal{A}' \subseteq \mathcal{A}$ of 'significant' focal elements. What characteristics of focal elements can be considered significant? These can be the mass of the focal element, its cardinality (a measure in the case of a measurable X), the number of other focal elements that intersect with the given one, etc. In [1], these characteristics were combined in the concept of the density of distribution of conflict focal elements. Non-overlapping focal elements are called conflicting.

A function $\psi_F : 2^X \to [0, 1]$ is called the conflict density distribution of the evidence body $F = (\mathcal{A}, m)$ if it satisfies the conditions:

1. $\psi_F(A) = 0$ if$B \cap A \neq \emptyset\ \forall B \in \mathcal{A}$;
2. $\psi_F(A) = 1$ if $B \cap A = \emptyset\ \forall B \in \mathcal{A}$;
3. $\psi_{\alpha F_1 + \beta F_2} = \alpha\psi_{F_1} + \beta\psi_{F_2}\ \forall F_1, F_2 \in \mathcal{F}(X)$, where $\alpha + \beta = 1, \alpha \geq 0, \beta \geq 0$.

It can be shown that the conflict density function will be equal to $\psi_F(A) = \sum_{B:A\cap B=\emptyset} m(B) = 1 - Pl(A)$. 'Significant' focal elements in [1] were those that maximize the function $\varphi_F(A) = m(A)\psi_F(A)$, $A \in \mathcal{A}$. The distance between the selected focal elements was another characteristic that was considered in [1] when choosing elements for $\mathcal{A}' \subseteq \mathcal{A}$. This distance should not be too small. Thus, the set $\mathcal{A}' \subseteq \mathcal{A}$ will consist of sets that provide a large value of the function φ_F and are located at a sufficiently large distance from each other. Let $d(A, B)$ be a metric on the set of focal elements. Then the algorithm for choosing the set $\mathcal{A}' \subseteq \mathcal{A}$ will be as follows.

Algorithm 2

Input data: body of evidence $F = (\mathcal{A}, m)$, the minimum possible value $h_1 > 0$ of $\varphi_F(A)$ for every $A \in \mathcal{A}'$; the minimum possible distance $h_2 > 0$ between focal elements from $\mathcal{A}'$.

Output data: the body of evidence $\mathcal{A}' \subseteq \mathcal{A}$.

1. Let the set of focal elements be ordered in descending order of the function φ_F: $\varphi_F(A_1) \geq \varphi_F(A_2) \geq \ldots \geq \varphi_F(A_k)$. Put $\mathcal{A}' = \{A_1\}$, $s := 2$.
2. If $\varphi_F(A_s) \leq h_1$, then the end. Otherwise, go to step 3.
3. If $\min_{A\in\mathcal{A}'} d(A, A_s) > h_2$, then $\mathcal{A}' := \mathcal{A}' \cup \{A_s\}$, $s := s + 1$, go to step 2. □

The function $d(A, B) = |A\Delta B|$ can be used as a metric between focal elements, where Δ is the symmetric difference of sets. To take into account not only the mutual position of focal elements, but also their masses, one can use metrics on the set of all evidence bodies $\mathcal{F}(X)$ [8]. For example, if a certain metric ρ is chosen on $\mathcal{F}(X)$, then the metric on 2^X can be defined as $d(A, B) = \rho(F_A^{m(A)}, F_B^{m(B)})$, where $F_A^{m(A)}$, $F_B^{m(B)}$ are simple evidence bodies. For example, $F_A^{m(A)} = m(A)F_A + (1 - m(A))F_X$. In particular, the following metric that is popular in evidence theory [9] between evidence bodies $F_1 = (\mathcal{A}_1, m_1)$ and $F_2 = (\mathcal{A}_2, m_2)$ can be used:

$$\rho_J(F_1, F_2) = \sqrt{\frac{1}{2}\sum_{A,B\in 2^X\setminus\{\emptyset\}} s_{A,B}(m_1(A) - m_2(A))(m_1(B) - m_2(B))},$$

where $s_{A,B} = |A \cap B| \big/ |A \cup B|$ is the Jaccard index. It is easy to see that $\rho_J(F_1, F_2) \in [0, 1]\ \forall F_1, F_2 \in \mathcal{F}(X)$. It can be shown that then the metric $d_J(A, B) = \rho_J(F_A^{m(A)}, F_B^{m(B)})$ takes the form.

Lemma 1 $d_J^2(A, B) = (m(A) - m(B))^2 + m(A)m(B)\frac{|A\Delta B|}{|A\cup B|} -$
$(m(B) - m(A))\ \frac{|B|m(B) - |A|m(A)}{|X|}$.

In particular, if $m(A) = m(B) = m$, *then* $d_J(A, B) = m\sqrt{|A\Delta B| \big/ |A \cup B|}$.

Note that Algorithm 2 can be considered as an evidential analogue of the popular 'point' the DBSCAN algorithm (DensityBased Spatial Clustering of Applications with Noise, [6]).

Example 3 Algorithm 2 will give the following result for the evidence body from Example 1 using the metric d_J, $h_1 = 0.1$, $h_2 = 0.2$.

Step 1. $\varphi_F(\{a\}) = 0.105$, $\varphi_F(\{a,b\}) = 0.045$, $\varphi_F(\{a,c\}) = 0.05$, $\varphi_F(\{d,e\}) = 0.175$, $\varphi_F(\{c,d,e\}) = 0.025$. Therefore, the set of focal elements will be ordered as follows: $\mathcal{A} = \{\{d,e\}, \{a\}, \{a,c\}, \{a,b\}, \{c,d,e\}\}$, $\mathcal{A}' = \{\{d,e\}\}$, $s := 2$.

Step 2. $\varphi_F(\{a\}) = 0.105 > h_1 \Rightarrow$ go to step 3.

Step 3. $d_J(\{d,e\}, \{a\}) \approx 0.317 > h_2 \Rightarrow \mathcal{A}' := \mathcal{A}' \cup \{\{a\}\} = \{\{d,e\}, \{a\}\}$, $s := 3$.

Step 2.1. $\varphi_F(\{a,c\}) = 0.05 < h_1 \Rightarrow$ the end.

As a result, we get a new set of focal elements $\mathcal{A}' = \{\{d,e\}, \{a\}\}$.

The general view of the body of evidence with the set of focal elements $\mathcal{A}'$ will be as follows $F'(x) = xF_{\{a\}} + (1-x)F_{\{d,e\}}$, $x \in [0,1]$. The masses of the focal elements of the body of evidence F' can be found from the condition of minimizing the distance between F and F'. For example, if we use the metric ρ_J, then the solution to the problem $\rho_J(F, F'(x)) \to \min$ will be as follows $x_0 = \frac{149}{240} \approx 0.62$. Then $F' = 0.62F_{\{a\}} + 0.38F_{\{d,e\}}$ and $\rho_J(F, F'(x_0)) = 0.196$. □

3.3 Clustering Based on Conflict Optimization

These methods are based on the assumption of the heterogeneity of those bodies of evidence that need clustering. This heterogeneity, in particular, may be a consequence of the aggregation in a given body of evidence $F = (\mathcal{A}, m)$ of information from different, sometimes contradictory, sources. In this case, it is required to find such a partition (or cover) of the set of focal elements $\mathcal{A}$ into subsets (clusters) $\{\mathcal{A}_1, \ldots, \mathcal{A}_l\}$ in order to optimize intracluster or intercluster conflict.

If a certain subset $\mathcal{A}' \subseteq \mathcal{A}$ of focal elements is selected, then we will further consider the following local redistribution of masses from $\mathcal{A}$ to $\mathcal{A}'$ (and such a body of evidence will be denoted by $F(\mathcal{A}') = (\mathcal{A}', m')$): $m'(A) = m(A)\ \forall A \in \mathcal{A}'$, $m'(X) = 1 - \sum_{A\in\mathcal{A}'} m(A)$. In particular, if , then $F(\{A\}) = F_A^{m(A)} = m(A)F_A + (1-m(A))F_X$ (simple evidence).

Then the following clustering optimization problem can be formulated. It is required to find such a partition (or cover) of the set of focal elements $\mathcal{A}$ into subsets (clusters) $\mathcal{C} = \{\mathcal{A}_1, \ldots, \mathcal{A}_l\}$ in order to maximize the external conflict between evidence clusters: $Con(F(\mathcal{A}_1), \ldots, F(\mathcal{A}_l)) \to \max$.

In the following algorithm, Algorithm 2 can be used to extract the set from l centers of new clusters $\mathcal{C} = \{\mathcal{A}_1, \ldots, \mathcal{A}_l\}$. The remaining focal elements from the set are redistributed among l clusters so that $Con(F(\mathcal{A}_1), \ldots, F(\mathcal{A}_l)) \to \max$.

Algorithm 3

Input data: body of evidence $F = (\mathcal{A}, m)$, a selected small set $\mathcal{A}' = \{A_1, \dots, A_l\}$ of l focal elements that will be the centers of new clusters.

Output data: partition (cover) $\mathcal{C} = \{\mathcal{A}_1, \dots, \mathcal{A}_l\}$ of the set of all focal elements $\mathcal{A}$.

1. Let $\mathcal{A}_i^{(0)} = \{A_i\}$, $i = 1, \dots, l$.
2. Focal elements from are redistributed among clusters $\mathcal{A}_1^{(0)}, \dots, \mathcal{A}_l^{(0)}$ according to the principle of conflict maximization between evidence clusters. The focal element $B \in \mathcal{A} \backslash \left\{ \mathcal{A}_1^{(0)}, \dots, \mathcal{A}_l^{(0)} \right\}$ will be assigned to that cluster $\mathcal{A}_i^{(0)}$ for which the maximum conflict measure is reached:

$$\mathcal{A}_i^{(0)} = \arg\max_{j : B \in \mathcal{A}_j^{(0)}} Con\left(F\left(\mathcal{A}_1^{(0)}\right), \dots, F\left(\mathcal{A}_j^{(0)} \cup \{B\}\right), \dots, F\left(\mathcal{A}_l^{(0)}\right) \right).$$

If equal maximum conflict values are obtained when assigning the element B to several clusters $\mathcal{A}_j^{(0)}$, $j \in J$, then this element B is included in all these clusters, and the mass value $m(B)$ is evenly distributed over the updated clusters, i.e. element B will be included in each cluster $\mathcal{A}_j^{(0)}$, $j \in J$ with weight $m(B) \big/ |J|$.

□

Example 4 Let's redistribute the remaining focal elements $\mathcal{A} \backslash \mathcal{A}' = \{\{a, b\}, \{a, c\}, \{c, d, e\}\}$ in accordance with Algorithm 4 for the body of evidence from Example 1 and the set of focal elements $\mathcal{A}' = \{\{d, e\}, \{a\}\}$ selected in Example 3.

Step 1. $\mathcal{A}_1^{(0)} = \{\{d, e\}\}$, $\mathcal{A}_2^{(0)} = \{\{a\}\}$.

Step 2. Let $B = \{a, b\}$. If $B \in \mathcal{A}_1$, then we obtain:

$F\left(\{B\} \cup \mathcal{A}_1^{(0)}\right) = 0.15F_{\{a,b\}} + 0.25F_{\{d,e\}} + 0.6F_X$,

$F\left(\mathcal{A}_2^{(0)}\right) = 0.35F_{\{a\}} + 0.65F_X$.

Then $Con\left(F\left(\{B\} \cup \mathcal{A}_1^{(0)}\right), F\left(\mathcal{A}_2^{(0)}\right)\right) = 0.25 \cdot 0.35 = 0.0875$.

If $B \in \mathcal{A}_2$, then we obtain:

$F\left(\mathcal{A}_1^{(0)}\right) = 0.25F_{\{d,e\}} + 0.75F_X$,

$F\left(\{B\} \cup \mathcal{A}_2^{(0)}\right) = 0.35F_{\{a\}} + 0.15F_{\{a,b\}} + 0.5F_X$

and $Con\left(F\left(\mathcal{A}_1^{(0)}\right), F\left(\{B\} \cup \mathcal{A}_2^{(0)}\right)\right) = 0.125$.

Thus, the element $B = \{a, b\}$ will be assigned to the cluster $\mathcal{A}_2$.

Let $B = \{a, c\}$. If $B \in \mathcal{A}_1$, then we obtain:

$F\left(\{B\} \cup \mathcal{A}_1^{(0)}\right) = 0.2F_{\{a,c\}} + 0.25F_{\{d,e\}} + 0.55F_X$,

$F\left(\mathcal{A}_2^{(0)}\right) = 0.35F_{\{a\}} + 0.65F_X$.

Then $Con\left(F\left(\{B\} \cup \mathcal{A}_1^{(0)}\right), F\left(\mathcal{A}_2^{(0)}\right)\right) = 0.25 \cdot 0.35 = 0.0875$.

If $B \in \mathcal{A}_2$, then we obtain:

$F\left(\mathcal{A}_1^{(0)}\right) = 0.25F_{\{d,e\}} + 0.75F_X,$
$F\left(\{B\} \cup \mathcal{A}_2^{(0)}\right) = 0.35F_{\{a\}} + 0.2F_{\{a,c\}} + 0.45F_X$
and $Con\left(F\left(\mathcal{A}_1^{(0)}\right), F\left(\{B\} \cup \mathcal{A}_2^{(0)}\right)\right) = 0.1375.$
Thus, the element $B = \{a, c\}$ will be assigned to the cluster $\mathcal{A}_2$.
Let $B = \{c, d, e\}$. If $B \in \mathcal{A}_1$, then we obtain:
$F\left(\{B\} \cup \mathcal{A}_1^{(0)}\right) = 0.25F_{\{d,e\}} + 0.05F_{\{c,d,e\}} + 0.7F_X,$
$F\left(\mathcal{A}_2^{(0)}\right) = 0.35F_{\{a\}} + 0.65F_X.$
Then $Con\left(F\left(\{B\} \cup \mathcal{A}_1^{(0)}\right), F\left(\mathcal{A}_2^{(0)}\right)\right) = 0.105.$
If $B \in \mathcal{A}_2$, then we obtain:
$F\left(\mathcal{A}_1^{(0)}\right) = 0.25F_{\{d,e\}} + 0.75F_X,$
$F\left(\{B\} \cup \mathcal{A}_2^{(0)}\right) = 0.35F_{\{a\}} + 0.05F_{\{c,d,e\}} + 0.6F_X$
and $Con\left(F\left(\mathcal{A}_1^{(0)}\right), F\left(\{B\} \cup \mathcal{A}_2^{(0)}\right)\right) = 0.0875.$
Thus, the element $B = \{a, c\}$ will be assigned to the cluster $\mathcal{A}_1$.
Thus, we get a partition $\mathcal{C} = \{\mathcal{A}_1, \mathcal{A}_2\}$, where $\mathcal{A}_1 = \{\{d, e\}, \{c, d, e\}\}$, $\mathcal{A}_2 = \{\{a\}, \{a, b\}, \{a, c\}\}$. □

Another variant of the optimization problem of evidence body clustering will be considered below. It is required to find such a partition (or cover) of the set of focal elements $\mathcal{A}$ into subsets (clusters) $\mathcal{C} = \{\mathcal{A}_1, \dots, \mathcal{A}_l\}$ in order to minimize the total internal conflict within evidence clusters: $\Phi = \sum_{i=1}^{l} Con_{in}(F(\mathcal{A}_i)) \to \min$, where Con_{in} is a measure of internal conflict. The total external conflict $Con_{in}(F(\mathcal{A}_i)) = \sum_{B\in\mathcal{A}_i} Con(F(\{B\}, C_i))$ between each body of evidence $F(\{B\})$, $B \in \mathcal{A}_i$ and some reference evidence (center) C_i of the i-th cluster can be considered as an internal conflict by analogy with the classical k-means algorithm

We will assume that center C_i has the form

$$C_i = \sum_{A\in\mathcal{A}_i} \alpha_i(A)F_A, \tag{1}$$

where $\boldsymbol{\alpha}_i = (\alpha_i(A))_{A\in\mathcal{A}_i} \in S_{|\mathcal{A}_i|}$, $S_k = \Big\{(t_1, \dots, t_k) : t_i \geq 0, i = 1, \dots, k,$ $\sum_{i=1}^{k} t_i = 1\Big\}$ is an k-dimensional simplex. The following theorem is true.

Theorem 1 *Let* $Pl_{\mathcal{A}_i}(A) = \sum_{\substack{B\in\mathcal{A}_i:\\ A\cap B\neq\emptyset}} m(B)$ *be the restriction of the plausibility function to the set* $\mathcal{A}_i$. *Then the minimum of the functional* Φ *for a fixed cover* $\mathcal{C} = \{\mathcal{A}_1, \dots, \mathcal{A}_l\}$ *will be achieved at*

$$\boldsymbol{\alpha}_i = (\alpha_i(A))_{A\in\overline{\mathcal{A}}_i} \in S_{\left|\overline{\mathcal{A}_i}\right|}, \quad i = 1, \dots, l, \tag{2}$$

where $\overline{\mathcal{A}_i} = \left\{ A \in \mathcal{A}_i : A = \underset{A \in \mathcal{A}_i}{\arg\max}\, Pl_{\mathcal{A}_i}(A) \right\}$.

Then the clustering algorithm (analogous to k-means) will be as follows.

Algorithm 4

Input data: body of evidence $F = (\mathcal{A}, m)$; number of clusters l; initial centers of clusters—bodies of evidence $C_i^{(0)}$, $i = 1, \ldots, l$; maximum conflict threshold within clusters $Con_{\max} \in [0, 1]$; $s = 0$.

Output data: partition (covering) $\mathcal{C} = \{\mathcal{A}_1, \ldots, \mathcal{A}_l\}$ of the set of all focal elements $\mathcal{A}$.

1. Focal elements are redistributed among clusters according to the principle of minimizing the conflict between evidence clusters and cluster centers. The focal element $B \in \mathcal{A}$ refers to the cluster $\mathcal{A}_i^{(s)}$ for which is achieved $\min_i Con\left(F(\{B\}), C_i^{(s)}\right) \le Con_{\max}$. If it is true that $\min_i Con\left(F(\{B\}), C_i^{(s)}\right) > Con_{\max}$, then the focal element B is assigned as the center of the new cluster. As a result, clusters $\mathcal{A}_i^{(s)}$, $i = 1, \ldots, l$ are obtained.
2. New cluster centers are calculated using formulas (1), (2), $s \leftarrow s + 1$.
3. Steps 1 and 2 are repeated until the clusters (or their centers) stabilize. □

Corollary 1 *Algorithm 4 converges in a finite number of steps.*

Example 5 Algorithm 4 will give the following result for the evidence body from Example 1. Let $l = 2$ be set, and the initial centers of the clusters coincide with the focal elements identified by Algorithm 2: $C_1^{(0)} = F(\{d, e\}) = 0.25F_{\{d,e\}} + 0.75F_X$, $C_2^{(0)} = F(\{a\}) = 0.35F_{\{a\}} + 0.65F_X$; $Con_{\max} = 1$; $s = 0$.

Step 1.1. We have

$Con\left(F(\{a\}), C_1^{(0)}\right) = 0.0875$, $Con\left(F(\{a, b\}), C_1^{(0)}\right) = 0.0375$,

$Con\left(F(\{a, c\}), C_1^{(0)}\right) = 0.05$,

$Con\left(F(\{d, e\}), C_1^{(0)}\right) = Con\left(F(\{c, d, e\}), C_1^{(0)}\right) = 0$,

$Con\left(F(\{a\}), C_2^{(0)}\right) = Con\left(F(\{a, b\}), C_2^{(0)}\right) = Con\left(F(\{a, c\}), C_2^{(0)}\right) = 0$,

$Con\left(F(\{d, e\}), C_2^{(0)}\right) = 0.0875$, $Con\left(F(\{c, d, e\}), C_2^{(0)}\right) = 0.0175$

Then the initial clustering will have the form according to the principle of minimizing the conflict between evidence clusters and cluster centers: $\mathcal{A}_1^{(0)} = \{\{d, e\}, \{c, d, e\}\}$, $\mathcal{A}_2^{(0)} = \{\{a\}, \{a, b\}, \{a, c\}\}$.

Step 1.2. New cluster centers are calculated using formulas (1), (2):

$Pl_{\mathcal{A}_1^{(0)}}(\{d, e\}) = Pl_{\mathcal{A}_1^{(0)}}(\{c, d, e\}) = 0.3$,

$Pl_{\mathcal{A}_2^{(0)}}(\{a\}) = Pl_{\mathcal{A}_2^{(0)}}(\{a, b\}) = Pl_{\mathcal{A}_2^{(0)}}(\{a, c\}) = 0.7$.

Therefore

$C_1^{(1)} = \alpha F_{\{d,e\}} + (1-\alpha) F_{\{c,d,e\}}$, $C_2^{(1)} = \beta F_{\{a\}} + \gamma F_{\{a,b\}} + (1-\beta-\gamma) F_{\{a,c\}}$, where $\alpha, \beta, \gamma \in [0,1]$, $\beta + \gamma \le 1$.

Step 2.1. Focal elements are redistributed:

$Con\left(F(\{a\}), C_1^{(1)}\right) = 0.35$, $Con\left(F(\{a,b\}), C_1^{(1)}\right) = 0.15$,

$Con\left(F(\{a,c\}), C_1^{(1)}\right) = 0.2\alpha$,

$Con\left(F(\{d,e\}), C_1^{(1)}\right) = Con\left(F(\{c,d,e\}), C_1^{(1)}\right) = 0$,

$Con\left(F(\{a\}), C_2^{(1)}\right) = Con\left(F(\{a,b\}), C_2^{(1)}\right) = Con\left(F(\{a,c\}), C_2^{(1)}\right) = 0$,

$Con\left(F(\{c,d,e\}), C_2^{(1)}\right) = 0.05(\beta+\gamma)$, $Con\left(F(\{d,e\}), C_2^{(1)}\right) = 0.25$.

Then $\mathcal{A}_1^{(0)} = \{\{d,e\},\{c,d,e\}\}$, $\mathcal{A}_2^{(0)} = \{\{a\},\{a,b\},\{a,c\}\}$. The clusters have stabilized. □

Remark 1 Since cluster centers may depend on parameters $\boldsymbol{\alpha} = (\alpha(A))_{A \in \overline{\mathcal{A}_i}} \in S_{\left|\overline{\mathcal{A}_i}\right|}$ (see formula (2)), additional procedures for choosing these parameters can be used in the algorithm, such as:

(1) cover minimization $\mathcal{C} = \{\mathcal{A}_1, \ldots, \mathcal{A}_l\}$. For example, $\sum_{i=1}^{l} |\mathcal{A}_i| \to \min$.
(2) minimizing the uncertainty of evidence bodies C_i, $i = 1, \ldots, l$. For example, (measure of imprecision [5]) $H(C_i) = \sum_{A \in \overline{\mathcal{A}_i}} \alpha_i(A)\,|A| \to \min$.
(3) minimizing the distance between the centers of clusters and the original body of evidence with respect to some metric ρ: $\rho(C_i, F) \to \min$, $i = 1, \ldots, l$; etc.

Remark 2 Clustering a body of evidence $F = (\mathcal{A}, m)$ can be used to evaluate its internal conflict. If $\mathcal{C} = \{\mathcal{A}_1, \ldots, \mathcal{A}_l\}$ is a cover (or partition) of the set of focal elements $\mathcal{A}$, then the internal conflict can be estimated by the formula $Con_{in}(F) = Con(F(\mathcal{A}_1), \ldots, F(\mathcal{A}_l))$. So, the measure of internal conflict of the body of evidence from Example 1 using the clustering of Example 5 (or Example 4) will be equal to $Con_{in}(F) = Con(F(\{d,e\},\{c,d,e\}_1), F(\{a\},\{a,b\},\{a,c\})) = 0.2$.

4 Conclusion

The article discusses the main known and currently being developed areas of evidence body clustering. In particular, the following classes of algorithms are considered: (a) hierarchical clustering algorithms; (b) clustering algorithms based on the density function; (c) clustering algorithms based on conflict optimization.

On the one hand, many of the considered algorithms are analogues of the corresponding algorithms for “point” data. On the other hand, the dual frequency-multiple nature of the bodies of evidence imposes peculiar restrictions, the need to use “one’s own” measures of proximity (for example, based on measures of conflict), etc. Some algorithms (for example, hierarchical ones) are explained by the peculiar goals of such clustering (for example, generating generalizations and specializations of the body of evidence).

All these features leave a lot of room for creativity in the development of algorithms for clustering bodies of evidence.

Acknowledgments The financial support from the Government of the Russian Federation within the framework of the implementation of the 5-100 Programme Roadmap of the National Research University Higher School of Economics is acknowledged.

References

1. Bronevich, A., Lepskiy, A.: Measures of conflict, basic axioms and their application to the clusterization of a body of evidence. Fuzzy Sets Syst. **446**, 277–300 (2022)
2. Dempster, A.P.: Upper and lower probabilities induced by multivalued mapping. Ann. Math. Statist. **38**, 325–339 (1967)
3. Denœux, T.: Inner and outer approximation of belief structures using a hierarchical clustering approach. Int. J. Uncertainty Fuzziness Knowledge-Based Syst. **9**(4), 437–460 (2001)
4. Dubois, D., Prade, H.: A set-theoretic view on belief functions: logical operations and approximations by fuzzy sets. Int. J. Gen. Syst. **12**, 193–226 (1986)
5. Dubois, D., Prade, H.: Consonant approximations of belief measures. Int. J. Approx. Reason. **4**, 419–449 (1990)
6. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proc. of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), pp. 226–231. AAAI Press, Washington (1996)
7. Harmanec, D.: Faithful approximations of belief functions. In: Laskey, K.B., Prade, H. (eds.) Uncertainty in Artificial Intelligence 15 (UAI99), Stockholm (1999)
8. Jousselme, A.-L., Maupin, P.: Distances in evidence theory: comprehensive survey and generalizations. Int. J. Approx. Reason. **53**, 118–145 (2012)
9. Jousselme, A.-L., Grenier, D., Bossé, É.: A new distance between two bodies of evidence. Inf. Fusion **2**, 91–101 (2001)
10. Lepskiy, A.: Analysis of information inconsistency in belief function theory. Part I: External conflict. Control Sci. **5**, 2–16 (2021)
11. Lepskiy, A.: Analysis of information inconsistency in belief function theory. Part II: Internal conflict. Control Sci. **6**, 2–12 (2021)
12. Lowrance, J.D, Garvey, T.D., Strat, T.M.: A framework for evidential reasoning systems. In: Kehler, T. et al. (eds.) Proceedings of AAAI'86, Philadelphia, August, vol. 2, pp.896–903 (1986)
13. Mirkin, B.: Core Data Analysis: Summarization, Correlation, and Visualization. Springer, Cham (2019)
14. Petit-Renaud, S., Denœux, T.: Handling different forms of uncertainty in regression analysis: a fuzzy belief structure approach. In: Hunter, A., Pearsons, S. (eds.) Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU'99), pp. 340–351. Springer, Berlin (1999)
15. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
16. Smets, P.: Decision making in TBM: the necessity of the pignistic transformation. Int. J. Approx. Reason. **38**, 133–147 (2005)

Inferring Multiple Consensus Trees and Supertrees Using Clustering: A Review

Vladimir Makarenkov, Gayane S. Barseghyan, and Nadia Tahiri

1 Introduction

The term phylogeny (i.e. phylogenetic tree or evolutionary tree) was introduced by Haeckel in 1866 [35], who defined it as “the history of the paleontological development of organisms by analogy with ontogeny or the history of individual development”. A phylogenetic tree represents a hypothesis about evolution of a given group of species which are usually associated with the tree leaves.

In mathematics, phylogenetic trees are called additive trees or X-trees (as their leaves are often associated with the set of species X; *[3]*). Let us now present some necessary mathematical definitions related to phylogenetic trees. The distance *δ(x,y)* between two vertices x and y in a phylogenetic tree T is defined as the sum of the edge lengths in the unique path linking x and y in T. Such a path is denoted *(x,y)*. A leaf is a vertex of degree one. Usually, a leaf represents a contemporary species (or a taxon).

Definition 1 *Let X be a finite set of n taxa. A dissimilarity d on X is a non-negative function on X × X such that for any x, y from X:*

$d(x,y) = d(y,x) \geq d(x,x) = 0.$

Definition 2 *A dissimilarity d on X satisfies the four-point condition if for any x, y, z, and w from X:* $d(x,y) + d(z,w) \leq \max\{d(x,z) + d(y,w); d(x,w) + d(y,z)\}$.

V. Makarenkov (✉) · G. S. Barseghyan
Département d’Informatique, Université du Québec à Montréal, Montreal, QC, Canada
e-mail: makarenkov.vladimir@uqam.ca

N. Tahiri
Département d’Informatique, Université de Sherbrooke, Sherbrooke, Québec, Canada

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_13

	x2	x3	x4	x5
x1	6	6	4	2
x2		2	4	6
x3			4	6
x4				4

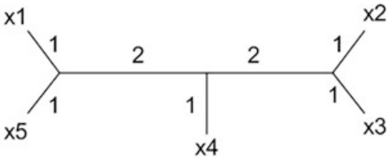


Fig. 1 An example of a tree metric on the set X of five taxa (on the left) and the corresponding phylogenetic tree (additive tree or X-tree) on the right

Definition 3 *For a finite set X, a phylogenetic tree (i.e. an additive tree or an X-tree, i.e. a tree whose leaves are labeled according to a final set of species X) is an ordered pair (T, φ) consisting of a tree T, with vertex set V, and a map φ: X → V with the property that, for all x ∈ X with degree at most two, x ∈ φ(X). A phylogenetic tree is binary if φ is a bijection from X into the leaf set of T and every interior vertex has degree three.*

The theorem relating the four-point condition and a dissimilarity representability by a phylogenetic tree is as follows:

Theorem 1 *(Zarestskii, Buneman, Patrinos & Hakimi, Dobson). Any dissimilarity satisfying the four-point condition on X × X (where X is a finite set of species) can be represented by a phylogenetic tree T such that for any x, y from X, d(x,y) is equal to the length of the path linking the leaves x and y in T. This dissimilarity is called a tree metric. Furthermore, this tree is unique.*

Figure 1 gives an example of a tree metric on the set *X* of five taxa and the corresponding phylogenetic tree.

Unfortunately, real-life evolutionary distances (or dissimilarities) rarely satisfy the four-point condition. Thus, one need to carry out an approximation algorithm to infer a tree metric matrix from a given matrix of evolutionary distances [32]. Among the most known distance-based approximation algorithms we can mention Neighbor-Joining [63], UPGMA [66], FITCH [31], and MW [45, 47].

Biologists often need to compare phylogenetic trees to each other in order to discover different evolutionary histories that govern a given set of species. There are several measures for comparing phylogenetic trees. The most popular of them include the Robinson and Foulds topological distance (*RF*) [61], the least-squares distance (*LS*), the bipartition dissimilarity (*BD*) [11], and the quartet distance (*QD*) [14]. In this literature review, we will mainly explore the methods based on the Robinson and Foulds distance. The Robinson and Foulds topological distance [61] between two trees is the minimum number of elementary operations (contraction and expansion) of nodes needed to transform one phylogenetic tree into another. It is also the number of splits (or bipartitions) that are present in one tree and absent in the other. The two phylogenetic trees in question must have the same set of taxa. The closer two phylogenetic trees are topologically, the smaller the value of the *RF* distance. It is often relevant to normalize the value of the *RF* distance by dividing it by its maximum possible value (equal to 2*n*-6) for two binary phylogenetic trees with n leaves. The RF distance calculation between two trees with n leaves can be carried out in $O(n)$ [22, 46, 48].

Often phylogenetic tree reconstruction methods do not return a single phylogenetic tree as output, but a collection of different trees [32]. Moreover, phylogenetic trees inferred for different genes often differ from each other. There is no absolute criterion for determining whether one tree is better than the others (except for the use of intrinsic criteria, e.g., the use of bootstrap scores). For this reason, it is preferable to seek a consensus representation of these trees, such that their concordant parts appear clearly in relation to the discordant parts. The resulting representation is called a *consensus tree*. Traditional consensus methods generate a single phylogenetic tree that is a representative of all of the input trees [15]. One of the first consensus methods was proposed by Adams (1972). Since then, a wide variety of methods have been developed. How to use them has been the subject of much debate [15, 27].

The main types of consensus trees are the following: the strict consensus tree [58, 67], the majority-rule consensus tree [53], the Nelson consensus tree [59], and the extended majority-rule consensus tree [30]. Let us briefly recall the main characteristics of each of these consensus trees.

The *strict consensus tree* (or Nelson's cladogram) is inferred by considering only those tree splits (i.e. bipartitions induced by the internal tree edges) that are identical in all trees compared. Conflicting parts of phylogenetic trees are represented by multifurcations in a strict consensus tree.

It is sometimes more convenient to have a less strict criterion than the one used by the strict consensus tree in order to allow bipartitions that are not necessarily present in all trees. When comparing a set of phylogenetic trees with different topologies, it is possible to search for the monophyletic groups that appear most frequently (often in more than 50% of the trees) among all the trees compared. The resulting tree is the *majority-rule consensus tree*.

The *extended majority-rule consensus tree* contains all majority bipartitions to which the remaining compatible bipartitions are added in turn, starting with the most frequent bipartitions for the given tree set. The process stops when a completely resolved (i.e. binary) tree is obtained. The extended majority consensus tree is the most frequently used in molecular biology, as it is always the best resolved among the three types of consensus trees discussed so far.

The *Nelson consensus tree* includes the heaviest set of compatible bipartitions. It consists in finding a clique of maximum weight in a compatibility graph of the entire bipartition set, which is NP-hard [15, 59].

Unfortunately, in many practical situations, phylogenetic trees used as input of consensus tree reconstruction methods can be quite divergent. This can happen, for example, when the input trees represent the evolution of different genes which have been affected by multiple reticulate evolutionary events such as horizontal gene transfer, hybridization or intragenic/intergenic recombination, ancient gene duplication or gene loss [2, 49, 57]. These evolutionary events can be unique for a subgroup of the input gene trees. Thus, it seems to be much more appropriate to represent this subgroup by its own consensus tree. However, the conventional consensus tree methods provide only one candidate tree for a given set of input gene phylogenies without considering their possible subgroups (or clusters) [44].

Figure 2 shows an example of four seven-leaf phylogenetic trees T_1, T_2, T_3, and T_4. Here, the solution consisting of two majority-rule consensus trees, T_{12} and T_{34}, seems to be much more appropriate than the conventional consensus solution consisting of a single majority-rule consensus tree, T_{1234}, i.e., here a star tree (a tree having no internal edges at all).

In many evolutionary studies gene trees to be combined are defined on different, but partially overlapping, sets of taxa (e.g. see Tree of Life project; [44]). It is very unlikely that all the genes considered have been sequenced for the same sets of species. In order to reconcile such trees, *supertree reconstruction* methods should be applied [7, 54, 75, 76]. Supertrees synthesize a given set of small (i.e. partial) trees with partial taxon overlap into comprehensive supertrees that include all taxa present in the given set of trees.

The most known supertree inference method is Matrix Representation with Parsimony (MRP) [6, 60] that carries out matrix-like aggregation of the given partial trees. The supertree reconstruction methods are commonly used for phylogenetic analysis of organisms with large genomes [8, 29, 38, 52]. For organisms with small genomes, such as prokaryotes, several approaches to genomic phylogenetic analysis have been adopted. In particular, supertree analysis provides new insights into the evolution of prokaryotes that could not be solved by many other approaches [21]. Recently, Makarenkov et al. [51] and Tahiri et al. [72] have used supertree phylogenetic analysis to characterize the evolution of SARS-CoV-2 genes.

As in the case of consensus trees, in many practical situations multiple conservative supertrees should be inferred to best represent the evolution of a given group of gene trees. Figure 3 shows an example of four phylogenetic trees T_1, T_2, T_3, and T_4 defined on different, but mutually overlapping, sets of seven taxa. Here, the solution consisting of two majority-rule supertrees, T_{12} and T_{34}, is more appropriate than that consisting of a single majority-rule supertree, T_{1234}, i.e., here a star tree, yielded by the traditional supertree approach.

The idea of building multiple consensus trees was originally formulated by Maddison [43]. He discovered that consensus trees for some subsets of input trees may differ a lot and that they are generally much better resolved than the single traditional consensus tree characterizing the whole set of the input trees. Many approaches have been developed to provide solutions for classifying phylogenetic trees based on the well-known clustering algorithms, such as k-means and k-medoids. We discuss their main features in the Methods section.

Partitioning is a clustering approach used to divide a given set of elements (or taxa) into a meaningful set of groups of elements (objects or entities) called clusters (or classes) [55, 56]. The objective of partitioning is to find groups of similar elements according to a given similarity measure. The four main partitioning approaches that can be used to group the elements based on the set of their features (or variables) are the following: (1) a center of gravity, i.e., the k-means algorithm [40, 42], where k denotes the number of clusters; (2) a geometric median, i.e., k-medians [13]; (3) a center containing the most frequent modes, i.e., k-modes [36];

5

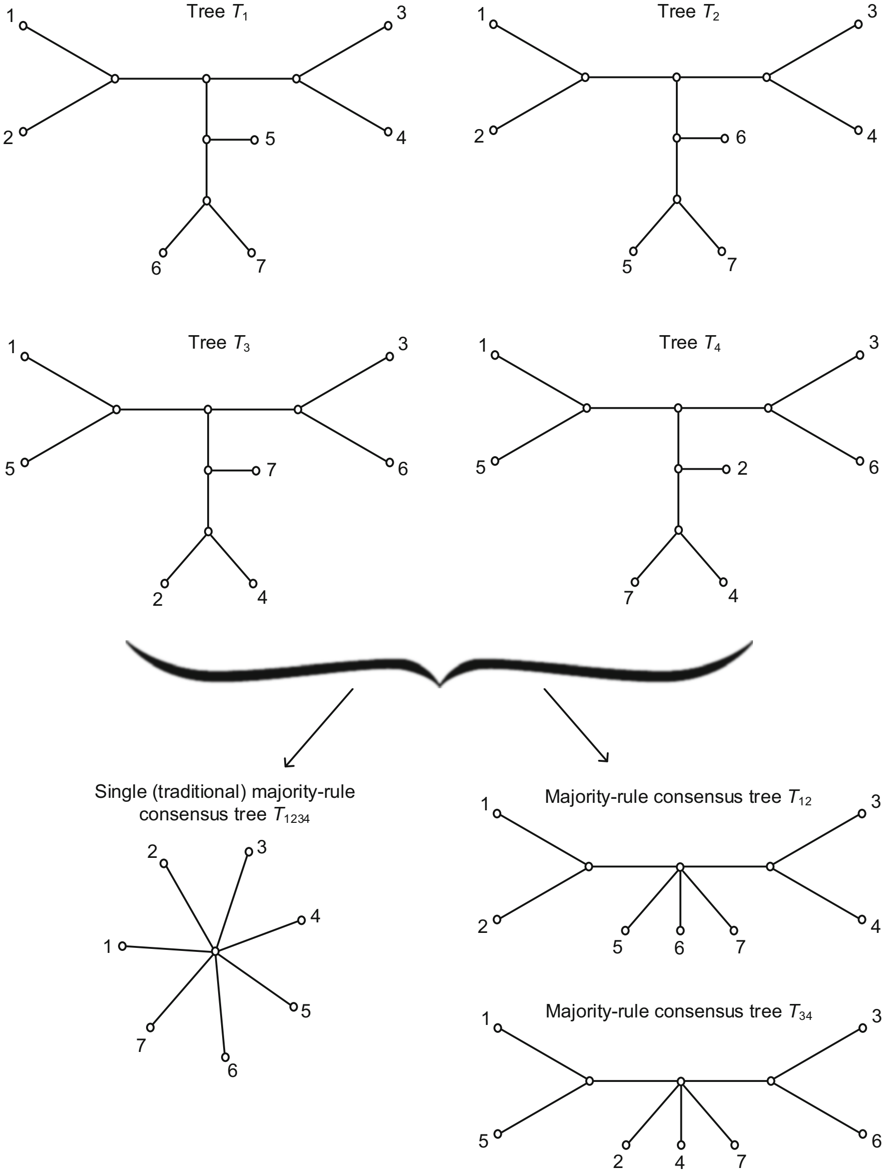

Tree T_1
1
2
3
4
5
6
7
Tree T_2
1
2
3
4
6
5
7
Tree T_3
1
5
3
6
7
2
4
Tree T_4
1
5
3
6
2
7
4
Single (traditional) majority-rule consensus tree T_{1234}
2
3
4
1
5
7
6
Majority-rule consensus tree T_{12}
1
2
3
4
5
6
7
Majority-rule consensus tree T_{34}
1
5
3
6
2
4
7

Fig. 2 Four phylogenetic trees T_1, T_2, T_3, and T_4 defined on the same set of seven leaves. Their single (traditional) majority-rule consensus tree is a star tree T_{1234}. The majority-rule consensus trees, T_{12} and T_{34}, constructed for the pairs of topologically close trees: T_1 and T_2, and T_3 and T_4, respectively

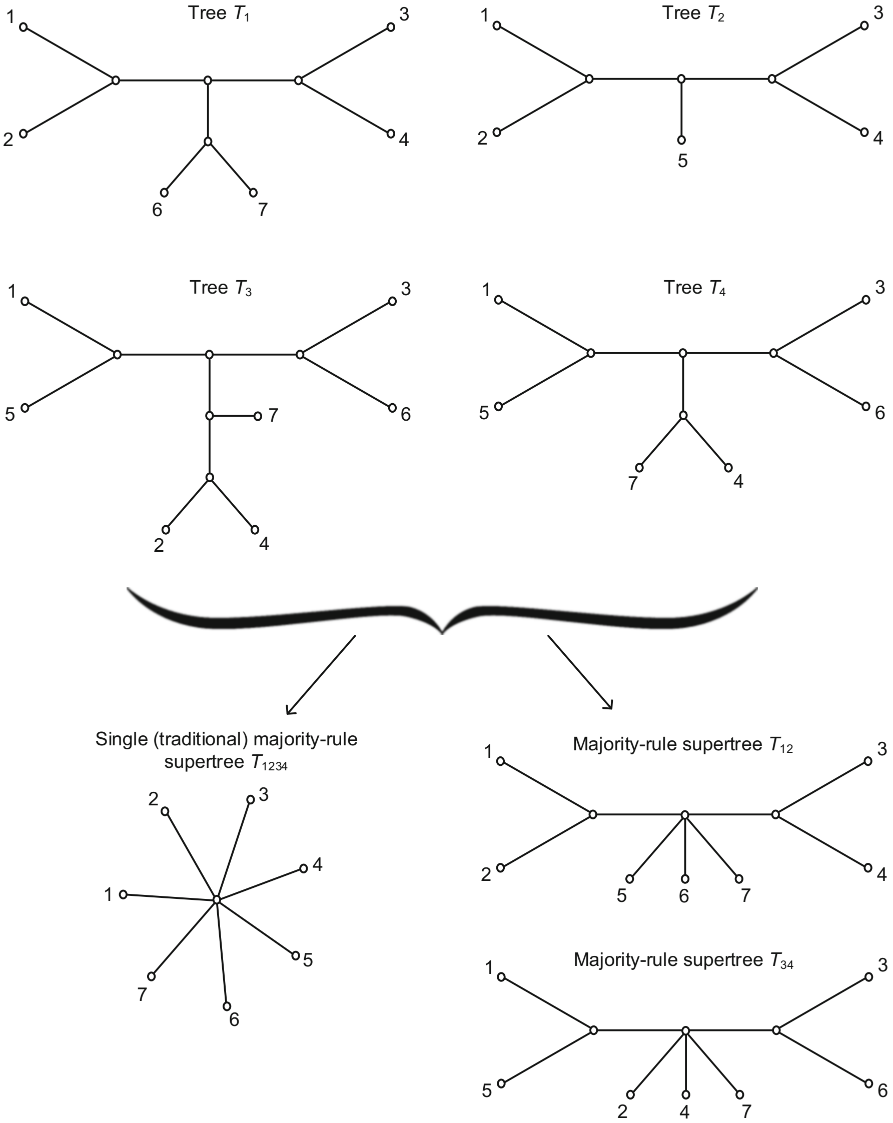

Tree T_1
1
2
3
4
6
7
Tree T_2
1
2
3
4
5
Tree T_3
1
5
3
6
7
2
4
Tree T_4
1
5
3
6
7
4
Single (traditional) majority-rule supertree T_{1234}
1
2
3
4
5
6
7
Majority-rule supertree T_{12}
1
2
3
4
5
6
7
Majority-rule supertree T_{34}
1
5
3
6
2
4
7

Fig. 3 Four phylogenetic trees T_1, T_2, T_3, and T_4 defined on different, but mutually overlapping, sets of seven taxa. Their single (traditional) majority-rule supertree is a star tree T_{1234}. The majority-rule supertrees, T_{12} and T_{34}, constructed for the pairs of topologically close trees: T_1 and T_2, and T_3 and T_4, respectively

(4) a medoid- based approach, in which a medoid is a cluster element that minimizes the sum of the distances between it and all other cluster elements, i.e., *k*-medoids [37]. In our literature review, we will mainly focus only on the *k*-means and *k*-

medoids algorithms as they have been extensively used in tree clustering (see the Methods section). Both of them are very fast, as the time complexity of k-means is $O(I \times K \times M \times N)$, where I is the number of iterations in the internal loop of k-means, K is the number of clusters, M is the number of features characterizing the given set of elements, and N is the number of elements, whereas the time complexity of k-medoids is $O(I \times K \times M \times (N - K)^2)$. It is worth noting that the k-medoids algorithm is much less sensitive to outliers than k-means. The Euclidean, Manhattan and Minkowski metrics are the most frequently used in the objective function of k-means and k-medoids [23, 24, 56]. However, in the case of tree clustering the Robison and Foulds topological distance or another tree distance should be used instead, and phylogenetic trees will play the role of cluster elements.

2 Methods

The Phylogenetic Islands [43] is a method that divides a collection of trees based on the branch length of the trees and the number of branch rearrangements by which the input trees differ. The author considers the three following types of branch rearrangement: NNI (nearest neighbor interchange), SPR (subtree pruning-regrafting), and TBR (tree bisection reconnection) [69, 70]. In NNI rearrangements, a clade (i.e. a subtree) can be moved to a nearby branch only, in SPR, it can be moved to a nearby or a distant branch, and in TBR, it can be moved to a nearby or a distant branch, with the clade also being rerooted. This method was developed to find the most-parsimonious trees using tree search algorithms, i.e., it starts with multiple starting points to find multiple islands. Maddison formally defines an *island of trees* of length L as a collection of n trees that satisfy three requirements: (1) all trees are of length $< L$; (2) each tree is connected to every other tree in the island through a series of trees, all of the length $< L$, with adjacent trees in the series differing only by a single rearrangement; and (3) all trees that satisfy criteria 1 and 2 are included in the set. Multiple islands can be discovered by performing many searches with a tree search tools available in PAUP* [69] and Henning86 [28], each search starting with a different tree. The trees are generally much more similar within islands than between islands, as shown by the analysis of partition metrics between trees (e.g. the Robinson and Foulds distance or the partition metric). The author concluded that trees on different islands may have different effects on trait evolution.

Characteristic trees that minimize the information loss [68] is an alternative approach to single consensus postprocessing methods in phylogenetic analysis. The presented approach was developed using popular clustering algorithms, namely k-means and agglomerative clustering [43]. The method proposed by Stockham et al. minimises the information loss using the characteristic tree concept. This method can be used to improve the resolution level of the output consensus trees and to provide more details about how the candidate trees are distributed. The major limitation of this method is that the input phylogenetic trees must have the same set of species (consensus case) and the method cannot address the case of homogeneous data (i.e. when the number of cluster $K = 1$). The objective function of the method considered by Stockham et al. [68] is as follow:

$$OF = \sum_{k=1}^{K} \sum_{i=1}^{N_k} RF^2 \left(T_k^{st}, T_{ki} \right), \tag{1}$$

where K is the number of clusters, N_k is the number of trees in cluster k, RF^2 is the squared Robinson and Foulds topological distance between the tree T_{ki} (i.e. tree i of cluster k) and the tree T_k^{st} that is the strict consensus tree of cluster k.

Multipolar Consensus (MPC) method [12] consists of finding a small set of trees including all splits with support greater than a predefined threshold. Given the splits to be displayed, the number of trees in the multipolar consensus must be minimised. This method can display more secondary evolutionary signals than majority-rule consensus. As the methods of Maddison [43] and Stockham et al. [68], the MPC method always generates as a solution multiple consensus trees and never a single one. Bonnard et al. [12] rely on a heuristic coloring scheme, called Greedy Coloring Algorithm, that uses two main steps: (1) to create an order on the vertices; and (2) to consider the vertices one by one in that order, assigning to a vertex the first color that is not assigned to an already colored vertex related to it. The MPC method differs from the other tree clustering methods in at least two ways: (1) it is more parsimonious, as each non-kernel split present in an input tree is represented only once; (2) it does not require prior clustering of the input trees. As a result, the time complexity of MPC is polynomial on the number of input splits, but only linear on the number of input trees.

The TreeOfTrees method [20] allows the comparison of X-tree topologies obtained from multiple sets of aligned gene sequences. The main goal of this method is to detect genes with identical histories using bootstrap sampling, and weighted or unweighted consensus. The comparison between tree sets is based on several tree metrics leading to a unique tree labelled by the gene trees (i.e. a kind of hierarchical tree clustering is presented). To estimate the robustness of the congruence between the input gene trees, a resampling procedure is used, which results in the construction of a "tree of gene trees" that provides both: a simple tree representation of the proximity of the gene trees, and a bootstrap value for each bipartition of the *tree of trees*. Each leaf of the tree of trees corresponds to a single gene (or a bootstrapped phylogenetic tree representing its evolution). The comparison between tree topologies starts by transforming each of the input trees into a pairwise distance matrix, counting the number of edges separating two taxa, or using a path length metric. The resulting tree distance matrix allows an unambiguous determination of the tree topology. The consensus tree T is constructed by enumerating all bipartitions belonging to the set of the input trees. Darlu and Guénoche propose the weighted consensus method, defined using the following weight function:

$$w\left(B_i\right) = \sum_{T_k \in S} \tau_k, \tag{2}$$

where B_i is the bipartition i, T_k is a tree of cluster k, S is the subset of the input trees containing the bipartition B_i, and τ_k is the measure of the quality of the tree T_k. Then, the authors define the weight of each of the input tree T_k as the sum of the weights of the internal edges contained in T_k using the following formula:

$$\Omega\left(T_k\right) = \sum_{B_i \in T_k} w\left(B_i\right). \tag{3}$$

Multiple Consensus Trees [34] is a tree clustering method intended to decide whether there is a single consensus among the input gene trees or not, and to detect divergent genes using a partitioning method. If the given gene trees are all congruent, they should be compatible with a single consensus tree. Otherwise, multiple consensus trees corresponding to divergent genetic patterns can be identified. The multiple consensus tree method optimises a generalised score, over a set of tree partitions to decide whether the given set of gene trees is homogeneous or not. The author considers unrooted X-trees only and focuses on the following consensus strategies: an X-tree is represented by a set of its bipartitions, each corresponding to an internal edge of the tree. Removing each internal edge results in a split, and hence a bipartition of the set of taxa X. The weight of each bipartition $B_i = X_i \cup X_i'$ is the number N_i of X-trees in the profile that contain that bipartition. The author defines the weight of an X-tree T, relative to the tree profile π of N trees, as follows:

$$W_\pi(T) = \sum_{B_i \in T_m} w\left(B_i\right) = \sum_{B_i \in T_m} N_i, \tag{4}$$

where $w(B_i)$ is the weight of each bipartition B_i of T, N_i is the number of internal majority edges (i.e. the edges satisfying the following condition $N_i > \frac{N}{2}$), and T_m is the tree T restricted to its majority edges. The weight of each bipartition B_i is the number N_i of X-trees in the profile containing this bipartition.

The author generalizes the score (4), defining it for a partition of trees P_π in k classes, as follows:

$$W^k\left(P_\pi\right) = \sum_{i=1,\ldots,k} p_i \times W_{\pi_i}\left(T_i^{\mathrm{maj}}\right), \tag{5}$$

where P_π is a partition of the set of trees π in k classes $(\pi_1, \ldots, \pi_k)$ containing respectively $\{p_1, \ldots, p_k\}$ trees, and T_i^{maj} is the majority consensus trees corresponding to class i.

Islands of Trees [65] is the method based on any appropriate pairwise tree-to-tree distance metric that extends the notion of island to any set or multiset of trees, such as those that can be generated by Bayesian or bootstrap methods and facilitates finding islands of trees *a posteriori*. This can be useful when the strict consensus of most parsimonious trees is relatively unresolved, although it relies on the analytical program (Silva and Wilkinson used PAUP*) to identify not only the number of islands, but also the constituents of most parsimonious trees. Distinct subsets of trees, such as tree islands, are complementary to other means of data exploration that involve attempts at partitioning sets of trees to obtain better summaries and promote better understanding of evolution. However, this method is of limited use for large phylogenetic tree distributions because it replaces the calculation of the distance with a very large number of pairwise comparisons of trees.

Inferring multiple consensus trees using *k*-medoids [71] is a fast method for inferring multiple consensus trees from a given set of phylogenetic trees defined on the same set of species. This method is based on the *k*-medoids partitioning algorithm to partition a given set of trees into multiple tree clusters. The well-known Silhouette and Caliński-Harabasz cluster validity indices have been adapted for tree clustering with *k*-medoids to determine the most appropriate number of clusters. It can be used to identify groups of gene trees that have similar evolutionary histories within the group and different evolutionary histories between the groups. This method is suitable for the analysis of large genomic and phylogenetic datasets.

Compared to the objective function used by Stockham et al. [68] (see Eq. 1), Tahiri et al. [71] used the majority-rule consensus tree instead of the strict consensus tree, and the unsquared *RF* distances instead of the squared one. The straightforward objective function to be minimized is then as follows:

$$OF = \sum_{k=1}^{K} \sum_{i=1}^{N_k} RF\left(T_k^{maj}, T_{ki}\right), \tag{6}$$

where *RF* is the Robinson and Foulds distance between the tree T_{ki} (i.e. tree *i* of cluster *k*) and T_k^{maj} that is the majority-rule consensus tree of cluster *k*. Nevertheless, computing the majority-rule consensus tree or the extended majority-rule consensus tree requires at least $O(nN)$ time, where *n* is the number of leaves (taxa or species) in each tree and *N* is the number of trees.

Thus, Tahiri et al. [71] used the following objective function in their method which is based on *k*-medoids:

$$OF_{med} = \sum_{k=1}^{K} \sum_{i=1}^{N_k} RF\left(T_k^{m}, T_{ki}\right), \tag{7}$$

where T_k^m is the medoid of cluster *k*, defined as a tree belonging to cluster *k* that minimizes the sum of the *RF* distances between it and all other trees in *k*. This version of the objective function is much faster than that based on Eq. (6) because it does not require the majority-rule consensus tree recomputation at each basic step of clustering algorithm. The running time of this method is $O(nN^2 + rK(N\text{-}K)^2I)$, where $O(nN^2)$ is the time needed to precalculate the matrix of pairwise *RF* distances of size ($N \times N$) between all input trees, K is the number of clusters, I is the number of iterations in the internal loop of k-medoids, and r is the number of different random starts used in k-medoids (usually hundreds of different random starts are needed to obtain good clustering results; [56]).

Inferring multiple consensus trees and supertrees using *k*-means [72] is a new method for inferring multiple alternative consensus trees and supertrees that best represent the main evolutionary patterns of a given set of gene trees. This method is based on the use of the popular k-means clustering algorithm and the Robinson and Foulds topological distance. It partitions a given set of trees into one, for homogeneous data, or multiple, for heterogeneous data, cluster(s) of trees. The authors show how the popular Caliński-Harabasz, Silhouette, Ball and Hall, and Gap cluster validity indices can be used in tree clustering with k-means. The Euclidean property of the square root of the Robinson and Foulds distance is used to define a fast and efficient objective function that is as follows:

$$OF_{EA} = \sum_{k=1}^{K} \frac{1}{N_k} \sum_{i=1}^{N_k-1} \sum_{j=i+1}^{N_k} RF\left(T_{ki}, T_{kj}\right), \tag{8}$$

The time complexity of the tree clustering algorithm based on Eq. (8) is $O(nN^2 + rNKI)$.

Moreover, the authors establish some interesting properties, and use them in the clustering process, of the general objective function defined in Eq. (6). Specifically, the lower and the upper bounds of this objective function OF are established in Theorem 2 below:

Theorem 2 [72]. *For a given cluster k containing N_k phylogenetic trees (i.e. additive trees or X-trees) the following inequalities hold:*

$$\begin{aligned} \frac{1}{N_k - 1} \sum_{i=1}^{N_k-1} \sum_{j=i+1}^{N_k} RF\left(T_{ki}, T_{kj}\right) &\le \sum_{i=1}^{N_k} RF\left(T_k^{\mathrm{maj}}, T_{ki}\right) \\ &\le \frac{2}{N_k} \sum_{i=1}^{N_k-1} \sum_{j=i+1}^{N_k} RF\left(T_{ki}, T_{kj}\right), \end{aligned} \tag{9}$$

where N_k is the number of trees in cluster k, T_{ki} and T_{kj} are, respectively, trees i and j in cluster k, and T_k^{maj} is the majority-rule consensus tree of cluster k.

In the same paper, Tahiri et al. show how their method can be extended to the case of supertree clustering. In the supertree clustering context, we assume that a given set of N unrooted phylogenetic trees may contain different, but mutually overlapping, sets of leaves. In this case, the original objective function OF shown in Eq. (6) can be reformulated as follows:

$$OF_{ST} = \sum_{k=1}^{K} \sum_{i=1}^{N_k} RF_{norm}\left(ST_k, T_{ki}\right) = \sum_{k=1}^{K} \sum_{i=1}^{N_k} \left(\frac{RF\left(ST_k, T_{ki}\right)}{2n\left(ST_k, T_{ki}\right) - 6} \right), \tag{10}$$

where K is the number of clusters, N_k is the number of trees in cluster k, $RF_{norm}(ST_k,T_{ki})$ is the normalized Robinson and Foulds topological distance between tree i of cluster k, denoted T_{ki}, and the majority-rule supertree of this cluster, denoted ST_k, reduced to a subtree having all leaves in common with T_{ki}. The RF distance is normalized here by dividing it by its maximum possible value (i.e. $2n(ST_k,T_{ki})$-6, where $n(ST_k,T_{ki})$ is the number of common leaves in ST_k and T_{ki}). The RF distance normalization is performed here to account equally the contribution of each tree to clustering. Clearly, Eq. (10) can be used only if the number of common leaves in ST_k and T_{ki} is larger than 3.

An analog of Eq. (8) can be used in supertree clustering to avoid supertree recalculations at each step of k-means. This can be done using the following objective function:

$$OF_{STEA} = \sum_{k=1}^{K} \frac{1}{N_k} \sum_{i=1}^{N_k-1} \sum_{j=i+1}^{N_k} \left(\frac{RF\left(T_{ki}, T_{kj}\right)}{2n\left(T_{ki}, T_{kj}\right) - 6} + \alpha \times \frac{n\left(T_{ki}\right) + n\left(T_{kj}\right) - 2n\left(T_{ki}, T_{kj}\right)}{n\left(T_{ki}\right) + n\left(T_{kj}\right)} \right), \tag{11}$$

where $n(T_{ki})$ is the number of leaves in tree T_{ki}, $n(T_{kj})$ is the number of leaves in tree T_{kj}, $n(T_{ki},T_{kj})$ is the number of common leaves in trees T_{ki} and T_{kj}, and α is the penalization (tuning) parameter, taking values between 0 and 1, needed to prevent from putting to the same cluster trees having small percentages of common leaves.

The simulations conducted by Tahiri et al. [72] illustrated that their new tree clustering method is faster and generally more efficient than the methods of Stockham et al. [68], Tahiri et al. [71] and Bonnard et al. [12] discussed earlier in this section.

3 Cluster Validity Indices Adapted to Tree Clustering

In this section, we show how the popular Caliński-Harabasz, Silhouette, Ball and Hall, and Gap cluster validity indices can be used in tree clustering with k-means.

3.1 Caliński-Harabasz Cluster Validity Index Adapted for Tree Clustering

The first cluster validity index we consider here is the Caliński-Harabasz index [17]. This index, sometimes called the variance ratio criterion, is defined as follows:

$$CH = \frac{SS_B}{SS_W} \times \frac{N-K}{K-1}, \tag{12}$$

where SS_B is the index of intergroup evaluation, SS_W is the index of intragroup evaluation, K is the number of clusters and N is the number of elements (i.e. trees in our case). The optimal number of clusters corresponds to the largest value of CH.

In the traditional version of CH, when the Euclidean distance is considered, the SS_B coefficient is evaluated by using the L$_2$-norm:

$$SS_B = \sum_{k=1}^{K} N_k \|m_k - m\|^2, \tag{13}$$

where m_k ($k = 1 \ldots K$) is the centroid of cluster k, m is the overall mean (i.e. centroid) of all elements in the given dataset X, and N_k is the number of elements in cluster k. In the context of the Euclidean distance, the SS_W index can be calculated using the two following equivalent expressions:

$$SS_W = \sum_{k=1}^{K}\sum_{i=1}^{N_k} \|x_{ki} - m_k\|^2 = \sum_{k=1}^{K} \frac{1}{N_k}\left(\sum_{i=1}^{N_k-1}\sum_{j=i+1}^{N_k} \left\|x_{ki} - x_{kj}\right\|^2\right), \tag{14}$$

where x_{ki} and x_{kj} are elements i and j of cluster k, respectively [17].

To use the analogues of Eqs. (13) and (14) in tree clustering, Tahiri et al. [72] used the concept of centroid for a given set of trees. The median tree [4, 5] plays the role of this centroid in a tree clustering algorithm. The median procedure [5] is defined below. The set of median trees, Md(Π), for a given set of trees $\Pi = \{T_1, \ldots, T_N\}$ having the same set of leaves S, is the set of all trees T defined on S, such that: $\sum_{i=1}^{N} RF\ (T, T_i)$ is minimized. If N is odd, then the majority-rule consensus tree, Maj(Π) of Π, is the only element of Md(Π). If N is even, then Md(Π) is composed of Maj(Π) and of some more resolved trees.

Tahiri et al. [72] proposed to use some formulas based on the properties of the Euclidean distance to define SS_B and SS_W in k-means-like tree clustering. These formulas do not require the computation of the majority (or the extended majority)-rule consensus trees at each iteration of k-means. Precisely, they replace the term $\|x_{ki} - x_{kj}\|^2$ in Eq. (14) by $RF(T_{ki}, T_{kj})$ to obtain the formula for SS_W:

$$SS_W = \sum_{k=1}^{K} \frac{1}{N_k} \sum_{i=1}^{N_k-1} \sum_{j=i+1}^{N_k} RF\left(T_{ki}, T_{kj}\right), \tag{15}$$

where T_{ki} and T_{kj} are trees i and j of cluster k, respectively.

Also, in the case of the Euclidean distance, the formula is as follows:

$$SS_B + SS_W = \frac{1}{N}\left(\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \left\|x_i - x_j\right\|^2\right), \tag{16}$$

where x_i and x_j are two different elements of X [17].

As a result, the approximation to the global variance between groups, SS_B, can be evaluated as follows:

$$SS_B = \frac{1}{N}\left(\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} RF\left(T_i, T_j\right)\right) - SS_W, \tag{17}$$

where T_i and T_j are trees i and j in the set of trees Π, and SS_W is calculated according to Eq. (15).

Based on the Euclidean properties of the square root of the Robinson and Foulds distance, Eqs. (15) and (17) establish the exact formulas for calculating the indices SS_B and SS_W for the objective function OF_{EA} defined by Eq. (8). Interestingly the objective function OF_{EA} can also be used as an approximation of the objective function defined in Eq. (6) (obviously, the centroid of a cluster of trees is not necessarily a consensus tree of the cluster; furthermore, it is not necessarily a phylogenetic tree).

3.2 Ball-Hall Index Adapted for Tree Clustering

Another relevant criterion to consider in this review is the Ball-Hall index. In 1965, Ball and Hall (*BH*) introduced the ISODATA procedure [1] to measure the average dispersion of groups of objects with respect to the mean square root distance, i.e. the intra-group distance. Unlike the *CH* index, the *BH* index can be used to find solutions consisting of a single consensus tree. Tahiri et al. [72] adapted the *BH* criterion for tree clustering with *k*-means, which led to the following formula:

$$BH = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{N_k} \sum_{i=1}^{N_k} RF\left(T_k^{\mathrm{maj}}, T_{ki}\right). \tag{18}$$

Furthermore, the following formula can be used to avoid the majority-rule tree calculation:

$$BH = \frac{1}{K} \sum_{k=1}^{K} \frac{1}{N_k^2} \sum_{i=1}^{N_k - 1} \sum_{j=i+1}^{N_k} RF\left(T_{ki}, T_{kj}\right). \tag{19}$$

3.3 Silhouette Index Adapted for Tree Clustering

The next popular criterion we consider here is the Silhouette (*SH*) width index [62]. Traditionally, the Silhouette width of cluster *k* is defined as follows:

$$s(k) = \frac{1}{N_k} \left[\sum_{i=1}^{N_k} \frac{b(i) - a(i)}{\max\left(a(i), b(i)\right)} \right], \tag{20}$$

where N_k is the number of elements belonging to cluster *k*, *a*(*i*) is the average distance between element *i* and all other elements belonging to cluster *k*, and *b*(*i*) is the smallest, over-all clusters k' different from *k*, of all average distances between *i* and all the elements of cluster *k'*.

Equations (21) and (22) can be used to calculate *a*(*i*) and *b*(*i*), respectively, in case of tree clustering:

$$a(i) = \frac{\sum_{j=1}^{N_k} RF\left(T_{ki}, T_{kj}\right)}{N_k}, \tag{21}$$

$$b(i) = \min_{1 \le k' \le K, k' \ne k} \frac{\sum_{j=1}^{N_{k'}} RF\left(T_{ki}, T_{k'j}\right)}{N_{k'}}, \tag{22}$$

where $T_{k'j}$ is tree *j* of cluster k', such that $k' \neq k$, and $N_{k'}$ is the number of trees in cluster k'.

The optimal number of clusters, *K*, corresponds to the maximum average value of *SH* that is calculated as follows:

$$SH = \overline{s}(K) = \sum_{k=1}^{K} \frac{[s(k)]}{K}. \tag{23}$$

The value of the *SH* index defined by Eq. (23) is located in the interval between -1 and $+1$.

3.4 Gap Statistic Adapted for Tree Clustering

The last criterion that we are discussing here is the *Gap* statistic [73]. As the *BH* index, *Gap* allows solutions consisting of a single consensus tree. The formulas proposed by Tibshirani et al. [73] are based on the properties of the Euclidean distance. In the context of tree clustering, Tahiri et al. [72] adapted the *Gap* statistic by defining the total intracluster distance, D_k, characterizing the cohesion between the trees belonging to the same cluster k, as follows:

$$D_k = \sum_{i=1}^{N_k} \sum_{j=1}^{N_k} RF\left(T_{ki}, T_{kj}\right). \tag{24}$$

The sum of the average total intracluster distances, V_K, can be calculated using the following formula:

$$V_K = \sum_{k=1}^{K} \frac{1}{2N_k} D_k. \tag{25}$$

The *Gap* statistic, which reflects the quality of a given clustering solution with K clusters, is traditionally defined as follows:

$$Gap_N(K) = E_N^* \left\{\log\left(V_K\right)\right\} - \log\left(V_K\right), \tag{26}$$

where E_N^* denotes expectation under a sample of size N from the reference distribution. The following formula [73] for the expectation of $log(V_K)$ was used in our method:

$$E_N^* \left\{\log\left(V_K\right)\right\} = \log\left(\frac{Nn}{12}\right) - \left(\frac{2}{n}\right)\log(K), \tag{27}$$

where n is the number of tree leaves. The largest value of the *Gap* statistic corresponds to the best clustering.

4 Example of Application to Evolutionary Data

Aminoacyl-tRNA synthetases (aaRSs) are enzymes that attach the appropriate amino acid to their cognate transfer RNA. The structure-function aspect of aaRSs has long interested biologists [33, 77]. It has been observed that the central role played by aaRSs in translation suggest that their evolutionary histories and that of the genetic code can be closely related [77]. This information would make aaRS gene domain analysis a key component of tree-of-life inference [16, 74]. Woese et al. examined the evolutionary profiles of each of the 20 standard aaRSs used by living cells to construct the evolutionary history of proteins organized into 5 groups (nonpolar aliphatic R group, nonpolar, aromatic R group, polar, uncharged R group, positively charged R group, and negatively charged R group). To conduct their famous aaRS analysis Woese et al. considered a total of 72 species from 3 main domains (Archaea, Eukarya and Bacteria), which can be represented by leaves of the related phylogenetic trees.

In our study, we used 36 aaRS phylogenetic trees (i.e. aaRS gene trees) originally constructed by Woese et al. These trees had different, but mutually overlapping, sets of leaves (in total 72 different species were considered). They are available on our GitHub repository along with our program at the following URL address: https://github.com/TahiriNadia/KMeansSuperTreeClustering. These 36 trees were used as input for our *KMeansSuperTreeClustering* algorithm [72]. Our supertree clustering algorithm was carried out with the following options: the Caliński-Harabasz [17] cluster validity index was used to select the best number of clusters (the number of clusters varied from 2 to 10 in our experiments) and the penalization parameter α was set to 1.

In these settings, our algorithm found that the best solution for these data corresponds to a 2-cluster partitioning. Each of these clusters of trees can be represented by its own supertree. The first obtained cluster includes 19 trees for a total of 61 different species, while the second obtained cluster includes 17 trees for a total of 56 species. The supertrees (see Figs. 3 and 4) for the two obtained tree clusters were inferred using the CLANN program [19]. In CLANN, we used the most similar supertree (dfit) method [18] with the mrp criterion. This criterion involves a matrix representation based on the parsimony criterion. Next, we inferred the most common (by cluster) horizontal gene transfers (HGT) that characterize the evolution of phylogenetic trees included in the two obtained clusters of trees. The HGT detection method by Boc et al. [11] was used for this purpose. It proceeds by reconciliation of the species and gene phylogenetic trees. In our case, the two obtained supertrees played the role of gene trees, while the species phylogenetic trees followed the NCBI taxonomic classification (see https://www.ncbi.nlm.nih.gov/Taxonomy/CommonTree/wwwcmt.cgi); they are presented by full edges in Figs. 4 and 5. These supertrees were not fully resolved (i.e. the first supertree, see Fig. 4 contains 9 internal nodes with degree greater than 3, whereas the second supertree, see Fig. 5 contains 10 internal nodes with degree greater than 3). We used the version of the HGT algorithm available on the T-Rex website [9] and Armadillo 1.1 [41] workflow platform to identify the scenarios of HGT events that reconcile each species tree with the corresponding supertree. The root of all of these trees was placed on the edge that splits the clade of Bacteria with those of Eukarya and Archaea. Two frequent horizontal gene transfers were found for the first supertree and four for the second supertree. Our results indicate that most of aminoacyl-tRNA synthetases underwent a two-way evolution. The obtained results are in line with the results of Dohm et al. [26] and Sharaf et al. [64] that aminoacyl-tRNA synthetases possess two versions of most tRS, one cytosolic and one mitochondrial.

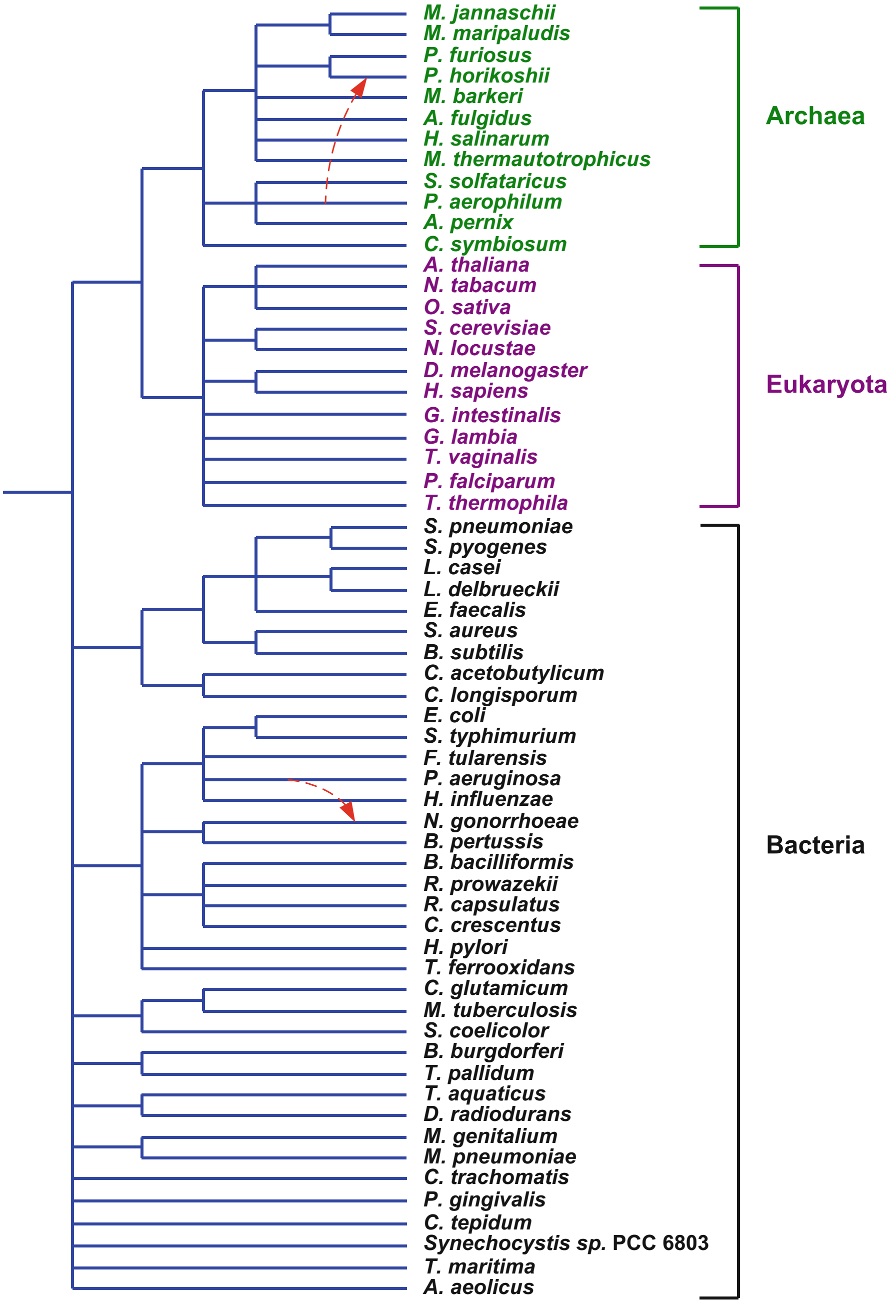


Fig. 4 Species tree (full edges) corresponding to the NCBI taxonomic classification constructed for 61 species from the first cluster of 19 aaRS phylogenetic trees. The two horizontal gene transfers (indicated by arrows) were found using the HGT-Detection program of Boc et al. [9]

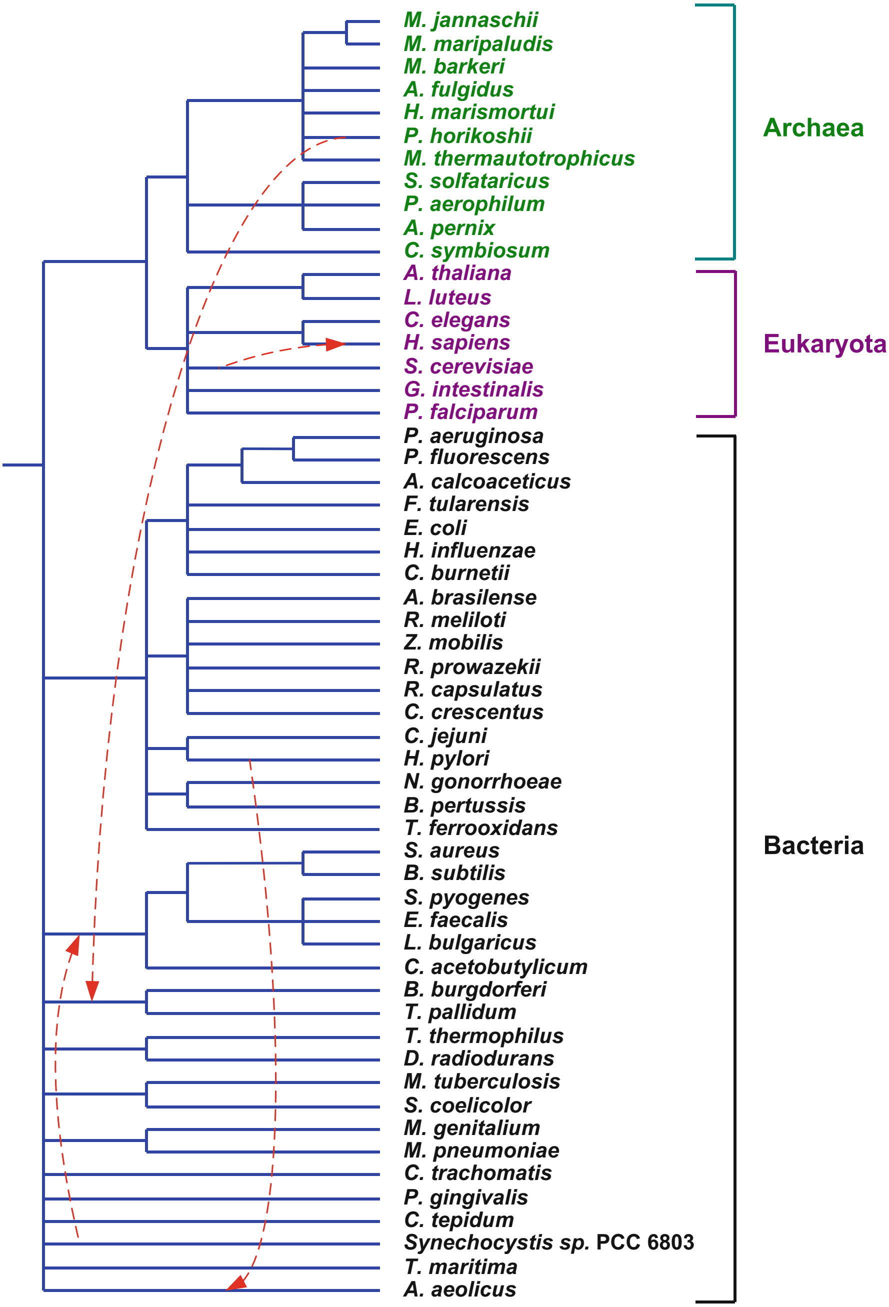


Fig. 5 Species tree (full edges) corresponding to the NCBI taxonomic classification constructed for 56 species from the first cluster of 17 aaRS phylogenetic trees. The four horizontal gene transfers (indicated by arrows) were found using the HGT-Detection program of Boc et al. [9]

5 Conclusion

In this paper, we have reviewed the state-of-the-art systematic methods for inferring multiple alternative consensus trees and supertrees from a given set of phylogenetic trees (i.e. additive trees, evolutionary trees or X-trees). Most of the reviewed papers describe algorithms proceeding by k-means or k-medoids clustering of tree topologies. In the case of consensus tree clustering problem, all the trees should be defined on the same set of taxa (i.e. species associated to the tree leaves), whereas in the case of supertree clustering problem, the trees can be defined on different, but mutually overlapping, sets of taxa. In many instances, multiple consensus trees and supertrees represent more relevant evolutionary models than traditional single consensus trees and supertrees. The resolution of multiple consensus trees and supertrees is generally much better than that of single consensus trees or supertrees inferred by conventional methods [43]. Thus, multiple consensus trees and supertrees have the potential of preserving much more plausible information from a set of given gene trees. Clustering seems to be an intuitive natural solution for inferring multiple consensus trees and supertrees. Tree clustering has a direct practical application in evolutionary studies. It allows one to identify sets of genes that have been affected to the same horizontal gene transfer, hybridization, intragenic/intergenic recombination events, or those that have undergone the same ancient gene duplications and gene losses during their evolution [2, 10, 25, 50].

Since the beginning of the Tree of Life inference project [44], the number of studies dealing with supertree theory has grown considerably. The methods described in this paper can be used for inferring multiple alternative subtrees of the Tree of Life as it contains many unresolved clades (i.e. subtrees with high degrees of its internal nodes). From the practical point of view the problem of constructing multiple alternative supertrees is more relevant than that of constructing multiple alternative consensus trees because most of currently available gene trees are not defined on exactly the same sets of taxa. However, to the best of our knowledge, the only study addressing this relevant problem remains the recent work of Tahiri et al. [72]. The authors of this work showed how some remarkable properties of the Robinson and Foulds topological distance (original or normalized) and the k-means partitioning algorithm can be used to achieve very promising tree clustering performance. Finally, in the application section, we showed how this method can be applied to cluster phylogenetic trees from the famous aaRS phylogenetic dataset originally described by Woese et al. [77].

An interesting option for further investigations consists in the use of some other popular tree distances in the objective function of clustering algorithms. Among them, we need to mention the branch score distance [39] and the quartet distance [14], which also have the Euclidean properties as the square root of the Robinson and Foulds distance.

References

1. Ball, G.H., Hall, D.J.: ISODATA, a Novel Method of Data Analysis and Pattern Classification. Stanford Research Institute, Menlo Park (1965)
2. Bapteste, E., Boucher, Y., Leigh, J., et al.: Phylogenetic reconstruction and lateral gene transfer. Trends Microbiol. **12**(9), 406–411 (2004)
3. Barthélemy, J.P., Guénoche, A.: Trees and Proximity Representations. Wiley, Chichester (1991)
4. Barthélemy, J.P., McMorris, F.R.: The median procedure for n-trees. J. Classif. **3**(2), 329–334 (1986)
5. Barthélemy, J.P., Monjardet, B.: The median procedure in cluster analysis and social choice theory. Math. Soc. Sci. **1**(3), 235–267 (1981)
6. Baum, B.R.: Combining trees as a way of combining data sets for phylogenetic inference, and the desirability of combining gene trees. Taxon. **41**(1), 3–10 (1992)
7. Bininda-Emonds, O.R. (ed.): Phylogenetic Supertrees: Combining Information to Reveal the Tree of Life. Springer (2004)
8. Bininda-Emonds, O.R., Cardillo, M., Jones, K.E., et al.: The delayed rise of present-day mammals. Nature. **446**, 507–512 (2007)
9. Boc, A., Diallo, A.B., Makarenkov, V.: T-REX: a web server for inferring, validating and visualizing phylogenetic trees and networks. Nucleic Acids Res. **40**(W1), W573–W579 (2012)
10. Boc, A., Makarenkov, V.: Towards an accurate identification of mosaic genes and partial horizontal gene transfers. Nucleic Acids Res. **39**(21), e144 (2011)
11. Boc, A., Philippe, H., Makarenkov, V.: Inferring and validating horizontal gene transfer events using bipartition dissimilarity. Syst. Biol. **59**(2), 195–211 (2010)
12. Bonnard, C., Berry, V., Lartillot, N.: Multipolar consensus for phylogenetic trees. Syst. Biol. **55**(5), 837–843 (2006)
13. Bradley, P.S., Mangasarian, O.L., Street, W.N.: Clustering via con-cave minimization. Adv. Neural Inf. Process. Syst. **9**, 368–374 (1997)
14. Bryant, D., Tsang, J., Kearney, P.E., et al.: Computing the quartet distance between evolutionary trees. SIAM J. Appl. Math. **9**(11), 285–286 (2000)
15. Bryant, D.: A classification of consensus methods for phylogenetics. DIMACS series in discrete mathematics and theoretical computer science. **61**, 163–184 (2003)
16. Bullwinkle, T.J., Ibba, M.: Emergence and evolution. Top. Curr. Chem. **344**, 43–87 (2014)
17. Caliński, T., Harabasz, J.: A dendrite method for cluster analysis. Commun. Stat. Theor. M. **3**(1), 1–27 (1974)
18. Creevey, C.J., Fitzpatrick, D.A., Philip, G.K., et al.: Does a tree-like phylogeny only exist at the tips in the prokaryotes? Proc. R. Soc. Lond. B Biol. Sci. **271**(1557), 2551–2558 (2004)
19. Creevey, C.J., McInerney, J.O.: Clann: investigating phylogenetic information through supertree analyses. Bioinformatics. **21**(3), 390–392 (2005)
20. Darlu, P., Guénoche, A.: TreeOfTrees method to evaluate the congruence between gene trees. J. Classif. **28**, 390–403 (2011)
21. Daubin, V., Gouy, M., Perrière, G.: Bacterial molecular phylogeny using supertree approach. Genome Inform. **22**, 155–164 (2001)
22. Day, W.H.: Optimal algorithms for comparing trees with labeled leaves. J. Classif. **2**, 7–28 (1985)
23. de Amorim, R.C., Mirkin, B.: Minkowski metric, feature weighting and anomalous cluster initializing in K-means clustering. Pattern Recogn. **45**(3), 1061–1075 (2012)
24. de Amorim, R.C., Makarenkov, V.: Applying subclustering and Lp distance in weighted K-means with distributed centroids. Neurocomputing. **173**, 700–707 (2016)
25. Diallo, A.B., Makarenkov, V., Blanchette, M.: Finding maximum likelihood Indel scenarios. In: Bourque, G., El-Mabrouk, N. (eds) comparative genomics. RCG 2006. Lect. Notes Comput. Sci. **4205**, 171–185 (2006)

26. Dohm, J.C., Vingron, M., Staub, E.: Horizontal gene transfer in aminoacyl-tRNA synthetases including leucine-specific subtypes. J. Mol. Evol. **63**(4), 437–447 (2006)
27. Dong, J., Fernández-Baca, D., McMorris, F.R.: Constructing majority-rule supertrees. Algorithms Mol. Biol. **5**(1), 2 (2010)
28. Farris, J.S.: Hennig86, Version 1.5. Distributed by the Author, Port Jefferson Station, New York (1988)
29. Faurby, S., Eiserhardt, W.L., Baker, W.J., et al.: An all-evidence species-level supertree for the palms (Arecaceae). Mol. Phylogenet. Evol. **100**, 57–69 (2016)
30. Felsenstein, J.: Confidence limits on phylogenies: an approach using the bootstrap. Evolution. **39**(4), 783–791 (1985)
31. Felsenstein, J.: Alternating least squares approach to inferring phylogenies from pairwise distances. Syst. Biol. **46**(1), 101–111 (1997)
32. Gascuel, O.: Mathematics of Evolution and Phylogeny, pp. 121–142. Oxford University Press, Oxford (2005)
33. Godwin, R.C., Macnamara, L.M., Alexander, R.W., et al.: Structure and dynamics of tRNAMet containing core substitutions. ACS Omega. **3**(9), 10668–10678 (2018)
34. Guénoche, A.: Multiple consensus trees: a method to separate divergent genes. BMC Bioinform. **14**(1), 46 (2013)
35. Haeckel, E.: Generelle Morphologie der Organismen [General Morphology of the Organisms]. G. Reimer, Berlin (1866)
36. Huang, Z.: Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Min. Knowl. Discov. **2**, 283–304 (1998)
37. Kaufman, L., Rousseeuw, P.J.: Partitioning around medoids (program PAM), pp. 68–125. Wiley Series in Probability and Statistics (1990)
38. Kimball, R.T., Oliveros, C.H., Wang, N., et al.: A phylogenomic supertree of birds. Diversity (2019)
39. Kuhner, M.K., Felsenstein, J.: A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. Mol. Biol. Evol. **11**(3), 459–468 (1994)
40. Lloyd, S.P.: Binary block coding. Bell. Labs Tech. J. **36**(2), 517–535 (1957)
41. Lord, E., Leclercq, M., Boc, Diallo, A.B., Makarenkov, V.: Armadillo 1.1: an original workflow platform for designing and conducting phylogenetic analysis and simulations. PLoS One. **7**(1), e29903 (2012)
42. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. **1**(14), 281–297 (1967)
43. Maddison, D.R.: The discovery and importance of multiple islands of most-parsimonious trees. Syst. Biol. **40**(3), 315–328 (1991)
44. Maddison, D.R., Schulz, K.S., Maddison, W.P.: The tree of life web project. Zootaxa. **1668**, 19–40 (2007)
45. Makarenkov, V., Leclerc, B.: Circular orders of tree metrics, and their uses for the reconstruction and fitting of phylogenetic trees. Math Hierarch. Biol., 183–208 (1996)
46. Makarenkov, V.: Propriétés combinatoires des distances d'arbre: Algorithmes et applications. Doctoral dissertation. EHESS, Paris (1997)
47. Makarenkov, V., Leclerc, B.: An algorithm for the fitting of a tree metric according to a weighted least-squares criterion. J. Classif. **16**(1), 3–26 (1999)
48. Makarenkov, V., Leclerc: Comparison of additive trees using circular orders. J. Comput. Biol. **7**(5), 731–744 (2000)
49. Makarenkov, V., Legendre, P.: Improving the additive tree representation of a dissimilarity matrix using reticulations. In: Data Analysis, Classification, and Related Methods, pp. 35–40. Springer, Berlin/Heidelberg (2000)
50. Makarenkov, V., Legendre, P., Desdevises, Y.: Modelling phylogenetic relationships using reticulated networks. Zool. Scr. **33**(1), 89–96 (2004)

51. Makarenkov, V., Mazoure, B., Rabusseau, G., et al.: Horizontal gene transfer and recombination analysis of SARS-CoV-2 genes helps discover its close relatives and shed light on its origin. BMC Ecol. Evol. **21**, 5 (2021)
52. Mank, J.E., Promislow, D.E.L., Avise, J.C.: Phylogenetic perspectives in the evolution of parental care in ray-finned fishes. Evolution. **59**, 1570–1578 (2005)
53. Margush, T., McMorris, F.R.: Consensus n-trees. B Math. Biol. **43**(2), 239–244 (1981)
54. McMorris, F.R., Wilkinson, M.: Conservative supertrees. Syst. Biol. **60**(2), 232–238 (2011)
55. Mirkin, B.: Mathematical classification and clustering, p. 1206. Kluwer Academic Publisher (1996)
56. Mirkin, B.: Clustering for data mining: a data recovery approach, p. 910. Chapman and Hall/CRC (2005)
57. Mirkin, B., Fenner, T.I., Galperin, M.Y., et al.: Algorithms for computing parsimonious evolutionary scenarios for genome evolution, the last universal common ancestor and dominance of horizontal gene transfer in the evolution of prokaryotes. BMC Evol. Biol. **3**(1), 1–34 (2003)
58. Moon, J., Eulenstein, O.: Synthesizing large-scale species trees using the strict consensus approach. J. Bioinforma. Comput. Biol. **15**(3), 1–17 (2017)
59. Nelson, G.: Cladistic analysis and synthesis: principles and definitions, with a historical note on Adanson's Familles des Plantes (1763–1764). Syst. Zool. **28**, 1–21 (1979)
60. Ragan, M.A.: Phylogenetic inference based on matrix representation of trees. Mol. Phylogenet. Evol. **1**(1), 53–58 (1992)
61. Robinson, D.F., Foulds, L.R.: Comparison of phylogenetic trees. Math. Biosci. **53**(1–2), 131–147 (1981)
62. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. **20**, 53–65 (1987)
63. Saitou, N.: Property and efficiency of the maximum likelihood method for molecular phylogeny. J. Mol. Evol. **27**, 261–273 (1988)
64. Sharaf, A., Gruber, A., Jiroutová, K., et al.: Characterization of aminoacyl-tRNA synthetases in Chromerids. Genes. **10**(8), 582 (2019)
65. Silva, A.S., Wilkinson, M.: On defining and finding islands of trees and mitigating large Island bias. Syst. Biol. **70**(6), 1282–1294 (2021)
66. Sokal, R.R., Michener, C.A.: A statistical method for evaluating systematic relationships. Kansas Univ. Sci. Bull. **38**, 1409–1438 (1958)
67. Sokal, R.R., Rohlf, F.J.: Syst. Zool. **30**, 309–325 (1981)
68. Stockham, C., Wang, L.S., Warnow, T.: Statistically based postprocessing of phylogenetic analysis by clustering. Bioinformatics. **18**(1), 285–293 (2002)
69. Swofford, D.L.: PAUP: Phylogenetic Analysis Using Parsimony, Version 3.0q. Illinois Natural History Survey, Champaign (1991)
70. Swofford, D.L., Olsen, G.J.: Phylogeny reconstruction. In: Hillis, D.M., Moritz, C. (eds.) Molecular systematics, pp. 411–501. Sinauer Associates, Sunderland (1990)
71. Tahiri, N., Willems, M., Makarenkov, V.: A new fast method for inferring multiple consensus trees using k-medoids. BMC Evol. Biol. **18**(1), 48 (2018)
72. Tahiri, N., Fichet, B., Makarenkov, V.: Building alternative consensus trees and supertrees using k-means and Robinson and Foulds distance, bioinformatics (in press), btac326 (2022)
73. Tibshirani, R., Walther, G., Hastie, T.: Estimating the number of clusters in a data set via the gap statistic. J. R. Statistical Soc. Ser. B. **63**(2), 411–423 (2001)
74. Unvert, K.E., Kovacs, F.A., Zhang, C., et al.: Evolution of leucyl-tRNA synthetase through eukaryotic speciation. Am. J. Undergrad. Res. **14**, 69–83 (2017)
75. Warnow, T. Supertree Construction: Opportunities and Challenges. ArXiv eprints, (2018). https://arxiv.org/abs/1805.03530
76. Wilkinson, M., Cotton, J.A., Lapointe, F.J., et al.: Properties of supertree methods in the consensus setting. Syst. Biol. **56**(2), 330–337 (2007)
77. Woese, C.R., Olsen, G.J., Ibba, M., et al.: Aminoacyl-tRNA synthetases, the genetic code, and the evolutionary process. Microbiol. Mol. Biol. Rev. **64**(1), 202–236 (2000)

Anomaly Detection with Neural Network Using a Generator

Alexander S. Markov, Evgeny Yu. Kotlyarov, Natalia P. Anosova, Vladimir A. Popov, Yakov M. Karandashev, and Darya E. Apushkinskaya

1 Introduction

Full body scanners are often used in facilities that require increased security control. They allow to make quickly a picture of a person in the X-ray range, where the operator of the full body scanner (FBS) can see all the objects on the body and visually confirm the presence of prohibited items.

The process has a number of significant drawbacks, including those related to the human factor: a manual analysis of the image requires considerable time and attention, which leads to tiredness of the FBS operator and can have a negative impact on the quality of image analysis. This process can be substantially automated, making it cheaper for the organization and more comfortable for the person.

Deep neural networks have been used to solve the problem. This paper presents ways to preprocess the data, create synthetic data with generator module to augment the dataset and train U-Net model with it. Analysis of the results are also provided.

A. S. Markov (✉) · E. Yu. Kotlyarov · N. P. Anosova · V. A. Popov · D. E. Apushkinskaya
Peoples' Friendship University of Russia (RUDN University), Moscow, Russian Federation
e-mail: anosova-np@rudn.ru; popov-va@rudn.ru; apushkinskaya_de@pfur.ru

Y. M. Karandashev
Peoples' Friendship University of Russia (RUDN University), Moscow, Russian Federation

Scientific Research Institute for System Analysis of Russian Academy of Sciences, Moscow, Russian Federation
e-mail: karandashev@niisi.ras.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_14

2 Problem Statement

It is required to develop a solution that matches each X-ray image with a Boolean mask, where true values correspond to the pixels of anomaly objects, such as phones, weapons, metal objects, etc.

The dataset is provided by a company specializing in the development of FBS. It includes 1654 original human body photos taken from four different FBS in the X-ray spectrum. Each picture is a 16-bit image in tiff format of $\sim 1600 \times 500$ pixels. The images contain various anomalies such as: clothing, accessories, weapons, prosthetics, etc.

Let us highlight three problems with the data:

1. The anomalies are faintly visible in the original images.
2. Different FBS produce images with different distribution of pixel values. For this reason, the original data are not suitable for automatic processing by a neural network.
3. Dataset contains very few amount of really dangerous items such as weapons, knifes etc.

Thus, we have to develop an algorithm for image preprocessing which solves our first and second tasks. After that, it is necessary to label the anomalies on the images. Finally, to automate this process, we have to train the neural network to detect anomalies.

3 Methods

3.1 Overview of Existing Approaches

Segmentation of objects on X-ray images is a very common problem. Firstly, the classical image processing methods [1, 2] used to solve this problem. Later, convolutional neural networks are started to be applied. In [3, 4], SegNet architecture and its modification, simplifying the original network and allowing to perform training on a small set of data, were provided. A further development of SegNet is the XNet architecture [5] perfectly fitted to X-ray images, especially for segmentation of soft tissue and bones.

Segmentation problem has been particularly widespread in the medical field. A huge number of works were dedicated to segmentation of cellular structures [6–8]. The most widely used architecture for segmentation problem is U-Net [6]. It uses only convolutional layers, which allows to pass images of arbitrary size to the input layer and get a mask with classes on the output layer.

3.2 Image Preprocessing

For the data labeling process and for improving the visual perception the original 16-bit images were modified. The image preprocessing algorithm consists of a sequence of transformations.

Firstly, we subtract the minimum values of the pixels from all pixels of the image. Secondly, the Gaussian Filter is applied to reduce the noise. After that, we apply the following pipeline of filters according to our previous work [9]:

$$\text{Threshold truncating} \rightarrow \text{Histogram Equalization} \rightarrow \text{Threshold truncating}$$
$$\rightarrow \text{Adaptive Histogram Equalization} \rightarrow \text{Threshold truncating}$$

The result of this algorithm can be seen in Fig. 1.

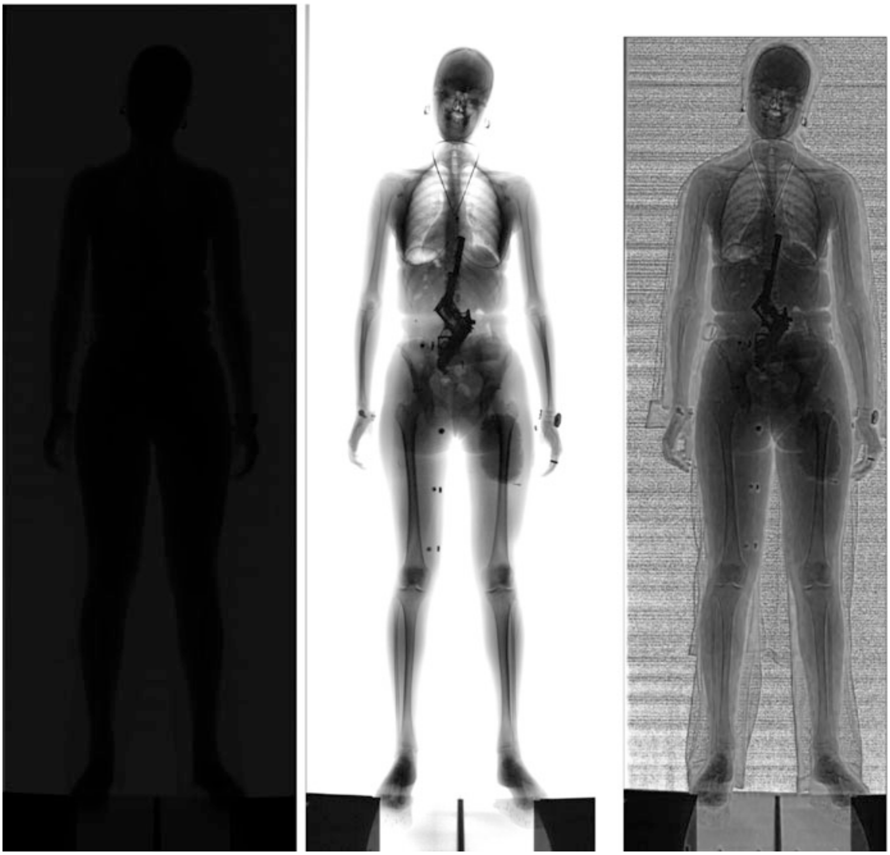


Fig. 1 The result of the novel algorithm. Original image (left), old preprocessing (middle), and current preprocessing (right)

3.3 Labeling

After processing the dataset, the anomalies became much more visible. Due to this, anomaly objects were manually labeled in the images. The detailed description can be found in [9].

3.4 Neural Network

To solve the problem of anomaly detection the U-Net network with three levels was chosen (see Fig. 2). Training was performed with the PyTorch framework on Nvidia gtx 1080ti graphics processor with 11gb of memory. Parameters of training are as follows: Adam algorithm [10] as optimizer, learning rate parameter 0.001, batch size 10. Training is performed on 1454 images. The size of the test dataset is 200 images.

The input of the model is a preprocessed grayscale image of size 512 × 512. The output is a Boolean mask of the same size characterizing the probabilities of finding anomalies in the corresponding parts of the image.

To augment the variability of the data we used a standard random cropping, which allowed us to generate a large number of images containing different body parts. Also the additional anomalies were added with the generator described below.

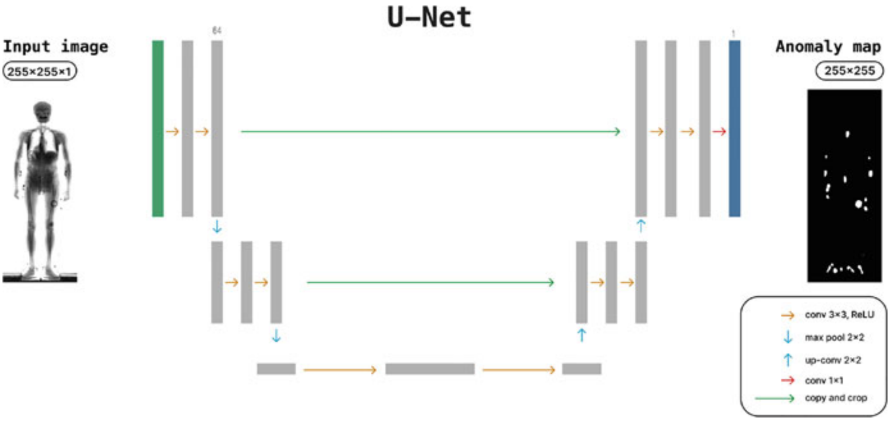


Fig. 2 The architecture of the neural network

4 Generator

In the process of research, we found that customer-supplied images contain practically no dangerous or anomalous objects. However, in order to get stable and high-quality results, it necessary to have a comprehensive training dataset.

To solve this problem we created the generator, an additional module that generate anomalies in real time and add them to the source images before pass them to neural network, thus increasing the number of anomaly examples during the training.

The schematic diagram of the generator is shown in Fig. 3. The upper arrow represents the learning process of the network without the generator, the lower one—with it.

The generator works in two modes:

1. generating random geometric polygons with pixel intensities picked from normal distribution, followed by applying a Gaussian filter to the polygon area;
2. choosing random objects from a manually created library of anomalies, containing various objects (such as weapons, clothing, accessories etc) in the X-ray spectrum; these objects are put on the original image using the following formula:

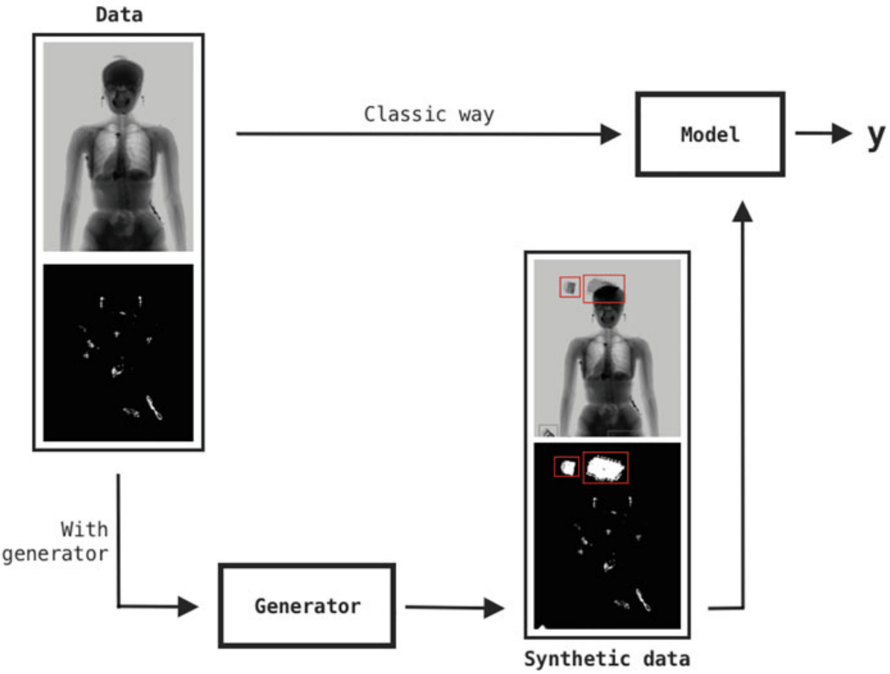


Fig. 3 Schematic diagram of the generator

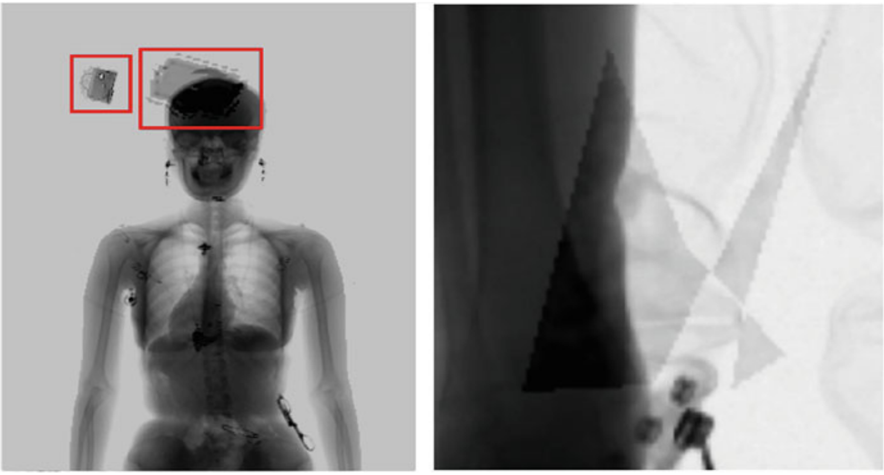
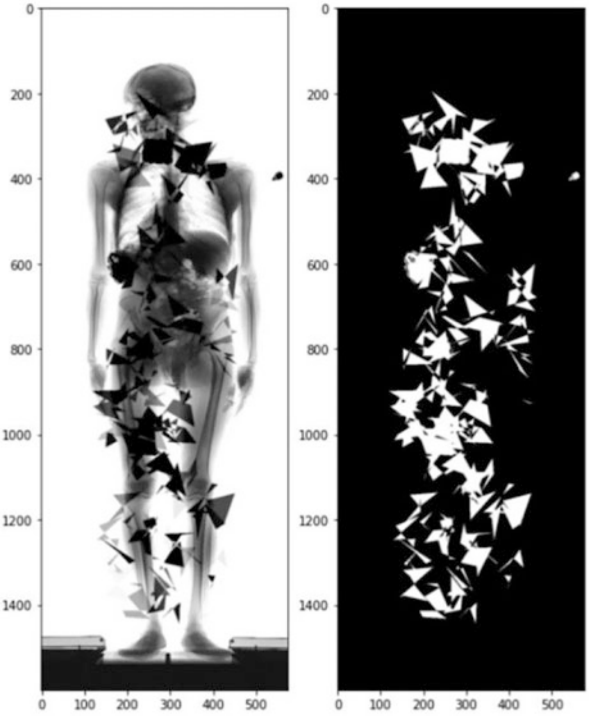


Fig. 4 Dual mode of the generator work

Fig. 5 Polygons overlaid by a generator on the human body



$$\mathrm{img} = \mathrm{img} - \gamma * \left([2^{16} - 1] - \mathrm{anomaly}\right)$$

Dual modes of generator work are illustrated in Figs. 4 and 5. It can be seen that the distribution of pixel intensity in the modified image regions is visually similar with the regions with native anomalies.

To increase the statistical diversity of generated examples, the following augmentation techniques were used:

- increase or decrease the generated anomaly by a random magnifying factor;
- rotate the generated anomaly by a random angle;
- change the pixel intensity of the superimposed object due to random coefficient γ.

5 Results

As a result of the work carried out, we developed the algorithm that preprocesses the images and maps them into Boolean mask with potentially dangerous or anomalous objects highlighted. The results of this model are shown in Fig. 6. One can see that the neural network has learned to identify large anomaly objects, but the borders of these objects are distinguished poorly.

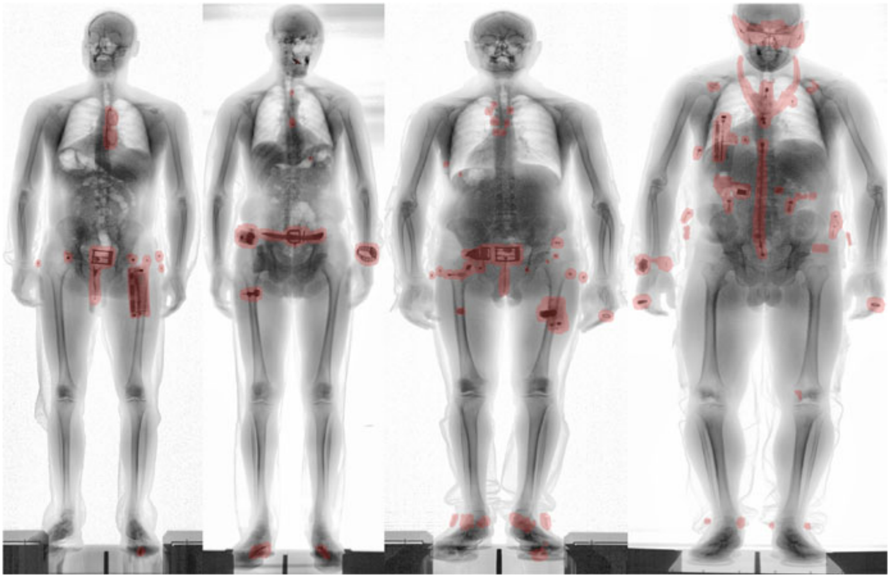


Fig. 6 Several examples of preprocessed images with resulting masks

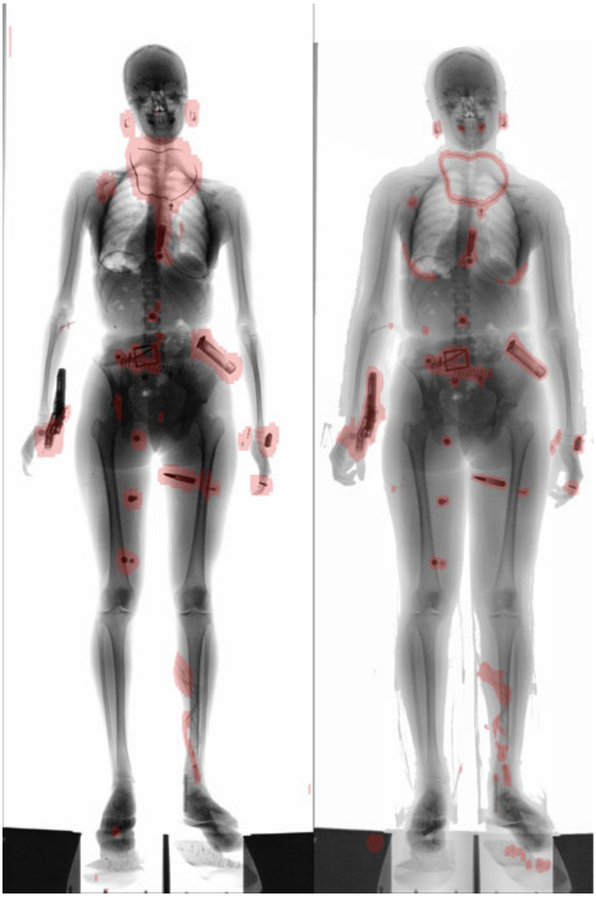


Fig. 7 Result without generator (left) and with generator (right)

On the CPU, it takes 4 seconds on average to process one image. Due to the auxiliary augmentation with the generator the results obtained by this model seems to be more accurate (see Figs. 7 and 8) then results provided in the work [9].

6 Conclusions

The proposed data preprocessing scheme can be used for various personal inspection scanners. It allows normalizing the data from different personal inspection devices (or with different radiation settings), and all objects become visible to humans. Using the proposed processing approach, a training dataset was generated.

The generator module allows us to greatly diversify the training samples, increasing the quality of the final neural network.

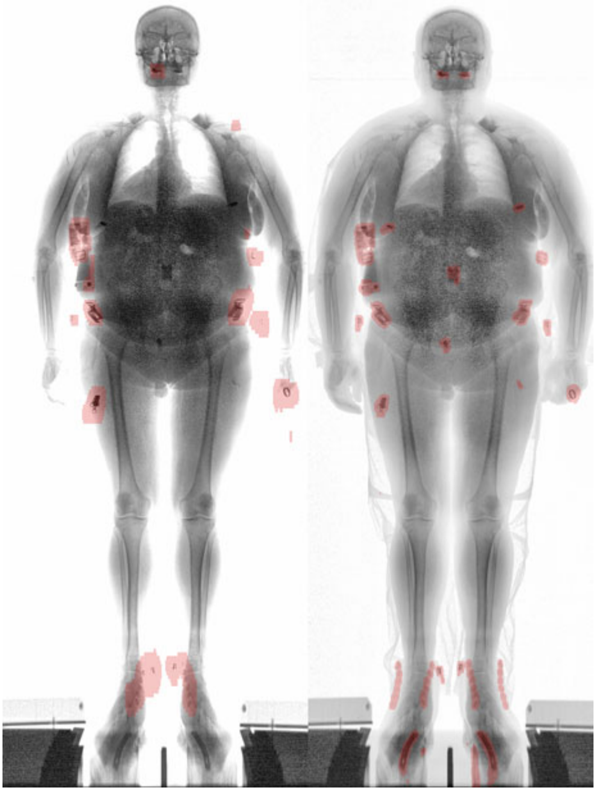


Fig. 8 Result without generator (left) and with generator (right)

The U-Net neural network was trained based on the created dataset. The segmentation quality of the trained model generally allows recognizing the anomalies of any size. The model can be used at industrial sites, as a means of automating the FBS to find objects such as weapons, phones, metal ingots and others, significantly increasing the speed and effectiveness of the operator.

References

1. Sharma, N., Aggarwal, L.M.: Automated medical image segmentation techniques. J. Med. Phys. **35**(1), 3–14 (2010)
2. Mansoor, A., Bagci, U., Foster, B., Xu, Z., Papadakis, G.Z., Folio, L.R., Udupa, J.K., Mollura, D.J.: Segmentation and image analysis of abnormal lungs at CT: current approaches, challenges, and future trend. Radiographics **35**(4), 1056–1076 (2015)
3. Badrinarayanan, V., Handa, A., Cipolla, R.: Segnet: a deep convolutional encoder-decoder architecture for robust semantic pixel-wise labeling (2015). e-print, arXiv:1505.07293. https://arxiv.org/abs/1505.07293
4. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2481–2495 (2017)

5. XNet: a convolutional neural network (CNN) implementation for medical X-ray image segmentation suitable for small datasets. In: Proc. of SPIE Medical Imaging Conference, pp. 453–463 (2019). https://doi.org/10.1117/12.2512451
6. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015, pp. 234–241. Springer International Publishing, Cham (2015)
7. Ciresan, D., Giusti, A., Gambardella, L. M., Schmidhuber, J.: Deep neural networks segment neuronal membranes in electron microscopy images. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 25, pp. 2843–2851. Curran Associates (2012)
8. Arganda-Carreras, I., Turaga, S.C., Berger, D.R., et al.: Crowdsourcing the creation of image segmentation algorithms for connectomics. Front. Neuroanatomy (2015). https://doi.org/10.3389/fnana.2015.00142
9. Markov, A.S., Kotlyarov, E.Yu., Anosova, N.P., Karandashev, Ya.M., Apushkinskaya, D.E.: Using neural networks to detect anomalies in X-ray images obtained with full-body scanners. Autom. Remote Control **83**(10), 1507–1516 (2022)
10. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization (2014). e-print, arXiv:1412.6980. https://arxiv.org/abs/1412.6980

Controllability of Triangular Systems with Phase Space Change

Irina Sergeevna Maximova

1 Literature Review

Problems with changing phase space are a subclass of the so-called composite (hybrid) systems. These problems are characterized by the fact that at successive time intervals the motion of an object is described by different systems of differential equations and by some couplings for trajectories' dockings.

The emergence of this type of problems is initially associated with the study of multistage processes of space flight [1]. Such models were characterized by fixed switching sequences. In this case, when the trajectory reaches a certain manifold, the dimensionality of the space, the control vector, or the equations of motion change. After that this class of problems began to be applied to the physical problems of launching a rocket from a controlled object (submerged or surface).

Reducing or increasing the dimensionality of phase spaces in problems with variable dimensionality is closely related to the concepts of aggregation and decomposition. One of the peculiarities of aggregation is dimensionality reduction. The most frequently encountered situation that makes the use of the aggregation required is dealing with a large set of data that is poorly observable and difficult to "work with". Decomposition methods, on the contrary, lead to an increase in dimensionality. Decomposition allows to carry out a consistent breakdown of the system into subsystems, which, in turn, can be broken down into their constituent parts. As a result, decomposition allows us to structure large and complex objects into subsystems that have the required properties. For example, [2] applies a method of sequential aggregation of variables to bring a nonlinear system to a special form,

I. S. Maximova (✉)
S. M. Nikolsky Mathematical Institute, Peoples' Friendship University of Russia (RUDN University), Moscow, Russian Federation
e-mail: maximova-is@rudn.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_15

with reduced dimensionality. The publication [3] researched a method of forecasting performance in technological process control systems based on the use of artificial neural networks. Dimensionality reduction in this model is made by pre-filtering of data. The problems of optimization of composite systems were studied by V.G. Boltyansky [4], L.T. Ashchepkov [5], V.N. Rozova [6, 7], V. R. Bargsegayn [8].

However, in the above-mentioned works, the matter of optimization for given quality criteria was mainly studied. In the meantime, all typical theorems of the existence of optimal control assume the existence of at least one admissible control, generating a trajectory satisfying the given boundary conditions. The latter is the essence of the controllability problem. Thus, the problem of controllability is important and relevant for solving optimization problems.

The controllability problems with a phase space change were researched by the author in [9–11]. In this paper, nonlinear systems of the so-called triangular form are considered. An important feature of this class of systems is that with a certain replacement of variables they are mapped to linear systems. The controllability of linear systems is thoroughly studied, which allows us to use various criteria to study them. Triangular systems describe a number of physical processes, such as orientation of a satellite in orbit, control of a robotic manipulator, etc. The class of triangular systems was first introduced and reviewed by V. I. Korobov [12]. The approach proposed by V. I. Korobov was further developed in [13].

In the theory of optimal control the following two problems play an important role: under what conditions there is a control that transfers the system from one position to another at certain interval of time and, if such control exists, its analytical representation has to be found. The first problem for linear systems is fully solved by now. A number forms of necessary and sufficient conditions for the existence of controls have been obtained. For nonlinear systems, however, the problem of controllability is far from being solved due to the diversity of classes of nonlinear systems and the complexity of their description. The problem of constructing an analytical representation of a control transitioning the system from one point to another was first solved by Kalman in [14, 15]. Broad classes of controls in explicit form for some systems transitioning an object from one position to another were obtained by V. I. Korobov, G. M. Sklyarov in [16].

In this paper for the problem of controllability with a phase space change, where the motion of the object is described by two nonlinear triangular systems on consecutive time segments, the conditions of controllability of the object from the initial set of one space to the finite set of the other space are obtained. Also in the paper the explicit form of the trajectories, which carry out this transition, was obtained.

2 Problem Statement

In two phase spaces $X = R^n$ and $Y = R^m$ of variables $x = (x_1, \dots, x_n)$ and $y = (y_1, \dots, y_m)$ the motion of the controlled object is described by the following non-linear systems of differential equations:

$$\begin{cases} \frac{dx_i}{dt} = f_i(x_1, \dots, x_{i+1}), \quad i = 1, \dots, n-1, \\ \frac{dx_n}{dt} = f_n(x_1, \dots, x_n; u). \end{cases} \tag{1}$$

$x \in X, \quad t \in [0, \tau], u(t) \in U.$

$$\begin{cases} \frac{dy_k}{dt} = g_k(y_1, \dots, y_{k+1}), \quad k = 1, \dots, m-1, \\ \frac{dy_m}{dt} = g_m(y_1, \dots, y_m; v). \end{cases} \tag{2}$$

$y \in Y, \quad t \in [\tau, T], \quad v(t) \in V.$

The time moments τ and T are given. In the space X the initial set M_0 and the transition hyperplane $\Gamma = (x, c)$ are given. The trajectories are docked using a given mapping $q : X \to Y$, $y(\tau) = q(x(\tau))$. The transition from one space to another is also implemented by means of this mapping. In the space Y there is a finite set M_1.

The controlled object moves according to the following scheme: on the time interval $[0, \tau]$ the object moves from the initial set M_0 by solutions of the system (1), at time τ the object gets on Γ and the transition to space Y occurs under the action of a linear mapping $q : X \to Y$, $q(x(\tau)) = y(\tau)$. The resulting point $y(\tau)$ is the starting point for the motion of the object in space Y. Further movement on the time interval $[\tau, T]$ is performed by the object from the point $y(\tau)$ to the set M_1 by solutions of the system (2). And $y(\tau) \notin M_1$ (otherwise the problem is solved).

The problem is to find the conditions under which the object described by the systems (1) and (2), is controllable on $[0, T]$ from the set M_0 of space X to the set M_1 of space Y. An object described by the systems (1) and (2),is called controllable from M_0 to M_1, [10] if on the segments $[0, \tau]$ and $[\tau, T]$ there are such admissible controls $u(t) \in U$ and $v(t) \in V$, that their corresponding solutions of the systems satisfy the boundary conditions $x(0) \in M_0$, $x(\tau) \in \Gamma$ and $y(\tau) = q(x(\tau))$, $y(T) \in M_1$.

The controllability conditions of the object described by systems (1) and (2) can be formulated as the following statement.

3 Main Result

Let functions $f_i(x_1, \cdots, x_{i+1}), i = 1, \dots, n$ and $g_k(y_1, \cdots, y_{k+1}), k = 1, \dots, m$ the systems (1) and (2), have the continuous partial derivatives up to $(n - i + 1)$-th and $(m - k + 1)$-th orders inclusive and let

$$\left| \frac{\partial f_i}{\partial x_{i+1}} \right| \geq a > 0, \quad atall \quad x_1, \cdots, x_{n+1},$$

$$\left| \frac{\partial g_k}{\partial y_{k+1}} \right| \geq b > 0, \quad atall \quad y_1, \cdots, y_{m+1},$$

where a and b—are constants independent of $x_1, \cdots, x_{n+1}$ and $y_1, \cdots, y_{m+1}$ respectively. And let the docking conditions for the trajectories $y(\tau) = q(x(\tau))$ be satisfied. Then the object described by the systems (1) and (2) is controllable from the initial set M_0 of space X to the finite set M_1 of space Y.

Let us investigate the motion of an object in space X from the initial set M_0 to the transition hyperplane Γ on the time interval $[0, \tau]$. Let's examine the following controllability problem—to choose the control u so as to get from the point $x_0 \in M_0$ to the point $x_1 \in \Gamma$ by the solutions of the system (1). Here is a way of constructing a control that solves the problem.

Let's examine the system (1):

$$\begin{cases} \dot{x}_1 = f_1(x_1, x_2), \\ \dot{x}_2 = f_2(x_1, x_2, x_3), \\ \dots \\ \dot{x}_{n-1} = f_{n-1}(x_1, \dots, x_n), \\ \dot{x}_n = f_n(x_1, \dots, x_n; u). \end{cases}$$

Lets introduce substitution of the variables as follows:

$$\begin{aligned} & z_1 = x_1 \equiv F_1(x_1), \\ & \qquad z_i = \frac{\partial F_{i-1}}{\partial x_i} f_1(x_1, x_2) + \dots + \frac{\partial F_{i-1}}{\partial x_{i-1}} f_{i-1}(x_1, \dots, x_i) \equiv \\ & \qquad \equiv F_i(x_1, \dots, x_i), i = 2, \dots, n. \end{aligned} \tag{3}$$

The introduced substitution of the variables can be written in the form $z = F(x)$, where

$$F(x) = \begin{pmatrix} F_1(x_1) \\ F_2(x_1, x_2) \\ \dots \\ F_n(x_1, \dots, x_n) \end{pmatrix} = \begin{pmatrix} P_0 x_1 \\ P_1 P_0 x_1 \\ \dots \\ P_{n-1} P_{n-2} \cdot \dots \cdot P_0 x_1 \end{pmatrix} \tag{4}$$

where $P_0, P_1, \dots, P_{n-1}$—differential operators of the following form

$$P_0 \equiv I, P_i = f_1 \frac{\partial}{\partial x_1} + \dots + f_i \frac{\partial}{\partial x_i}, i = 1, \dots, n-1, \tag{5}$$

I—is an identical operator. Let's designate a new control as z_{n+1}:

$$\begin{aligned} z_{n+1} = \frac{\partial F_n}{\partial x_1} f_1(x_1, x_2) + \dots + \frac{\partial F_n}{\partial x_n} f_n(x_1, \dots + x_n, x_{n+1}) \equiv \\ \equiv F_{n+1}(x_1, \dots + x_n, x_{n+1}) = P_n P_{n-1} P_{n-2} \cdot \dots \cdot P_0 x_1, \end{aligned} \tag{6}$$

where $P_n = f_1 \frac{\partial}{\partial x_1} + \ldots + f_n \frac{\partial}{\partial x_n}$. After this replacement of variables, the system (1) is reduced to the form

$$\dot{z}_i = z_{i+1}, i = 1, \ldots, n. \tag{7}$$

The resulting linear system (7) is fully controllable at time period τ. This is proved by from the Kalman rank criterion. It is known that if a system

$$\dot{x} = Ax + Bu$$

is linear on x and u, and if the rank of the matrix $(B, AB, \ldots, A^{n-1}B)$ is n, then the system is completely controllable at time period τ.

A system of equations is called completely controllable at time period τ, if there exists an admissible control $u(t)$ with which the corresponding trajectory of the system connects any given points at time period τ.

Since $x_0 \in M_0$ -is an arbitrary point of the initial set, and $x_1 \in \Gamma$—is an arbitrary point on the transition hyperplane, with the complete controllability of the system (7), there exists an admissible control transitioning the object from the point x_0 to the point x_1 to the point τ.

In the system (7) the new control z_{n+1} in a from of a function from t will be chosen in the way so that in time period τ we can get from a point

$$z(0) = (F_1(x_{10}), \ldots, F_n(x_{10}, \ldots, x_{n0}))^T \tag{8}$$

to the point

$$z(\tau) = (F_1(x_{11}), \ldots, F_n(x_{11}, \ldots, x_{n1}))^T. \tag{9}$$

A control z_{n+1}, e.g. [13], can be chosen as

$$z_{n+1}(t) = -b_0^T e^{-A_0^T t} N^{-1}(z_0 - e^{-A_0 \tau} z_\tau),$$

where

$$N = \int_0^\tau e^{-A_0 t} b_0 b_0^T e^{-A_0^T t} dt.$$

By substituting the functions $z_i(t), i = 1, \ldots, n+1$ into the left hand sides of (3) and (6), we consecutively find the functions $x_1(t), \ldots, x_{n+1}(t)$ from these equations. Indeed, the first equality from the formulas (3) and (6) gives $x_1 \equiv F(x_1) = z_1(t)$. If the functions $x_1(t), \ldots, x_{i-1}(t)$ are found through $z_1(t), \ldots, z_{i-1}(t)$ (let $x_j(t) = H_j(z_1(t), \ldots, z_j(t)), j = 1, \ldots, i-1$), then the function $x_i(t)$ is found from i to the equality of the relations (3) and (6):

$$F_i(x_1(t), \ldots, x_{i-1}(t), x_i(t)) = z_i(t). \tag{10}$$

For the solvability of Eq. (10) it is sufficient to establish that

$$\frac{\partial F_i}{\partial x_i} = \frac{\partial f_1}{\partial x_2}\frac{\partial f_2}{\partial x_3}\cdot\ldots\cdot\frac{\partial f_{i-1}}{\partial x_i}, i = 2,\ldots,n+1 \tag{11}$$

then $|\frac{\partial F_i}{\partial x_i}| \geq a > 0$, which means that the function $z_i = F_i(x_1,\ldots,x_i)$ is strictly monotone on x_i and with the fixed values of $x_1,\ldots,x_{i-1}$ and changing x_i continuously maps the interval $(-\infty,\infty)$ to the interval $(-\infty,\infty)$, which means that Eq. (10) is solvable. The relation (11) comes from (3) and (6) as

$$\frac{\partial F_2}{\partial x_2} = \frac{\partial f_1}{\partial x_2},$$

$$\frac{\partial F_3}{\partial x_3} = \frac{\partial}{\partial x_3}\left(\frac{\partial F_2(x_1,x_2)}{\partial x_1}f_1(x_1,x_2) + \frac{\partial F_2(x_1,x_2)}{\partial x_2}f_2(x_1,x_2,x_3)\right) = \frac{\partial f_1}{\partial x_2}\frac{\partial f_2}{\partial x_3}$$

etc. Let us show that the functions

$$x_i(t) = H_i(z_1(t),\ldots,z_n(t)), i = 1,\ldots,n$$

satisfy the system (1) under the obtained control

$$x_{n+1} = H_{n+1}(z_1(t),\ldots,z_{n+1}(t)),$$

which is measurable, because $H_{n+1}, z_i, i = 1,\ldots,n$ are continuous from their arguments, and $z_{n+1}(t)$ is continuous by t. From (3) we have

$$\dot{z}_i = \sum_{j=1}^{i}\frac{\partial F_i(x_1(t),\ldots,x_i(t))}{\partial x_j}\frac{dx_j}{dt}. \tag{12}$$

Since $\dot{z}_i = z_{n+1}(t) = F_{i+1}(x_1(t),\ldots,x_{i+1}(t))$, then

$$\dot{z}_i = \frac{\partial F_i(x_1(t),\ldots,x_i(t))}{\partial x_j}f_j(x_1(t),\ldots,x_{j+1}(t)). \tag{13}$$

Thus, Eqs. (12) and (13) are

$$\frac{dz_i}{dt} = \sum_{j=1}^{i}\frac{\partial F_i(x_1(t),\ldots,x_i(t))}{\partial x_j}\times\left(\frac{dx_j}{dt} - f_j(x_1(t),\ldots,x_{j+1}(t)\right)$$
$$= 0, i = 1,\ldots,n. \tag{14}$$

The Δ determinant of the resulting system with respect to

$$\frac{dx_j}{dt} - f_j(x_1(t),\ldots,x_{j+1}(t)), j = 1,\ldots,n$$

is different from zero, since

$$\Delta = \frac{\partial F_1}{\partial x_1}\frac{\partial F_2}{\partial x_2}\dots\frac{\partial F_n}{\partial x_n} = \left(\frac{\partial f_1}{\partial x_2}\right)^{n-1}\left(\frac{\partial f_2}{\partial x_3}\right)^{n-2}\cdot\ldots\cdot\left(\frac{\partial f_n}{\partial x_n}\right) \neq 0.$$

Then it follows from (14) that

$$\frac{dx_j}{dt} - f_j(x_1(t), \dots, x_{j+1}(t)),\; j = 1, \dots, n$$

where $x_{n+1}(t) = u(t)$. Since the trajectory $z(t)$ passes through the points (8), (9), then due to the unique solvability of the relation (3) with respect to $x_1, \dots, x_n$, the resulting functions $x_i(t)$ satisfy the boundary conditions $x_i(0) = x_{i0}, x_i(\tau) = x_{i1}, i = 1, \dots, n$.

After the object falls on the transition hyperplane Γ, we make a transition to the space Y using the mapping $q : X \to Y$ and obtain the starting point for the motion of the object in the space Y $y(\tau) = q(x(\tau))$. This point does not belong to the finite set $M_1 \in Y$. Thus we obtained the following problem in space Y:

for an object whose motion is described by a system of equations

$$\begin{cases} \frac{dy_k}{dt} = g_k(y_1, \dots, y_{k+1}), \quad k = 1, \dots, m-1, \\ \frac{dy_m}{dt} = g_m(y_1, \dots, y_m; v). \end{cases} \tag{15}$$

$y \in Y = R^m, \quad t \in [\tau, T], \quad v(t) \in V$, find an admissible control v with that the corresponding solution of the system (15) satisfies the boundary conditions $y(\tau) = q(x(\tau))$, $y(T) \in M_1$. Similarly to the space X, we replace the variables and reduce the nonlinear system to a linear one.

$$\begin{aligned} z_1 = y_1 \equiv G_1(y_1), \\ z_k = \frac{\partial G_{k-1}}{\partial y_k} g_1(y_1, y_2) + \ldots + \frac{\partial G_{k-1}}{\partial y_{k-1}} g_{k-1}(y_1, \dots, y_k) \equiv \\ \equiv G_k(y_1, \dots, y_k), k = 2, \dots, m. \end{aligned} \tag{16}$$

We designate the control by

$$\begin{aligned} z_{m+1} = \frac{\partial G_m}{\partial y_1} g_1(y_1, y_2) + \ldots + \frac{\partial G_m}{\partial y_m} g_m(y_1, \ldots + y_m, y_{m+1}) \equiv \\ \equiv G_{m+1}(y_1, \ldots + y_m, y_{m+1}). \end{aligned} \tag{17}$$

As a result of this replacement, the system (15) is reduced to the form

$$\dot{z}_k = z_{k+1}, \quad k = 1, \dots, m. \tag{18}$$

As in the previous case, the system (18) is, by virtue of the Kalman rank criterion, completely controllable. That is, there exists an admissible control v which transfers the object described by this system from any point to any point on the time interval $[\tau, T]$. Due to complete controllability of the system, we assume $y(\tau)$ as the initial point and an arbitrary point $y(T) \in M_1$ as the final point. In the system (18) lets choose a new control z_{m+1} as a function of t so that in time $T - \tau$ to get from the point

$$z(\tau) = (G_1(y_{10}), \dots, G_m(y_{10}, \dots, y_{m0}))^T \tag{19}$$

to the point

$$z(T) = (G_1(y_{11}), \dots, G_m(y_{11}, \dots, y_{m1}))^T. \tag{20}$$

The control z_{m+1} will be chosen as

$$z_{m+1}(t) = b_0^T e^{C_0^T(T-t)} N^{-1}(z_T - e^{C_0^T(T-\tau)} z_\tau),$$

where

$$N = \int_\tau^T e^{C_0(T-t)} b_0 b_0^T e^{C_0^T(T-t)} dt.$$

By substituting the functions $z_i(t), i = 1, \dots, m+1$ into the left-hand sides of formulas (16) and (17), we find the functions $y_1(t), \dots, y_{m+1}(t)$ from obtained equalities. Due to the unique solvability of the relation (16) (which is proved analogously to the space X), the obtained functions $y_1(t), \dots, y_{m+1}(t)$ satisfy the boundary conditions $y_i(\tau) = y_{i\tau}, y_i(T) = y_{iT}, i = 1, \dots, m$. Thus, the controllability of the object described by the systems (1) and (2) from the initial set M_0 of space X to the finite set M_1 of space Y on the time interval $[0, T]$ is proved. The equations of trajectories satisfying the given boundary conditions are also explicitly obtained. Which proves the statement.

Let us consider an example that illustrates this approach to research.

4 Example

In the $X = R^3$ and $Y = R^3$ spaces, the motion of the controlled object is given by the following nonlinear systems of differential equations:

$$\begin{cases} \dot{x}_1 = x_1^4 + x_2, \\ \dot{x}_2 = -4x_1^3 x_2 + x_3, \\ \dot{x}_3 = -28x_1^{10} - 28x_1^6 x_2 + u, \quad t \in [0, 1]. \end{cases} \tag{21}$$

$$\begin{cases} \dot{y}_1 = 2y_1^2 + y_2, \\ \dot{y}_2 = -4y_1y_2 + y_3, \\ \dot{y}_3 = -48y_1^4 - 24y_1^2y_2 + v, \quad t \in [1, 2]. \end{cases} \tag{22}$$

In the space $X = R^3$ the initial set $M_0 = (1, 1, 2)$ is given, in the space $Y = R^3$ the final set $M_1 = (0, -1, 1)$ is given. The motion of the object is carried out according to the following scheme: on the time interval $[0, 1]$, the object moves by solutions of the system (21) from the initial set M_0 to the point $(0, 0, 0)$, then it moves to the space $Y = R^3$, given by the mapping $q(x_1, x_2, x_3) = (y_1, y_2, y_3)$ and further movement on the time interval $[1, 2]$ is performed by solutions of the system (22). It is required to determine whether the object is controllable from the set $M_0 \in X$ to the set $M_1 \in Y$ on the interval $[0, 2]$ and find the trajectories implementing this transition. Let us apply the above approach to the study. Lets consider the motion of the object in the space $X = R^3$. We investigate the controllability problem from the point $x(0) = (1, 1, 2)^T$ to the point $x(1) = (0, 0, 0)^T$ on the segment $[0, 1]$. By replacing the variables

$$\begin{cases} z_1 = x_1, \\ z_2 = x_1^4 + x_2, \\ z_3 = 4x_1^7 + x_3 \end{cases} \tag{23}$$

system (21) is mapped to a linear system

$$\begin{cases} \dot{z}_1 = z_2, \\ \dot{z}_2 = z_3, \\ \dot{z}_3 = u. \end{cases} \tag{24}$$

The resulting linear system (24) is, by virtue of the Kalman rank criterion, completely controllable. We choose the new control so that for the time $T = 1$ we get from the point $z(0) = (1, 2, 6)^T$ to the point $z(1) = (0, 0, 0)^T$. It can be taken [13], for example, as

$$u(t) = -b_0^T e^{-A_0^T t} W^{-1}(z(0) - e^{-A_0 1} z(1)),$$

where W^{-1} is matrix inverse to the matrix

$$W = \int_0^1 e^{-A_0 t} b_0 b_0^T e^{-A_0^T t} dt.$$

Then the control has a form

$$u(t) = \begin{pmatrix} 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ -t & 1 & 0 \\ \frac{t^2}{2} & -t & 1 \end{pmatrix} \begin{pmatrix} 720 & 360 & 60 \\ 360 & 192 & 36 \\ 60 & 36 & 9 \end{pmatrix} \begin{pmatrix} 1 \\ 2 \\ 6 \end{pmatrix} = -900t^2 + 960t - 186.$$

Substituting the obtained control into the system (24) and considering the boundary conditions, we obtain

$$\begin{cases} z_1(t) = -15t^5 + 40t^4 - 31t^3 + 3t^2 + 2t + 1, \\ z_2(t) = -75t^4 + 160t^3 - 93t^2 + 6t + 2, \\ z_3(t) = -300t^3 + 480t^2 - 186t + 6. \end{cases} \tag{25}$$

By making the inverse replacement, we obtain that the trajectories of the system (21) connecting the points $x(0) = (1, 1, 2)^T$ and $x(1) = (0, 0, 0)^T$, on the time interval [0, 1] have the form

$$\begin{cases} x_1(t) = z_1(t) = -15t^5 + 40t^4 - 31t^3 + 3t^2 + 2t + 1, \\ x_2(t) = z_2(t) - x_1^4 = -75t^4 + 160t^3 - 93t^2 + 6t + 2 - (75t^4 + 160t^3 - 93t^2 + 6t + 2)^4, \\ x_3(t) = z_3(t) - 4x_1^7 = -300t^3 + 480t^2 - 186t + 6 - 4(75t^4 + 160t^3 - 93t^2 + 6t + 2)^7. \end{cases} \tag{26}$$

Now, using the mapping $q : X \to Y$, $q(x_1, x_2, x_3) = (y_1, y_2, y_3)$ we move to the space $Y = R^3$. The resulting point $y(1) = q(0, 0, 0) = (0, 0, 0)$ is the initial point when the object moves in this space by solutions of the system (22). Thus, we obtained the following controllability problem: from the point $y(1) = (0, 0, 0)^T$ get to the point $y(2) = (0, -1, 1)^T$ on the time interval $[1, 2]$. Let us reduce the system (22) to a linear one by changing the variables

$$\begin{cases} z_1 = y_1, \\ z_2 = 2y_1^2 + y_2, \\ z_3 = 8y_1^3 + y_3. \end{cases} \tag{27}$$

After this replacement the system will of the form

$$\begin{cases} \dot{z}_1 = z_2, \\ \dot{z}_2 = z_3, \\ \dot{z}_3 = v. \end{cases} \tag{28}$$

Similarly to the previous case, due to the complete controllability of the resulting linear system, the control transitioning the system (28) from the point $z(1) = (0, 0, 0)^T$ to the point $z(2) = (0, -1, 1)^T$ will be chosen as

$$v(t) = b_0^T e^{A_0^T(2-t)} N^{-1}(z(2) - e^{A_0(2-t)} z(1)),$$

where N^{-1}—is the inverse of the matrix

$$N = \int_1^2 e^{A_0(2-t)} b_0 b_0^T e^{A_0^T(2-t)} dt.$$

The control has the form

$$v(t) = \begin{pmatrix}0 & 0 & 1\end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 2-t & 1 & 0 \\ \frac{(2-t)^2}{2} & 2-t & 1 \end{pmatrix} \begin{pmatrix} 720 & -360 & 60 \\ -360 & 192 & -36 \\ 60 & -36 & 9 \end{pmatrix} \begin{pmatrix} 0 \\ -1 \\ 1 \end{pmatrix} = 210t^2 - 612t + 429.$$

Substituting the obtained control into the system (28), we find the trajectories

$$\begin{cases} z_1(t) = 3,5t^5 - 25,5t^4 + 71,5t^3 - 96,5t^2 + 63t - 16, \\ z_2(t) = 17,5t^4 - 102t^3 + 214,5t^2 - 193t + 63, \\ z_3(t) = 70t^3 - 306t^2 + 429t - 193. \end{cases} \tag{29}$$

From the formula (27) we obtain the trajectories of the original system (22) connecting the points $y(1) = (0, 0, 0)^T$ and $y(2) = (0, -1, 1)^T$ on the time interval $[1, 2]$.

$$\begin{cases} y_1(t) = z_1 = 3,5t^5 - 25,5t^4 + 71,5t^3 - 96,5t^2 + 63t - 16, \\ y_2(t) = z_2 - 2y_1^2 = \\ \qquad = 17,5t^4 - 102t^3 + 214,5t^2 - 193t + 63 \\ \qquad\qquad -2(3,5t^5 - 25,5t^4 + 71,5t^3 - 96,5t^2 + 63t - 16)^2, \\ y_3(t) = z_3 - 8y_1^3 = \\ \qquad = 70t^3 - 306t^2 + 429t - 193 \\ \qquad\qquad -8(3,5t^5 - 25,5t^4 + 71,5t^3 - 96,5t^2 + 63t - 16)^3. \end{cases} \tag{30}$$

Thus, we get that the object described by the systems (21) and (22) is controlled from the set $M_0 = (1, 1, 2)$ of space $X = R^3$ to the set $M_1 = (0, -1, 1)$ of space $Y = R^3$ on the time interval $[0, 2]$. The trajectories along which the transition is happening have the form

$$\begin{cases} x_1(t) = -15t^5 + 40t^4 - 31t^3 + 3t^2 + 2t + 1, \\ x_2(t) = -75t^4 + 160t^3 - 93t^2 + 6t + 2 - (75t^4 + 160t^3 - 93t^2 + 6t + 2)^4, \\ x_3(t) = -300t^3 + 480t^2 - 186t + 6 - 4(75t^4 + 160t^3 - 93t^2 + 6t + 2)^7, \quad t \in [0, 1] \end{cases}$$

$$\begin{cases} y_1(t) = 3,5t^5 - 25,5t^4 + 71,5t^3 - 96,5t^2 + 63t - 16, \\ y_2(t) = 17,5t^4 - 102t^3 + 214,5t^2 - 193t + 63 \\ \qquad\qquad -2(3,5t^5 - 25,5t^4 + 71,5t^3 - 96,5t^2 + 63t - 16)^2, \\ y_3(t) = 70t^3 - 306t^2 + 429t - 193 \\ \qquad\qquad -8(3,5t^5 - 25,5t^4 + 71,5t^3 - 96,5t^2 + 63t - 16)^3, \quad t \in [1, 2]. \end{cases}$$

References

1. Velichenko, V.V.: On optimal control problems for equations with discontinuous right sides. Autom. Telemech. **7**, 20–30 (1966)
2. Zemlyakov, A.S.: Synthesis of nonlinear multidimensional systems of control by decomposition and aggregation. KSTU A.N. Tupolev Reports (4), 40–48 (2002)
3. Samburskiy, G.A., Lukyanov, O.V., Khrapov, I.V.: Approach to the construction of hybrid forecasting systems. Reports TGTU. Trans. TSTU **17**(4), 932–935 (2011)
4. Boltaynskiy, V.G.: Optimization problem with phase space change. Differential Equations **XIX**(3), 518–521 (1983)
5. Ashepkov, L.T.: Optimal system control with intermediate conditions. Appl. Math. Mech. **45**(2), 215–222 (1981)
6. Medvedev, V.A., Rozova, V.N.: Optimal control of step systems. Autom. Telemech. **33**(3), 27–32 (1972)
7. Rozova, V.N.: Optimal control of step systems. RUDN Reports. Series: Natural and Technical Sciences, no. 3, pp. 15–23 (2006)
8. Bargsegayn, V. R.: Constructive approach to the study of control problems of linear composite systems. Control Prob. **4**, 11–17 (2012)
9. Maksimova, I.S., Rozova, V.N.: The sufficient conditions for controllability in the problem with phase space change. Tambov University Reports, Series: Natural and Technical Sciences, vol. 3, pp. 742–747 (2011)
10. Maksimova, I.S., Rozova, V.N.: Local controllability in the problem with phase space change. RUDN Reports. Series: Natural and Technical Sciences, vol. 25, no. 4, pp. 331–338 (2017)
11. Maksimova, I.S.: Controllability of nonlinear systems with phase space change. Tavrichesky Vestnik Inf. Math. **2**(51), 53–64 (2021)
12. Korobov, V.I.: Controllability, stability of non-linear systems. Differential Equations **IX**(4), 614–619 (1973)
13. Korobov, V.I.: Method of the Controllability Function. M.Izhevsk. Institute for Computer Research, p. 576 (2007)
14. Kalman, R.E., Ho, Y.C., Narendza, K.S.: Controllability of linear dynamical systems. Contrib. Differential Equations **1**(2), 189–213 (1962)
15. Kalman, R.E.: Mathematical description of linear dynamical systems. SIAM J. Control **1**, 152–192 (1963)
16. Korobov, V.I., Sclayr, G.M.: Methods of constructing positional controls and the admissible maximum principle. Differential Equations **26**(11), 1914–1924 (1990)

A Parallel Linear Active Set Method

E. Dov Neimand and Şerban Sabău

1 Introduction

For years, interior point methods have dominated the field of linear constrained convex minimization [16, 20]. These methods, though powerful, often exhibit three downsides. First, many interior point methods do not lend themselves to parallel implementations without imposing additional criteria. Second, they often require the feasible space be nonempty, [12], or even require a starting feasible point, and when one is unavailable fall back on a second optimization problem, Phase I Method [5]. Third, they typically terminate when they are within an $\epsilon > 0$ distance of the true optimal point, rendering their complexity a function of their accuracy [5, 10].

Here we introduce a linear-inequality-constrained convex minimization method that alleviates these drawbacks. Our method can offer superior performance to state-of-the-art methods when the number of processors is polynomial as a function of the number of constraints in Euclidean space. When this is not the case, though computationally more complex, our method's simple implementation, non-asymptotic convergence, and broad applicability offer considerable value.

Minimization of convex objective functions over non-convex polyhedra struggles to balance slower accurate methods, those with global solutions, against heuristic algorithms that offer a local optimum or pseudo optimal points that may or may not

The work of the authors is based upon research supported by the National Science Foundation under the CAREER Award Number # 1653756.

E. D. Neimand (✉) · Ş. Sabău
Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ, USA
e-mail: eneimand@stevens.edu; ssabau@stevens.edu

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_16

be in the feasible space, Diamond et al. [7]. We present a second algorithm, modified from the first that optimizes over non-convex polyhedra. The method does not compromise on accuracy and has similar complexity to the convex method. Our non-convex method takes advantage of information about the non-convex polyhedron's faces for improved performance over the convex algorithm.

For a simple brute force approach to the three problems facing standard interior point methods, [19] presents a progenitor to Algorithm 1, in finding the projection, $\Pi_P(y)$, of a point, $y \in \mathbb{R}^n$, onto a convex polyhedron, $P \subset \mathbb{R}^n$. Their algorithm first checks if $y \in P$, and if it is not, considers each subset of P's defining inequality constraints, as equality constraints. Projections onto these sets of equality constraints are easily found. A filter removes the affine projections that are outside P, and of those that remain, the closest to y is $\Pi_P(y)$.

In expanding from polyhedral projections in $\mathbb{R}^n$ to a generic convex objective function in a Hilbert space, our algorithm makes use of a black-box linear-equality constrained convex minimization method for our objective function $f : \mathbb{H} \to \mathbb{R}$. Textbooks and papers on unconstrained minimization in Hilbert spaces are now ubiquitous, [3, 4, 6] provide examples. Recently [11] and [14] presented unconstrained minimization methods atop the plethora of preceding research. Given a set of linear-equality constraints, Boyd et al. [5], suggests eliminating the linear equality constraints with a change in variable, reducing the problem to unconstrained minimization in fewer dimensions. Reliance on our black-box method is well-founded.

Unconstrained convex functions can often be optimized quickly. Some functions, like projection functions can be optimized in $O(n^3)$ operations over an affine space in $\mathbb{R}^n$, Plesnik [15]. Note that there is no $\epsilon > 0$ term in the complexity.

Our algorithm employs a test that, together with the black box method, reviews a set of linear inequality constraints, L. The test passes L only if the black-box method can generate the constrained optimal point by treating L's elements as equality constraints. Necessary criteria often allow for the test to fast fail L without using the black-box method, instead looking back at previous applications of the test on subsets of L that have one less inequality than L. This fast fail, as a function of the number of dimensions, has quadratic sequential complexity, and can be completely multi-threaded down to near constant complexity. When the test is unable to fast fail, it resorts to calling the black-box method on the inequality turned equality constraints in L. In both cases the test generates the optimal point of f over L.

Iterative and largely parallel application of the test over growing sets of inequality constraints yields Algorithm 1, which returns $\arg\min_P f$. Algorithm 1 does not employ the test for sets larger than $\min(r, n)$, where r is the total number of constraints and $n \in \mathbb{N} \cup \{\infty\}$ the dimension of the Hilbert space $\mathbb{H}$. Unlike [19], which continues to project onto all the affine spaces after computing and in order to confirm $\Pi_P(y)$, Algorithm 1 ceases its search as soon as the black-box method computes the optimal points.

Our algorithm does not utilize an iterative minimization sequence and therefor preserves valuable properties of the underlying unconstrained minimization method.

When $\arg\min_{\mathbb{H}} f$ finds an exact answer without the need for an iteration arriving within an ϵ distance of the optimal point, so too does our algorithm.

Because of the finite number of operations required to compute the projection onto an arbitrary affine space, our methods excel as a projection function. Recently, Rutkowski, [17], made progress with non-asymptotic parallel projections in a Hilbert space. Where the number of inequality constraints is r, we figure the complexity of their algorithm to be $O(2^{r-1}r^3)$ before parallelization, and $O(r^3)$ over 2^{r-1} processors. Our method compares favorably with theirs as a function of the number of constraints.

Contributions of the Paper: Our methods have distributed complexity. We eliminate common assumptions like the needs for nonempty feasible spaces, a starting feasible point, and a nonempty interior. We develop polyhedral properties to construct easy-to-check, necessary conditions that allow for skipping many of the affine spaces that slow down their forebears. All these reasons will likely lead to the common usage of our convex algorithm on systems capable of large scale multi threading and our non-convex algorithm when even a small amount of multi threading is available and an accurate result is required.

For a quick peek at our algorithm's complexity, let our objective function $f : \mathbb{H} \to \mathbb{R}$, where $\mathbb{H}$ is an $n \in \mathbb{N} \cup \{\infty\}$ dimensional Hilbert space with the standard inner product, $\langle \cdot, \cdot \rangle$, be the projection function, $O(n^3)$, and have $r \in \mathbb{N}$ inequality constraints. If $r >> n$, the complexity comes out to $O(r^{n+1}n^4)$. This complexity result is weaker than the polynomial time of interior point methods reviewed by Polik et al. [16], however when a large number of threads are available to process the problem in parallel, the time complexity of the algorithm is $O(n^4)$, constant as a function of the number of inequalities.

In Sect. 2, we introduce definitions necessary for reading the algorithm. In Sect. 3, we present the algorithm. In Sect. 4, we state and prove the algorithm's foundation. In Sect. 5, we prove that the algorithm works and find its complexity. In Sect. 6, we expand our work to minimization over non-convex polyhedra and present Algorithm 2, the adaptation of Algorithm 1 for non-convex polyhedra.

2 Some Definitions

We present a handful of prerequisite definitions before proceeding to our algorithm.

Definition 2.1 Let P be a **convex polyhedron** and $\mathcal{H}_P$ a finite collection of $r \in \mathbb{N}$ closed half-spaces in $\mathbb{H}$, an $n \in \mathbb{N} \cup \{\infty\}$ dimensional Hilbert space. This lets $P = \bigcap \mathcal{H}_P$, the intersection of the r half spaces in $\mathcal{H}_P$. For all $H \in \mathcal{H}_P$ we define the boundary hyperplane ∂H, the vector $\mathbf{n}_H \in \mathbb{H}$ normal to ∂H, and $b_H \in \mathbb{R}$ such that $H = \{\mathbf{x} \in \mathbb{H} | \langle \mathbf{x}, \mathbf{n}_H \rangle \leq b_H\}$. For any $H \in \mathcal{H}_P$ we say that H is a half-space of P and ∂H a hyperplane of P.

We use the term polyhedron to refer to convex polyhedra. For the non-convex polyhedra we address in Sect. 6, we state their non convexity explicitly.

Example 2.2 Examples of polyhedra include $\mathbb{H}$, $\varnothing$, $\{42\}$, a rectangle, and a set we'll call the 'A' polyhedron, a simple unbounded example we will use to illustrate more complex ideas later on. 'A'$:= \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^2 | \mathbf{y} \leq \frac{1}{2}$ and $\mathbf{x} + \mathbf{y} \leq 1$ and $-\mathbf{x} + \mathbf{y} \leq 1\}$. We have $\mathcal{H}_{\text{'A'}} = \{\bar{F}, \grave{G}, \acute{H}\}$ with $\bar{F} := \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^2 | \mathbf{y} \leq \frac{1}{2}\}$, $\grave{G} := \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^2 | \mathbf{x} + \mathbf{y} \leq 1\}$, and $\acute{H} := \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^2 | -\mathbf{x} + \mathbf{y} \leq 1\}$. Both the name of the 'A' polyhedron and the half-space accents were selected for their iconicity to avoid confusion when we come back to this example.

For a convex objective function, $f : \mathbb{H} \to \mathbb{R}$, constrained to a polyhedron, P, the minimization algorithm below determines if P is empty, or finds the set $\arg\min_P f$. Throughout the paper we will use $f : \mathbb{H} \to \mathbb{R}$ for an arbitrary convex objective function constrained by an arbitrary polyhedron, P.

Example 2.3 Given some $y \in \mathbb{H}$, let $f(\mathbf{x}) = \|\mathbf{x} - \mathbf{y}\|$. We consider the projection problem $\Pi_P(\mathbf{y}) := \arg\min_P f$. Here, f is strictly convex and the optimal set $\arg\min_P f$ will always have a unique value, Boyd et al. [5].

Definition 2.4 We say A is an **affine space of** P if it is a nonempty intersection of a subset of P's hyperplanes. We will denote the set of P**'s affine spaces** with $\mathcal{A}_P := \{\bigcap_{H \in \eta} \partial H | \eta \subseteq \mathcal{H}_P\} \setminus \{\varnothing\}$. Note that $\mathcal{A}_P$ has at most $\sum_{i=1}^{n} \binom{r}{i} \leq \min(r^n, 2^r)$ elements since the intersection of more than n distinct hyperplanes will be an empty set, or redundant with an intersection of fewer hyperplanes.

Example 2.5 If $\mathcal{H}_P = \{F, G, H\}$ then $\mathcal{A}_P = \{\mathbb{H}, \partial H, \partial G, \partial F, \partial H \cap \partial G, \partial H \cap \partial F, \partial F \cap \partial G, \partial H \cap \partial G \cap \partial F\}$. If $P \subset \mathbb{R}^3$, ∂H might be a plane, $\partial H \cap \partial G$ a line, and $\partial H \cap \partial G \cap \partial F$ a single point. However, if any of those intersections are empty then they are not included in $\mathcal{A}_P$. We have $\mathbb{H} \in \mathcal{A}_P$ since if we choose $\eta = \varnothing$ then for all $\mathbf{x} \in \mathbb{H}$ we trivially have $\mathbf{x} \in H$ for all $H \in \eta$, therefor $\mathbf{x} \in \bigcap_{H \in \varnothing} H = \mathbb{H}$.

Example 2.6 Consider the 'A' polyhedron from Example 2.2. It's worth noting that 'A' has an affine space, in this case the point $\partial \grave{G} \cap \partial \acute{H}$, that is disjoint with 'A'. The affine space that is a point at the top of the 'A' is outside of our polyhedron, but still a member of $\mathcal{A}_{\text{'A'}}$. This is a common occurrence.

Definition 2.7 For $A \in \mathcal{A}_P$, we define the P**-cone** of A as $P_A := \bigcap\{H \in \mathcal{H}_P | \partial H \supseteq A\}$, the polyhedron whose hyperplanes, a subset of the hyperplanes of P, intersect to equal A. These are subsets of f's linear-inequality constraints.

Example 2.8 We have $\mathbb{H} \in \mathcal{A}_P$, so it is appropriate to note that for a polyhedron, P we have $P_{\mathbb{H}} = \mathbb{H}$. We comment on this here since in the algorithm presented below we will consider the P-cone for every $A \in \mathcal{A}_P$.

Example 2.9 If we use the 'A' polyhedron (2.2), then the 'A'-cone of the top point $\text{'A'}_{\partial \acute{H} \cap \partial \grave{G}} = \acute{H} \cap \grave{G}$. Note that $\bar{F} \cap \grave{G} \cap \acute{H} = \text{'A'} \subset \text{'A'}_{\partial \bar{F} \cap \partial \grave{G}}$.

Definition 2.10 For $A, B \in \mathcal{A}_P$, we say that B is an **immediate superspace** of A if $B \supsetneq A$ and there exists an $H \in \mathcal{H}_P$ such that $A = \partial H \cap B$. We will also say that A is an **immediate subspace** of B. We will denote the set of all of A's superspaces with $\mathcal{B}_A$.

Algorithm 1: Finds $\arg\min_P f$

```
   Input: A set of half-spaces $\mathcal{H}_P$ and a function $f : \mathbb{H} \xrightarrow{conv.} \mathbb{R}$
   Output: $\arg\min_P f$
1  for $i \leftarrow 0$ to $\min(n, r)$ do
2      for $A \in \mathcal{A}_P$ with $\mathrm{codim}(A) = i$ in parallel do
3          if $\exists B \in \mathcal{B}_A$ s.t. $m_B \cap (P_A \setminus A) \neq \varnothing$ then
4              $m_A \leftarrow m_B \cap P_A$
5          else
6              $m_A \leftarrow \arg\min_A f$ is computed and saved.
7              if $m_A \cap P \neq \varnothing$ then
8                  return $m_A \cap P$
9  return $\arg\min_P f$ is empty.
```

Example 2.11 In the 'A' example (2.2). The immediate superspaces of $\partial \grave{G} \cap \partial \acute{H}$ are $\partial \grave{G}$ and $\partial \acute{H}$. The immediate superspace of $\partial \bar{F}$ is $\mathbb{R}^2$. Observe that if an arbitrary A has co-dimension i, then its immediate superspaces have co-dimensions $i - 1$.

3 The Optimization Algorithm

Algorithm 1 uses the test presented in the *if else* statement on Line 3 to find the optimal point of f in P by iterating over all the affine spaces of P until an affine space $A \in \mathcal{A}_P$ that has nonempty $\arg\min_A f \cap P$ is found, and then returns the optimal point courtesy of the black-box method. In Theorem 5.3 below, we guarantee that the algorithm returns $\arg\min_P f$.

In the Algorithm 1, for some $A \in \mathcal{A}_P$ we use m_A as a place to store $\arg\min_{P_A} f$, previously computed with a call to the black-box method.

In the introduction we described the use of a test to determine if an affine space $A \in \mathcal{A}_P$ is the active set of constraints. What we really want to know is, does $\min_A f = \min_P f$? For that matter, does such an A even exist? And if it does, how will the test recognize it?

We prove our results regarding the answers to these questions in Sects. 4 and 5, but we'll work through a couple of examples for finding that A now. Yes, such an A does exist, and when we refer to the test that recognizes that A, we're referring to lines 3 and 7. The purpose of these examples is to aid in an intuitive understanding of the algorithm.

Example 3.1 Consider a polyhedron, $P \subseteq \mathbb{R}^3$, with a typical vertex, A, to which we will apply the test, optimizing some strictly convex function, f.

When we say that A is a typical vertex, we mean that it's the intersection of three planes. That lets us build P_A, a polyhedral cone, as the intersection of the three plane's half spaces.

The test first looks at all the immediate superspaces of A. We find each of these by removing one of the three planes. Each of A's three immediate superspace is the intersection of two planes. These lines are the edges of the cone that is P_A, and they intersect at A. We'll call these lines B, C and D. Each one has its own P-cone, P_B, P_C and P_D. These cones are all the intersections of two of P_A's three half spaces.

By the time we arrive at the test for A, the algorithm has already computed the optimal points for each of the cones, P_B, P_C and P_D. Those optimal points were stored respectively as m_B, m_C and m_D. Still on Line 3, the test checks if any of those points are in P_A. If so, then A is not the active constraint set. This is the fast fail since we don't need to compute $\arg\min_A f$. Suppose, without loss of generality, the test found that $m_C \in P_A$. A nice result of the fast fail is that we now know that m_C is the optimal point of P_A. That is, $m_A \leftarrow m_C$, which, if there were more dimensions, would be useful later on.

If all m_B, m_C and m_D are outside of P_A, then we progress to the **else** statement now knowing that $\min_{P_A} f = \min_A f$. And that's where the black-box method comes in, because it can compute $\arg\min_A f$. We save that computation as m_A for future use.

There's one last thing to do. We've verified that $m_B, m_C, m_D \in {P_A}^c$, and computed m_A. If $m_A \in P$, then m_A is the optimal point over P and the algorithm concludes. If it's not, we move on to apply the test to some other affine space of P.

By checking the affine spaces in order of co-dimension, we ensure that we've already done the work on immediate superspaces to set the test up for success.

There are lots of *why* questions to be asked about Example 3.1. Sections 4 and 5 should answer those questions. You can find a complete and detailed run through of Algorithm 1 in Example 3.2.

Example 3.2 We will revisit Example 2.2 by walking the problem $\Pi_{\text{`A'}}(1, 1)$ through Algorithm 1. Refer to Fig. 1 throughout this example for your convenience.

We begin Line 1 with $i \leftarrow 0$, setting us up to consider on Line 2 all the affine spaces in $\mathcal{A}_P$ with co-dimension 0. The only such affine space is $\mathbb{H}$, so $A \leftarrow \mathbb{H}$. On Line 3, we note that $\mathbb{H}$ has no immediate superspaces, so $\mathcal{B}_{\mathbb{H}} = \varnothing$, and the condition in the **if**, statement is false. We proceed to the **else** statement and compute $m_{\mathbb{H}} \leftarrow \Pi_{\mathbb{H}}(1, 1) = (1, 1)$. We now check the condition on Line 7 and find $m_{\mathbb{H}}$ as $(1, 1)$ is not in P. The condition is false. The inner loop completes an iteration, and with no more affine spaces of co-dimension 0, the inner loop concludes. The outer loop on Line 1 progresses to $i \leftarrow 1$, to look at all of P's affine spaces of co-dimension 1 on Line 2.

There are three affine spaces of co-dimension 1, $\partial \acute{H}$, $\partial \grave{G}$, and $\partial \bar{F}$. Each affine space of co-dimension 1 has the same set of immediate superspaces, $\mathcal{B}_{\partial \acute{H}} = \mathcal{B}_{\partial \grave{G}} = \mathcal{B}_{\partial \bar{F}} = \{\mathbb{H}\}$.

On Line 2, we will arbitrarily look at $A \leftarrow \partial \acute{H}$ first, though ideally all three affine spaces would be considered in parallel. On Line 3, we review every $B \in \mathcal{B}_{\partial \acute{H}} = \{\mathbb{H}\}$ to check if $m_B \in P_{\partial \acute{H}} \setminus \partial \acute{H}$. There's just the one, $m_{\mathbb{H}} = (1, 1)$, so the check is easy.

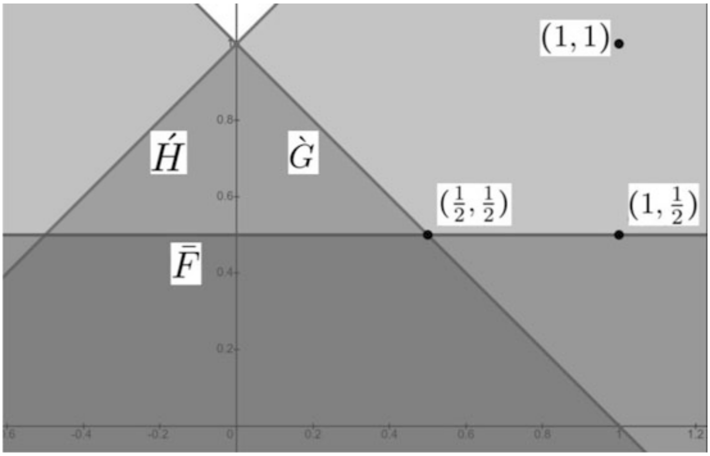


Fig. 1 Example 3.2

Is $(1, 1) \in P_{\partial \acute{H}} \setminus \partial \acute{H}$? We have $P_{\partial \acute{H}} = H$. Yes, $-1 + 1 < 1$. The condition on Line 3 is true. We proceed to Line 4 and assign $m_{\partial \acute{H}} \leftarrow (1, 1)$. Completing the inner loop iteration for $\acute{H}$, we move onto $A \leftarrow \partial \grave{G}$ and $A \leftarrow \partial \bar{F}$.

For both $A \leftarrow \partial \grave{G}$ and $A \leftarrow \partial \bar{F}$, on Line 3 we have m_B as $(1, 1)$. We check the condition on Line 3. Is m_B as $(1, 1)$ in $\bar{F} \setminus \partial \bar{F}$? Is it in $\grave{G} \setminus \partial \grave{G}$? No. Both A as $\partial \bar{F}$ and $\partial \grave{G}$ go to the **else** statement where we compute $m_{\partial \bar{F}} = \Pi_{\partial \bar{F}}(1, 1) = (1, \frac{1}{2})$ and $m_{\partial \grave{G}} = \Pi_{\partial \grave{G}}(1, 1) = (\frac{1}{2}, \frac{1}{2})$. However, on line 7, different things happen to them. We check $m_{\grave{G}}$ and $m_{\bar{F}}$ for membership in P on Line 7. The point $(1, \frac{1}{2}) \in P^c$, but the point $(\frac{1}{2}, \frac{1}{2}) \in P$, taking A as $\partial \grave{G}$ to the **return** statement on Line 8. We conclude $\Pi_{\text{‘A’}}(1, 1) = (\frac{1}{2}, \frac{1}{2})$.

Note that if both conditions on Line 7 had turned out false, we now know $m_{\bar{F}}, m_{\acute{H}}$, and $m_{\grave{G}}$, preparing us for the next iteration of the outer loop where we consider affine spaces of co-dimension $i \leftarrow 2$.

Remark 3.3 Below, in Theorem 5.11 we present the complexity of Algorithm 1. If the Hilbert space is finite dimensional, uses the standard inner product, $r >> n$, and the black-box method takes $M(n)$ operations, then the complexity of the algorithm is $O(r^n \cdot (r \cdot n + M(n)))$ when run sequentially, and $O(n(n + M(n)))$ when run in parallel.

4 Polyhedral Proofs

In this section we present novel necessary and sufficient conditions for an affine-space A to have $\min_A f = \min_P f$ and guarantee A's existence for the case when

$\arg\min_P f \neq \varnothing$. While The Sufficient Criteria (4.14) require the computation $\min_A f$, The Necessary Criteria (4.10) do not. This significantly reduces the number of affine spaces over which we call the black-box method to calculate $\arg\min_A f$.

4.1 Preliminary Proofs

Definition 4.1 For $\mathbf{a}, \mathbf{b} \in \mathbb{H}$, we use $\overset{\vdash\!\!\dashv}{\mathbf{a}, \mathbf{b}}$ to denote the closed **line segment** from $\mathbf{a}$ to $\mathbf{b}$ and $\overline{\mathbf{a}, \mathbf{b}}$ to denote the **line** containing $\mathbf{a}$ and $\mathbf{b}$.

We include Lemma 4.2 and 4.3 for the reader's convenience. They are proved in Neimand et al. [13].

Lemma 4.2 *Let $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{H}$. If H is a half-space such that $\boldsymbol{a} \in H$ and $\boldsymbol{b} \in H^c$, then $\partial H \cap \overset{\vdash\!\!\dashv}{\boldsymbol{a}, \boldsymbol{b}}$ has exactly one point.*

Lemma 4.3 *Let $\boldsymbol{a}, \boldsymbol{b}$, and $\boldsymbol{c}$ be distinct points in $\mathbb{H}$ with $\boldsymbol{b} \in \overset{\vdash\!\!\dashv}{\boldsymbol{a}, \boldsymbol{c}}$.*

1. $\|\boldsymbol{a} - \boldsymbol{b}\| + \|\boldsymbol{b} - \boldsymbol{c}\| = \|\boldsymbol{a} - \boldsymbol{c}\|$
2. $\|\boldsymbol{a} - \boldsymbol{b}\| < \|\boldsymbol{a} - \boldsymbol{c}\|$.
3. If $f : \mathbb{H} \to \mathbb{R}$ is convex and $f(\boldsymbol{a}) < f(\boldsymbol{c})$ then $f(\boldsymbol{b}) < f(\boldsymbol{c})$.
4. If $f : \mathbb{H} \to \mathbb{R}$ is convex and $f(\boldsymbol{a}) \leq f(\boldsymbol{c})$ then $f(\boldsymbol{b}) \leq f(\boldsymbol{c})$.

Definition 4.4 We use the following notations. For any $X \subset \mathbb{H}$ we use $\text{aff}(X)$ to denote the **affine hull** of X, $B_r(\mathbf{y})$ to denote the open **ball** centered at $\mathbf{y} \in \mathbb{H}$ with a radius of $r \in \mathbb{R}$, $\text{int}(X)$ for the **interior** of X, and $\text{relint}\, X$ to denote the **relative interior** of X.

Lemma 4.5 *Let $K \subseteq P$ be a nonempty convex set and A be the smallest space with regards to inclusion in $\mathcal{A}_P$ such that $K \subseteq A$, and let $y \in \text{relint}\, K$, then for some $H \in \mathcal{H}_P$ (Def. 2.1), if $y \in \partial H$ then $A \subseteq \partial H$.*

Proof Let $H \in \mathcal{H}_P$ such that $\mathbf{y} \in \partial H \cap \text{relint}\, K$. There exists an $\epsilon > 0$ and $N := B_\epsilon(\mathbf{y}) \cap \text{aff}(K)$, such that $N \subseteq K \subseteq P \cap A$.

Let us falsely assume A is not a subset of ∂H. If $K \subseteq \partial H$, then by the definition of A, $A \subseteq \partial H$ in contradiction to the false assumption we just made. Therefor, K is not a subset of ∂H and there exists an $\mathbf{a} \in K \setminus \partial H$. Since $K \subseteq P$ it follows that $\mathbf{a} \in \text{int}(H)$.

Let $t_\epsilon := 1 + \frac{\epsilon}{2\|\mathbf{a}-\mathbf{y}\|} \in \mathbb{R}$ and $\mathbf{y}_\epsilon := (1 - t_\epsilon)\mathbf{a} + t_\epsilon \mathbf{y}$. Observe that $\|\mathbf{y}_\epsilon - \mathbf{y}\| = \|(1-t_\epsilon)\mathbf{a} + t_\epsilon \mathbf{y} - \mathbf{y}\| = \frac{\epsilon}{2\|\mathbf{a}-\mathbf{y}\|}\|\mathbf{a} - \mathbf{y}\| = \frac{\epsilon}{2}$, giving $\mathbf{y}_\epsilon \in B_\epsilon(\mathbf{y}) \cap \overline{\mathbf{a}, \mathbf{y}}$. Note that any line containing two points in an affine space is entirely in that affine space; since $\mathbf{a}, \mathbf{y} \in \text{aff}\, K$, we have $\overline{\mathbf{a}, \mathbf{y}} \subseteq \text{aff}\, K$. Since $\mathbf{y}_\epsilon \in \overline{\mathbf{a}, \mathbf{y}}$, we have $\mathbf{y}_\epsilon \in \text{aff}\, K$, and we may conclude $\mathbf{y}_\epsilon \in N$.

Let $t_y := (\|\mathbf{a} - \mathbf{y}\| + 2^{-1}\epsilon)^{-1}\|\mathbf{a} - \mathbf{y}\|$. From our earlier definition of $\mathbf{y}_\epsilon$, we have $\mathbf{y}_\epsilon = (-2^{-1}\|\mathbf{a} - \mathbf{y}\|^{-1}\epsilon)\mathbf{a} + 2^{-1}\|\mathbf{a} - \mathbf{y}\|^{-1}(2\|\mathbf{a} - \mathbf{y}\| + \epsilon)\mathbf{y}$. By isolating $\mathbf{y}$ and substituting in t_y, we get $\mathbf{y} = (1 - t_y)\mathbf{a} + t_y \mathbf{y}_\epsilon$, giving $\mathbf{y} \in \overset{\vdash\!\!\dashv}{\mathbf{a}, \mathbf{y}_\epsilon}$.

If $\mathbf{y}_\epsilon$ is in $\operatorname{int}(H)$, then by convexity of $\operatorname{int}(H)$, we have $\overline{\mathbf{a}, \mathbf{y}_\epsilon} \subset \operatorname{int}(H)$, including $\mathbf{y}$, a contradiction to $\mathbf{y} \in \partial H$.

If $\mathbf{y}_\epsilon$ is in ∂H, we have two points of $\overline{\mathbf{a}, \mathbf{y}}$ in ∂H. It follows that $\overline{\mathbf{a}, \mathbf{y}} \subseteq \partial H$ and $\mathbf{a} \in \partial H$, a contradiction.

All that remains is for $\mathbf{y}_\epsilon \in H^c \subseteq P^c$. But $\mathbf{y}_\epsilon \in N$ and $N \subseteq P$, a contradiction. □

Proposition 4.6 *Let $K \subseteq P$ be a nonempty convex set and A be the smallest space with regards to inclusion in $\mathcal{A}_P$ such that $K \subseteq A$. Then for any $\boldsymbol{x} \in \operatorname{relint} K$ there exists an $\epsilon > 0$ such that $P_A \cap B_\epsilon(\boldsymbol{x}) = P \cap B_\epsilon(\boldsymbol{x})$.*

Proof We may assume that $\mathcal{H}_P$ is nonempty and that $A \neq \mathbb{H}$, otherwise the proof is trivial.

Let $x \in \operatorname{relint} K$. Let $Q \subseteq \mathbb{H}$ be a polyhedron such that $\mathcal{H}_Q = \mathcal{H}_P \setminus \mathcal{H}_{P_A}$. Then we can define $\epsilon := \min_{y \in \partial Q} \|\mathbf{y} - \mathbf{x}\|$. If we falsely assume $\epsilon = 0$, then there exists an $H \in \mathcal{H}_Q$ with $\mathbf{x} \in \partial H \cap P$. Since $\mathbf{x} \in \operatorname{relint} K$, we may conclude from Lemma 4.5 that $A \subset \partial H$ and that $H \in \mathcal{H}_{P_A}$, a contradiction. We may conclude $\epsilon > 0$.

($\subseteq$) Let $\mathbf{y} \in B_\epsilon(\mathbf{x}) \cap P_A$. Let's falsely assume $\mathbf{y} \in P^c$. There exists an $H \in \mathcal{H}_P$ such that $\mathbf{y} \in H^c$. We have $\mathcal{H}_P = \mathcal{H}_Q \,\dot\cup\, \mathcal{H}_{P_A}$. Since $\mathbf{y} \in P_A$ it follows that $H \in \mathcal{H}_Q$. Since $\mathbf{x} \in P \subseteq H$, by Lemma 4.2 we may consider the unique $\partial H \cap \overline{x, y}$, and from Lemma 4.3 conclude that $\|\partial H \cap \overline{\mathbf{x}, \mathbf{y}} - \mathbf{x}\| < \|\mathbf{x} - \mathbf{y}\| < \epsilon$, a contradiction to our choice of epsilon. We may conclude that $P_A \cap B_\epsilon(\mathbf{x}) \subseteq P \cap B_\epsilon(\mathbf{x})$.

($\supseteq$) With $P \subseteq P_A$, it follows that $P_A \cap B_\epsilon(\mathbf{x}) \supseteq P \cap B_\epsilon(\mathbf{x})$. □

Lemma 4.7 *For any convex $K \subset \mathbb{H}$, the set $\arg\min_K f$ is convex.*

Lemma 4.7 is proved in Niemand et al. [13].

4.2 The Necessary Criteria

Definition 4.8 If $\arg\min_P f \neq \varnothing$, we define the **min space** of f on P as the smallest $A \in \mathcal{A}_P$ with regards to inclusion that has $\arg\min_P f \subseteq A$. Equivalently, the min space is the intersection of all the hyperplanes of P that contain $\arg\min_P f$. Where f and P are implied, we omit them.

Remark 4.9 If $\arg\min_P f \neq \varnothing$, then the min space exists and is unique. If there are no hyperplanes of P that contain $\arg\min_P f$, giving $\arg\min_P f \subseteq \arg\min_{\mathbb{H}} f$, then the min space is $\mathbb{H}$.

Theorem 4.10 (The Necessary Criteria) *Let A be the min space for some f on P, then A meets the Necessary Criteria which are as follows:*

1. $\arg\min_P f \subseteq \arg\min_A f$
2. $\arg\min_A f = \arg\min_{P_A} f$

Proof (4.10.1) From Definition 4.8, we have $\min_A f \leq \min_P f$.

Let's falsely assume there exists a $\mathbf{a} \in A$ such that $f(\mathbf{a}) < \min_P f$ and let $\mathbf{x} \in \operatorname{relint} \arg\min_P f$.

By Proposition 4.6, there exists an $\epsilon > 0$ such that $B_\epsilon(\mathbf{x}) \cap P = B_\epsilon(\mathbf{x}) \cap P_A$. The line segment $\overline{\mathbf{a}, \mathbf{x}}$ is entirely in $A \subset P_A$, so we may choose $t_y := 1 - \frac{\epsilon}{2\|\mathbf{a}-\mathbf{x}\|} \in (0, 1)$ so that $\mathbf{y} := (1 - t_y)\mathbf{a} + t_y\mathbf{x} \in \overline{\mathbf{a}, \mathbf{x}} \cap B_\epsilon(\mathbf{x}) \cap P_A$. Since $\mathbf{y} \in \overline{\mathbf{a}, \mathbf{x}}$, by Lemma 4.3.3 we have $f(\mathbf{y}) < f(\mathbf{x}) = \min_P f$. Proposition 4.6 gives $\mathbf{y} \in P$, a contradiction. □

Proof (4.10.2) Let's falsely assume that there exists an $\mathbf{x} \in (P_A \setminus A)$ such that $f(\mathbf{x}) \leq \min_P f$, which by Definition 4.8 has $\arg\min_P f \subset A$, and let $\mathbf{y} \in \operatorname{relint} \arg\min_P f$. Then by Proposition 4.6, we can let $\epsilon > 0$ such that $B_\epsilon(\mathbf{y}) \cap P = B_\epsilon(\mathbf{y}) \cap P_A$.

Since $\mathbf{x} \in P_A \setminus A$, it follows from convexity of P_A that $\overline{\mathbf{x}, \mathbf{y}} \setminus \{\mathbf{y}\} \subset P_A \setminus A$. If there was a second point beside $\mathbf{y}$ in A, then by the definition of an affine space, $\mathbf{x}$ would be in A as well.

As in 4.10.1, we may choose a $\mathbf{z} \in \overline{\mathbf{x}, \mathbf{y}} \cap B_\epsilon(\mathbf{y}) \subset P \cap P_A$ with a distance of $\frac{\epsilon}{2}$ from $\mathbf{y}$. We have $\mathbf{z} \in P \setminus A$, and by Lemma 4.3, $f(\mathbf{z}) \leq f(\mathbf{y})$. If $f(\mathbf{z}) = f(\mathbf{y})$, this stands in contradiction to $\arg\min_P f \subseteq A$. If $f(\mathbf{z}) < f(\mathbf{y})$, we have a contradiction to $\mathbf{y} \in \arg\min_P f$.

We may conclude that for all $\mathbf{x} \in P_A \setminus A$, $f(\mathbf{x}) > \min_P f$. From 4.10.1, we see that if $\mathbf{x} \in A$, then $f(\mathbf{x}) \geq \min_P f$. Combining these two and the fact that $\arg\min_P f \subseteq A$, we achieve the desired result. □

4.3 The Sufficient Criteria

For those affine spaces that meet the necessary criteria (4.10), we next consider The Sufficient Criteria

Definition 4.11 Let $A, B \in \mathcal{A}_P$, with $A \subsetneq B$. We can say that B **disqualifies** A from P, with regards to f, if B is the min space of f on P_A. If there is no such B, then we say A is a **candidate** for f on P. Where f and P are implied, they are omitted.

Lemma 4.12 *The min space is a candidate.*

Proof Let $A, B \in \mathcal{A}_P$ such that B disqualifies A. The min space of P_A is B. There exists an $\mathbf{x} \in \arg\min_{P_A} f \setminus A$, otherwise A would be the min space over P_A and not B. But this is a contradiction to The Necessary Criteria (4.10.2). □

Lemma 4.13 *If and only if $A \in \mathcal{A}_P$ is a candidate, then* $\arg\min_A f = \arg\min_{P_A} f$.

Proof Let A be a candidate, and falsely assume $\arg\min_A f \neq \arg\min_{P_A} f$. This means P_A has a min space other than A, and that min space disqualifies A, in contradiction to A being a candidate.

Let $\arg\min_A f = \arg\min_{P_A} f$. Let's falsely assume there exists a B that disqualifies A. That means there exists an $\mathbf{x} \in (P_A \setminus A) \cap \arg\min_{P_B} f$ in contradiction to $\arg\min_{P_A} f = \arg\min_A f$. □

Proposition 4.14 (The Sufficient Criteria) *Let A be a candidate and $\arg\min_A f \cap P \neq \varnothing$. Then $\arg\min_A f \cap P = \arg\min_P f$.*

Proof Let A be a candidate of P with $\arg\min_A f \cap P \neq \varnothing$.

Let $\mathbf{x} \in \arg\min_{P_A} f \cap P$. Since $P \subseteq P_A$, for all $\mathbf{y} \in P$ we have $f(x) \geq y$. But $\mathbf{x} \in P$ so $\mathbf{x} \in \arg\min_P f$. Therefore, (1) $\arg\min_{P_A} f \cap P \subseteq \arg\min_P f$. Let $\mathbf{x} \in \arg\min_P f$. Since $f(\mathbf{x}) = \min_P f = \min_{P_A} f$ and $\mathbf{x} \in P_A$ we have $\mathbf{x} \in \arg\min_{P_A} f$. That is to say, (2) $\arg\min_P f \subseteq \arg\min_{P_A} f$. We also have (3) $\arg\min_P f \subseteq P$. We can combine the three set inequalities to conclude $\arg\min_{P_A} f \cap P = \arg\min_P f$.

To complete the proof we again recall Lemma 4.13, and note $\arg\min_A f \cap P = \arg\min_{P_A} f \cap P = \arg\min_P f$. □

Let $A \in \mathcal{A}_P$. If for all $B \in \mathcal{A}_P$ with $B \supsetneq A$ we know $\arg\min_B f$, we can use disqualification to determine that A is not the min space, without expensively computing $\arg\min_A f$. Furthermore, if B disqualifies A, then we also know $\arg\min_{P_A} f = \arg\min_B f = \arg\min_{P_B} f$ which was previously computed. This is not our fast fail, A may have too many superspaces, but we're getting closer.

Remark 4.15 If A is the intersection of m hyperplanes of P, then it has m immediate superspaces, each can be generated by taking the intersection of $m - 1$ of the hyperplanes that intersect to make A. Note that $m < \min(n, r)$ since A can't be the intersection of more than the total number of hyperplanes, or more hyperplanes than there are dimensions.

Theorem 4.16 *Let $A \in \mathcal{A}_P$, then A is disqualified from P, if and only if there exists an immediate superspace, B, such that $\arg\min_{P_B} f \cap (P_A \setminus A)$ is nonempty.*

Proof ($\Rightarrow$) Let $A \in \mathcal{A}_P$, such that A is disqualified from P.

If A is disqualified by some $B \in \mathcal{B}_A$, then there exists an $\mathbf{x} \in \arg\min_B f \cap P_A \cap A^c$. Since B is a min space for P_A, the Necessary Criteria (4.10) give $\arg\min_B f = \arg\min_{P_B} f$ achieving the desired result.

If A is disqualified by some C that is not an immediate superspace of A, then C is the min space of A, and there exists a $\mathbf{c} \in \arg\min_C f = \arg\min_{P_C} f$ (Theorem 4.10) with $\mathbf{c} \in P_A \setminus A$, such that for all $\mathbf{x} \in P_C$, we have $f(\mathbf{x}) \geq f(\mathbf{c})$. Since $\mathbf{c} \in P_A$ and $P_A \subseteq P_B$ we have $\mathbf{c} \in P_B$. Since $P_B \subset P_C$ we have $\mathbf{c} \in \arg\min_{P_B} f$, the desired result.

($\Leftarrow$) Let B be an immediate superspace of A, and let $\mathbf{b} \in \arg\min_{P_B} f \cap (P_A \setminus A)$. Since $P_A \subsetneq P_B$ and $\mathbf{x} \in \arg\min_{P_B} f$, then $\mathbf{b} \in \arg\min_{P_A} f$. The min space of P_A contains $\mathbf{b}$, which is in A^c, so that space is not A, and therefor disqualifies A. □

Remark 4.17 Let $A \in \mathcal{A}_P$. Proposition 4.16 and its results allow us to determine if A meets the necessary criteria by looking exclusively at A's immediate superspaces, $\mathcal{B}_A$, and their P-cones. For any $B \in \mathcal{B}_A$ there exists an $H_B \in \mathcal{H}_P$ such that

$A = \partial H_B \cap B$; if $\arg\min_{P_B} f \cap (H_B \setminus A)$ is nonempty, then $\arg\min_{P_B} f \cap (P_A \setminus A)$ is nonempty, and A is disqualified. We check if any $B \in \mathcal{B}_A$ disqualifies A by confirming $\langle \arg\min_{P_B} f, n_{H_B} \rangle \leq b_H$ is nonempty. If the complexity of computing the inner product is n and $m := \operatorname{codim} A = |\mathcal{B}_A| \leq n$, then when $\arg\min_{P_B} f$ is known for all $B \in \mathcal{B}_A$, Remark 4.15 and Theorem 4.16 let us check the Necessary Criteria for A in $O(m \cdot n)$. This same check, when the inequality holds, yields $\arg\min_{P_A} f$. This is the fast fail.

We can now detail the test method introduced in Sect. 1. If the fast fail is successful for an affine space A, then we have the optimal points of the disqualifying set that are in P_A as $\arg\min_{P_A} f$; there is no need for any additional computation. If the fast fail is unsuccessful, then A is a candidate and Lemma 4.13 tells us we can use the black-box method to compute $m_A \leftarrow \arg\min_A f$ where $m_A := \arg\min_{P_A} f$. With the test complete and knowledge of $\arg\min_{P_A} f$, we prepare to apply the test to A's immediate sub-spaces.

This result lends itself to Algorithm 1, wherein we begin by finding the optimum over $\mathbb{H}$, then at each iteration find the optimum of all the P-cones of the immediate sub-spaces, until one of those spaces meets the necessary and sufficient criteria.

5 Algorithm Proofs and Analysis

5.1 Proof of Function

Lemma 5.1 *When the if else statement in Algorithm 1 Line 3 accesses m_B for some $B \in \mathcal{B}_A$ that m_B has already been saved to memory.*

Proof We will prove by induction on the affine space's co-dimension. The base A is $\mathbb{H}$, since it is the only affine space of P with co-dimension 0. The Hilbert space has no immediate superspaces, that is $\mathcal{B}_{\mathbb{H}} = \varnothing$, and therefor m_B for some $B \in \mathcal{B}_{\mathbb{H}}$ is never called. For an affine space with co-dimension j, we will assume that all the affine spaces of co-dimension $j-1$ had their requisite input available. We note that every affine space, B of co-dimension $j-1$ was put up for review by Line 2, and generated an m_B on Line 4 or Line 6. The superspaces of A's and the minimums over their P-cones are all available. □

Lemma 5.2 *The if else statement on Line 3 goes to the else statement, if and only if A is a candidate.*

Proof Let's assume conditions in the *if* statement are not met and the *else* statement is reached. This means the *if* statement on Line 3 determined that for every immediate superspace, $B \in \mathcal{B}_A$, we have $m_B \cap (P_A \setminus A) = \varnothing$. Equivalently, $\arg\min_{P_B} f \subset (P_A \setminus A)^c$ which by Corollary 4.16 gives A as a candidate.

Let A be a candidate, then the *if* statement on Line 3 will find that for all $B \in \mathcal{B}_A$ we have $m_B \cap (P_A \setminus A) = \varnothing$, and the *else* statement will be reached. □

Theorem 5.3 *The return set of Algorithm 1 is equal to* $\arg\min_P f$.

Proof By Remark 4.9, if $\arg\min_P f \neq \varnothing$, the min space exists, and by Lemma 4.12 the min space is a candidate. The two *for* loops will iterate over every affine space of P until a candidate is found that meets the sufficient criteria, checked with a true statement on line 7 and a false one on line 3. By Corollary 4.14 the min space meets the Sufficient Criteria (4.14). If $\arg\min_P f$ is nonempty, then a *return* set is guaranteed.

Let A be candidate (see Lemma 5.2) and the *else* statement reached. If Line 7 finds that The Sufficient Criteria (4.14) are met, then the conditions for Proposition 4.14 are satisfied, insuring the algorithm returns $\arg\min_P f$.

If $\arg\min_P f = \varnothing$, then the conditions for The Sufficient Criteria (4.14) are never met and the *if* statement on Line 7 will reject every A. Once all the affine spaces have been reviewed, the final *return* statement is called and an empty set is returned. □

Example 5.4 Referring back to Example 3.2, $\partial\grave{G}$, whose minimum is the minimum for 'A' is not the min space; $\partial\grave{G} \cap \partial\bar{F}$ is. However $\partial\grave{G}$ is a candidate and The Sufficient Criteria are met. What the min space definition gives us is that if a minimum exists, we can find its min space. But our set of candidates that meet the Sufficient Criteria is broader.

Proposition 4.14 insures that, in spite of the algorithm not having found the min space, $\arg\min_P f$ is still returned.

5.2 Complexity

Lemma 5.5 *If f is strictly convex, then for any convex K,* $\arg\min_K f$ *has at most one element.*

We will limit the scope of this complexity analysis to strictly-convex f. This significantly simplifies our work and implementation of the algorithm by insuring that each m_B in Algorithm 1 has a single element. Computing weather $m_B \cap P_A = \varnothing$ then becomes $m_B \in P_A$.

Definition 5.6 For clarity, we use brackets to indicate the computational complexity of a process, as a function of n and possibly some $\epsilon > 0$. Thus $[\langle\cdot,\cdot\rangle]$ is the number of steps it takes to compute inner product, ranging from n to n^3 for finite inner products and likely a function of ϵ for infinite Hilbert spaces. For some affine space A, we have $[\arg\min_A f]$ as the number of steps it takes to compute our black-box method.

Corollary 5.7 *Checking if $m_B \cap P_A \neq \varnothing$ on Line 3 has the same complexity as computing inner product, $O([\langle\cdot,\cdot\rangle])$.*

Proof This is a direct result of Remark 4.17. □

Lemma 5.8 *Checking if $\exists B \in \mathcal{B}_A$ s.t. $m_B \cap (P_A \setminus A) \neq \varnothing$ on Line 3 has $O(\min(n, r) \cdot [\langle\cdot, \cdot\rangle])$ sequential computational complexity and $O([\langle\cdot, \cdot\rangle])$ time complexity if run in parallel over $min(n, r)$ processors.*

Proof By Corollary 5.7, Checking $m_B \cap P_A \neq \varnothing$ has complexity $O([\langle\cdot, \cdot\rangle])$. A loop checks this once for each $B \in \mathcal{B}_A$, with Remark 4.15 giving $|\mathcal{B}_A| \leq \min(n, r)$. Each $B \in \mathcal{B}_A$ can be checked for $m_B \cap (P_A \setminus A) \neq \varnothing$ independently of one another, so they can all be checked in parallel. □

Lemma 5.9 *The if statement on Line 7 is $O(r \cdot [\langle\cdot, \cdot\rangle])$ sequential computational complexity and $O([\langle\cdot, \cdot\rangle])$ when run in parallel over r processors.*

Proof Checking if a point is in P requires checking that the point is in each $H \in \mathcal{H}_P$. Checking if a point is in a half-space is $O([\langle\cdot, \cdot\rangle])$ and since these r checks are independent of one another, they can be done in parallel. □

Lemma 5.10 *Running the entire if else statement that begins on Line 3 has $O(r \cdot [\langle\cdot, \cdot\rangle] + [\arg\min_A f])$ sequential computational complexity, or $O([\langle\cdot, \cdot\rangle] + [\arg\min_A f])$ time complexity if run in parallel over r processors.*

Proof We saw in Lemma 5.8 the *if* statement's complexity. If there is no fast fail, the *else* portion computes $\arg\min_A f$.

The inner *if* statement on Line 7 is $O(r)$, so adding these three components we get $O(\min(n, r) \cdot [\langle\cdot, \cdot\rangle] + [\arg\min_A f] + r \cdot [\langle\cdot, \cdot\rangle])$ computational complexity. In simplifying, note that $\min(n, r) \leq r$.

For the parallel case, we have, $O([\langle\cdot, \cdot\rangle] + [\arg\min_A f] + [\langle\cdot, \cdot\rangle])$, which also simplifies to the desired expression.

The same r threads that are used on Line 3 can be used again on Line 7, so there's no need for more than r processors. □

Theorem 5.11 *Algorithm 1 has $O(\min(r^n, 2^r) \cdot (r \cdot [\langle\cdot, \cdot\rangle] + [\arg\min_A f]))$ sequential computational complexity, and $O(\min(n, r) \cdot ([\langle\cdot, \cdot\rangle] + [\arg\min_A f]))$ time complexity when run in parallel over $O(\min(r^{\frac{1}{2}} \cdot 2^{r+\frac{1}{2}}, r^{n+1}))$ processors.*

Proof For computational complexity we note that the two *for* loops in Algorithm 1 iterate over all the affine spaces in $\mathcal{A}_P$, so we multiply our results from Lemma 5.10 by $|\mathcal{A}_P|$.

For the parallel case, the outer loop cannot be run in parallel. The inner can. The number of iterations for the inner loop, for any $i \leq \min(r, n)$ is $\binom{r}{i}$, because each affine space of co-dimension i is the intersection of i hyperplanes of P. Consequently, with $\max_{i<\min(r,n)} \binom{r}{i}$ processors, the inner loop approaches $O(1)$ parallel time complexity. The number of iterations of the outer loop is $\min(n, r)$.

We note that r is a maximum number of iterations for the outer loop since the co-dimension of an affine space $A \in \mathcal{A}_P$ is the number of hyperplanes that intersect to make A. That number of hyperplanes, and therefore the co-dimension, cannot exceed the number of P's hyperplanes, r. We have n as a maximum because the intersection of more than n hyperplanes will be an empty set or redundant with the intersection of fewer hyperplanes.

All that remains is to compute $\max_{i \leq \min(n,r)} \binom{r}{i}$. If $n > \frac{r}{2}$, Pascal's triangle tells us that we have the maximum at $i = \frac{r}{2}$, the Central Binomial Coefficient. Stirling's formula [18] tells us $\binom{r}{\frac{r}{2}} \sim (\pi r)^{-\frac{1}{2}} 2^{r+\frac{1}{2}}$. If $n < \frac{r}{2}$, then the maximum number of processors for the inner loop becomes $\binom{r}{n} \leq r^n$. This puts the total number of processors for the inner loop at $O(\min(r^{-\frac{1}{2}} \cdot 2^{r+\frac{1}{2}}, r^n))$.

Multiplying by the number of processors we need for the *if else* statement gives us the desired result. □

When $r >> n$ we have polynomial sequential complexity as a function of r, and parallel complexity that's constant using a polynomial number of threads, as function of r. When $n >> r$ then sequential and parallel complexities, as well as the number of processors, as a function of n are the complexity of the black-box method plus the inner product method.

Note that unlike many interior point methods, the complexity is not a function of accuracy; outside of the black-box method, there is no ϵ term that compromises speed with the desired distance from the correct answer.

6 Non-Convex Polyhedra

This section expands the results of the previous section to conclude with a multi-threaded algorithm for computing the global minimum in the case of non-convex polyhedral constraints. Since the algorithm is not dependent on a starting feasible point, we find all the local optimum as they meet the necessary criteria, and the optimal of the points that meet the necessary criteria is the global optimum. Our non-convex constraints algorithm exploits the representation of non-convex polyhedra to achieve faster results than the convex algorithm presented above.

We will work with the description from [8] for non-convex polyhedra, where the polyhedron is represented by its faces, where each face, a convex polyhedron itself, has knowledge of its own faces and its neighbors. Together with the definition of non-convex polyhedra in [9], we define a non-convex polyhedron as follows.

Definition 6.1 A non-convex polyhedron $P \subset \mathbb{R}^n$ is the union of a set of convex polyhedra, $\mathcal{P}$. Namely, $P = \bigcup \mathcal{P}$. We denote the set of faces of P with $\mathcal{F}_P$ and include $P \in \mathcal{F}_P$ as the lone exception to the requirement that P's faces be convex. Note that $\mathcal{F}_P$ is closed to intersections.

Definition 6.2 We can redefine P's affine spaces, $\mathcal{A}_P$ so that $\mathcal{A}_P = \{A | \forall \mathcal{P} \exists Q \in \mathcal{P}, \text{with } A \in \mathcal{A}_Q \text{ and } \exists F \in \mathcal{F}_P \text{ such that } \operatorname{aff} F = A\} \cup \{\mathbb{R}^n\}$.

Lemma 6.3 *If P is convex, then $\mathcal{A}_P$ under Definition 6.2 is a subset of $\mathcal{A}_P$ under Definition 2.4, and that subset includes every affine space that has a non empty intersection with P.*

Proof Let $A \in \mathcal{A}_P$ for Definition 6.2. Then there exists some $Q \in \mathcal{P}$ and $F \in \mathcal{F}_P$ so that $\operatorname{aff} F = A$. Each $n-1$ dimensional face in $\mathcal{F}$ has $\operatorname{aff} F = \partial H$ for some $H \in \mathcal{H}_P$, and each lower dimensional face is an intersection of those hyperplanes. We may conclude that $A \in \mathcal{A}_P$ for Definition 2.4 since it is the intersection of hyperplanes of P. The intersection of A and P is nonempty since A contains a face of P. □

Though $\partial P \subseteq \bigcup \mathcal{A}_P$, in many cases, $\mathcal{A}_P$ under Definition 6.2 is substantially smaller than it is under Definition 2.4. Definition 6.2 excludes affine spaces that have an empty intersection with P. The pruning is possible because of the additional information in our non-convex polyhedral representation.

We use the following result to construct $\mathcal{A}_P$, Definition 6.2.

Lemma 6.4 *A necessary condition for a set of* $n-1$*-dimensional faces* $\phi \subseteq \mathcal{F}_P$ *to have* $\operatorname{aff}(\bigcap_{F\in\phi} F) \in \mathcal{A}_P$ *is that the angles between every pair of faces in* ϕ *is less than* 180 *degrees.*

Proof Let $F, G \in \phi$ with the angle between them greater than 180 degrees, we can choose a point $\mathbf{x} \in \operatorname{int}(F)$ so that the angle between $\overline{\mathbf{x}, \Pi_{F\cap G}(\mathbf{x})}$ and $\overline{\Pi_G(\mathbf{x}), \Pi_{F\cap G}(\mathbf{x})}$ is greater than 180 degrees. While $\mathbf{x} \in P$ and $\Pi_G(\mathbf{x}) \in P$ the line $\overline{\mathbf{x}, \Pi_G(\mathbf{x})}$, excluding its endpoints, is outside of P. There is no convex set with faces F and G, and therefor it is possible to construct an arrangement for $\mathcal{P}$ without the affine space. □

We can restrict the elements of $\mathcal{A}_P$ because an optimal point $\mathbf{x}$ over P is also the optimal point over some polyhedron $Q \in \mathcal{P}$, and therefore it can be found with the necessary criteria by looking at all the affine spaces of Q that contain faces of P.

Algorithms exist for decomposing non-convex polyhedra into their convex components, [2], however we achieve better results by maintaining the non-convex form. By iterating over $\mathcal{A}_P$ from definition 6.2, we iterate over every face of each polyhedron in $\mathcal{P}$ that might contain P's optimal point.

Corollary 6.5 *Let* $G \in \mathcal{P}$*, then if the optimal point* $\boldsymbol{x}$ *of* P *has* $\boldsymbol{x} \in G$*, either* $\boldsymbol{x} \in \arg\min_{\mathbb{R}^n} f$ *or* $\boldsymbol{x} \in \partial P$*.*

Proof We may consider the more general statement: If $\mathbf{x}$ is an optimal point of P, then $\mathbf{x} \in \arg\min_{\mathbb{R}^n} f$ or $\mathbf{x} \in \arg\min_{\partial P} f$ which is a direct result of the convexity of f. □

For purposes of checking the necessary criteria, we need to define the P-cone of an affine space, $A \in \mathcal{A}_P$, where P is non convex. The natural choice is to find a convex $Q \in \mathcal{P}$ and use Q_A. However, since we don't know the composition of $\mathcal{P}$, we need a practical way to build P_A. We do this exactly as we did in Algorithm 1.

Definition 6.6 If $A \in \mathcal{A}_P$, then there exists an $F \in \mathcal{F}_P$ such that $\operatorname{aff} F = A$. Every such F is the intersection $n-1$ dimensional faces, $\phi \subseteq \mathcal{F}_P$ such that $F = \bigcap \phi$. For each $G \in \phi$ we have an $H_G \in \mathcal{H}_P$ such that $\partial H_G = \operatorname{aff} G$. Then $P_A = \bigcap_{G\in\phi} H_G$.

Lemma 6.7 *If P is convex, then Definition 6.6 is equivalent to Definition 2.7.*

Remark 6.8 Let Q, R be convex polyhedra with $A \in \mathcal{A}_Q \cap \mathcal{A}_R$ and $\mathcal{H}_{Q_A} = \mathcal{H}_{R_A}$, then if A meets the Necessary Criteria 4.10 for Q, it also does for R. That is to say, the elements of $\mathcal{P}$ don't matter, only the neighborhood of A.

Definition 6.9 We redefine a min space and say that $A \in \mathcal{A}_P$ is a min space on a non-convex polyhedron, P, if there is a convex polyhedron $Q \subseteq P$ such that A is a min space on Q.

Existence of a min space (Definition 6.9) is immediate from the definition of a non-convex polyhedron, though unlike in Definition 4.8, it is not unique. The following corollary follows.

Corollary 6.10 *Each min space (Definition 6.9) meets The Necessary Criteria 4.10.*

Proof The necessary conditions for a space to be a min space remain the same, because for any $\mathbf{x} \in \arg\min_P f$ we have a $Q \in \mathcal{P}$ so that $\mathbf{x} \in \arg\min_Q f$. □

This means that if some $A \in \mathcal{A}_P$ meets the Necessary Criteria (4.10), exactly which $Q \in \mathcal{P}$ it's in doesn't matter.

The sufficient conditions, checking if $\mathbf{x} \in P$ change a bit. We don't know the polyhedra of $\mathcal{Q}$ and it will not work to check if the point is in all of the half spaces of P, since P is not necessarily the intersection of half spaces. We therefor do not check The Sufficient Criteria (4.14).

Proposition 6.11 (The Sufficient Criteria for a Non-convex Polyhedron) *Let $\mathcal{M}$ be the set of affine spaces that are candidates and have that for each $A \in \mathcal{M}$ there exists an $F \in \mathcal{F}_p$ such that* $\operatorname{aff} F = A$ *with* $\arg\min_A f \in F$, *then* $\arg\min_P f = \arg\min\{f(\boldsymbol{x}) | \boldsymbol{x} \in \bigcup \mathcal{M}\}$.

Proof Let $\mathbf{x} \in A \in \mathcal{M}$, then by the assumptions set above, $\mathbf{x} \in P$.

Remark 6.10 gives us $\arg\min_P f = \arg\min\{f(\mathbf{x}) | \mathbf{x} \in P$ and $\mathbf{x} \in \arg\min_A f$ where A meets the Nec. Criteria $\}$. □

Since the minimum on the right hand side of the equation is a taken from a finite set, it's easy to compute.

Remark 6.12 We have $P \in \mathcal{F}_P$, often with $\operatorname{aff} P = \mathbb{R}^n \in \mathcal{A}_P$. If $\mathbb{R}^n \in \mathcal{M}$, we can check $\arg\min_{\mathbb{R}^n} f$ for membership in P with an algorithm like the one in Akopyan et al. [1]. For checking membership in any other $F \in \mathcal{F}_P$, we note that F is a convex polyhedron. Checking membership in a F is substantially faster than checking membership P.

With the curated $\mathcal{A}_P$, and the adjusted membership test, Algorithm 1 may proceed as above, except that when a point is found to be in P, it is saved and the algorithm continues. On completion, the minimum of all the points that have been saved is the minimum of P. If the set of saved points is empty, there is no minimum. For details, see Algorithm 2.

Algorithm 2: Finds $\arg\min_P f$ for a non-convex polyhedron P

```
Input: A set of faces $\mathcal{F}_P$ and a function $f : \mathbb{R}^n \xrightarrow{conv.} \mathbb{R}$
Output: $\min_P f$
$\mathcal{M} \leftarrow \varnothing$
for $i \leftarrow 0$ to $\min(n, r)$ do
    for $A \in \mathcal{A}_P$ with $\operatorname{codim}(A) = i$ in parallel do
        if $\exists B \in \mathcal{B}_A$ s.t. $m_B \cap (P_A \setminus A) \neq \varnothing$ then
            $m_A \leftarrow m_B \cap P_A$
        else
            $m_A \leftarrow \arg\min_A f$ is computed and saved.
            Let $F \in \mathcal{F}$ such that aff $F = A$
            if $m_A \cap F \neq \varnothing$ then
                add $m_A$ to $\mathcal{M}$.
return $\arg\min\{f(\boldsymbol{x}) | \boldsymbol{x} \in \bigcup \mathcal{M}\}$
```

References

1. Akopyan, A., Bárány, I., Robins, S.: Algebraic vertices of non-convex polyhedra. Adv. Math. **308**, 627–644 (2017). Available from: https://doi.org/10.1016/j.aim.2016.12.026
2. Bajaj, C.L., Dey, T.K.: Convex decomposition of polyhedra and robustness. SIAM J. Comput. **21**(2), 339–364 (1992). Available from: https://doi.org/10.1137/0221025
3. Balakrishnan, A.V.: Introduction to Optimization Theory in a Hilbert Space. Lecture Notes in Operations Research and Mathematical Systems, vol. 42. Springer, Berlin (1971)
4. Bauschke, H.H., Combettes, P.L.: Convex Analysis and Monotone Operator Theory in Hilbert Spaces. CMS Books in Mathematics/Ouvrages de Mathématiques de la SMC. Springer, New York (2011). With a foreword by Hédy Attouch. Available from: https://doi.org/10.1007/978-1-4419-9467-7
5. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004). Available from: https://doi.org/10.1017/CBO9780511804441
6. Debnath, L., Mikusiński, P.: Introduction to Hilbert Spaces with Applications, 2nd edn. Academic, San Diego (1999)
7. Diamond, S., Takapoui, R., Boyd, S.: A general system for heuristic minimization of convex functions over non-convex sets. Optim. Methods Softw. **33**(1), 165–193 (2018). Available from: https://doi.org/10.1080/10556788.2017.1304548
8. Edelsbrunner, H.: Algebraic decomposition of non-convex polyhedra. In: 36th Annual Symposium on Foundations of Computer Science (Milwaukee, WI, 1995), pp. 248–257. IEEE Comput. Soc. Press, Los Alamitos (1995). Available from: https://doi.org/10.1109/SFCS.1995.492480
9. Edelsbrunner, H.: Geometry and Topology for Mesh Generation, vol. 7. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge (2006). Reprint of the 2001 original
10. Frank, M., Wolfe, P.: An algorithm for quadratic programming. Naval Res. Logist. Quart. **3**, 95–110 (1956). Available from: https://doi.org/10.1002/nav.3800030109
11. Gasnikov, A., Kabanikhin, S., Mohammed, A., Shishlenin, M.: Convex optimization in Hilbert space with applications to inverse problems. arXiv preprint arXiv:1703.00267 (2017)
12. Laiu, M.P., Tits, A.L.: An infeasible-start framework for convex quadratic optimization, with application to constraint-reduced interior-point methods. arXiv (2019). Available from: https://arxiv.org/abs/1912.04335

13. Neimand, E.D., Sabau, S.: Polyhedral constrained optimization using parallel reduction of linear inequality to equality constraints. arXiv (2021). Available from: https://arxiv.org/abs/2109.07565
14. Okelo, N.B.: On certain conditions for convex optimization in Hilbert spaces. Khayyam J. Math. **5**(2), 108–112 (2019). Available from: https://doi.org/10.22034/kjm.2019.88084
15. Plesník, J.: Finding the orthogonal projection of a point onto an affine subspace. Linear Algebra Appl. **422**(2–3), 455–470 (2007). Available from: https://doi.org/10.1016/j.laa.2006.11.003
16. Pólik, I., Terlaky, T.: Interior point methods for nonlinear optimization. In: Nonlinear Optimization, vol. 1989. Lecture Notes in Math., pp. 215–276. Springer, Berlin (2010). Available from: https://doi.org/10.1007/978-3-642-11339-0_4
17. Rutkowski, K.E.: Closed-form expressions for projectors onto polyhedral sets in Hilbert spaces. SIAM J. Optim. **27**(3), 1758–1771 (2017). Available from: https://doi.org/10.1137/16M1087540
18. Sloane, N.J.A.: Sequence a000984 central binomial coefficients. Available from: https://oeis.org/A000984
19. user2566092 (https://math.stackexchange.com/users/87313/user2566092). Find point on polyhedron nearest given point. Mathematics Stack Exchange. https://math.stackexchange.com/q/1134236 (version: 2015-02-05)
20. Wright, S.J.: Primal-Dual Interior-Point Methods. Society for Industrial and Applied Mathematics (1997). Available from: https://epubs.siam.org/doi/abs/10.1137/1.9781611971453

Mean Values: A Multicriterial Analysis

Vladislav V. Podinovski and Andrey P. Nelyubin

1 Introduction

Mean values are widely used in management, economics, sociology, engineering and other areas of theory and practice. In statistics (see, for example, [8, 22]), mean values are aggregate representations of the varying characteristics of a group of homogeneous objects. Mean values cancel out random variations of a particular characteristic and tend to represent the effect caused by the main factors affecting it. Mean values allow us to compare the levels of the same characteristic in different groups of objects and to investigate the causes of such differences.

It is known that it is impossible to define a universally applicable notion of the mean value which satisfies all desirable properties [1, 8]. Instead, different notions of the mean value are required for different problems and situations. However, in some applications, it may be unclear which of the known mean values should be used, and different means may point to different conclusions. Policy recommendations in such situations may become problematic [6, 8, 10, 12, 16].

Grabisch et al. [7] regarded mean values as idempotent aggregation functions and concluded that the class of such functions "is huge, making the problem of choosing the right function (or family) for a given application a difficult one".

In this paper, we consider new approaches to the definition of the mean value based on the ideas and methods of multicriteria optimization. Such means turn out to be multi-valued, i.e., represented by sets of points. These allow two interpretations,

V. V. Podinovski
National Research University Higher School of Economics, Moscow, Russian Federation
e-mail: podinovski@mail.ru

A. P. Nelyubin (✉)
Mechanical Engineering Research Institute RAS, Moscow, Russian Federation
e-mail: nelubin@gmail.com

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_17

either as the range of possible mean values in some specific situations (characterized by scale properties, such as equal importance or ordinality, and/or transfer principles), or as whole sets for the given sample.

2 Definition of Mean Values as Nondominated Points

Let X be the set of real numbers consisting of at least $n \geq 2$ elements referred to as data or points. These elements are typically obtained as a result of measurement of some characteristic:

$$X = \{x_1, x_2, \ldots, x_n\}. \tag{1}$$

These data are assumed homogeneous in the sense that they are obtained by utilizing the same scale of measurement [21, 23]. We assume that the data (1) are quantitative, i.e., the measurement is performed either on the interval scale or on the ratio scale [15].

The elements of the set (1) can be ranked in the non-decreasing and non-increasing order.

$$X_{\uparrow} =< x_{(1)}, x_{(2)}, \ldots, x_{(n)} >; X_{\downarrow} =< x_{[1]}, x_{[2]}, \ldots, x_{[n]} >, \tag{2}$$

where $x_{(1)} \leq x_{(2)} \leq \ldots \leq x_{(n)}$ and $x_{[1]} \geq x_{[2]} \geq \ldots \geq x_{[n]}$. In statistics, the set (1) is typically referred to as a sample and its non-decreasing sequence $X_{\uparrow}$ as a variational series.

Let x be an arbitrary fixed number (a point in Re). Its distance from any point x_i from X is given by $y_i = |x - x_i|$. Then the distance from x to the dataset X can be characterized by the vector $y = (y_1, y_2, \ldots, y_n)$. We can view this vector as the value of the vector criterion $f(x) = (f_1(x), f_2(x), \ldots, f_n(x))$, where $f_i(x) = |x - x_i|$, which is an element of the nonnegative quadrant $\mathrm{Re}^n_+ = [0, +\infty)^n$.

Let P^Γ be a preference relation (strict partial order) on Re^n_+, where Γ is information about the preferences with respect to distance: if $yP^\Gamma y'$, where $y = f(x)$ and $y' = f(x')$, then the point x is closer to the dataset X than x'.

The relation P^Γ generates the corresponding relation P_Γ on the numeric axis Re: $xP_\Gamma x' \iff f(x)P^\Gamma f(x')$.

Therefore, any candidate that we may choose as the closest to X and representing the set X must be nondominated under P_Γ. If the set $G^\Gamma(X)$ of nondominated under P_Γ points is externally stable, we refer to all such points as pn-means (principal new means) and, more specifically (reflecting the information Γ), as the means with respect to P_Γ.

If there is no further information about the preferences of the DM on Re^n_+, we obtain the Pareto relation $P^\varnothing$ defined as follows:

$$yP^\varnothing z \iff y_i \leq z_i, i = 1, 2, \ldots, n; y \neq z.$$

Relation $P^{\varnothing}$ generates the Pareto relation $P_{\varnothing}$ on Re: $xP_{\varnothing}x' \iff f(x)P^{\varnothing}f(x')$.

Theorem 1 The set of all means of the dataset (1) with respect to $P_{\varnothing}$ is the segment $G^{\varnothing}(X) = \overline{X} = [x_{(1)}, x_{(n)}]$, where $x_{(1)} = \min_{i\in N} x_i$ and $x_{(n)} = \max_{i\in N} x_i$, $N = \{1, 2, \ldots, n\}$. This set is externally stable.

Therefore, the notion of the means with respect to $P_{\varnothing}$ is equivalent to the means in the sense of Cauchy.

Proofs of this and the following theorems can be found in [19, 20].

Let us note that, if the function φ is increasing on Re_+, then changing the original criteria $f_i(x) = |x - x_i|$ by $\varphi(f_i(x))$ does not change the set $G^{\varnothing}(X)$. For example, one can use "smooth" criteria $f_i(x) = (x - x_i)^2$. Therefore, the original use of formula $f_i(x) = |x - x_i|$ as a measure of distance is not essential and is not a limiting assumption for the suggested approach.

3 Mean Values for Equally Important Criteria

In this section, we assume that all criteria are equally important [17] and denote this information E. In this case, the distance from the point x to the dataset X is represented by the preference relation P_E on Re, which is defined by the two equivalent decision rules [17], where $f_i(x) = |x - x_i|$:

$$xP_Ex' \iff \left(f_{(1)}(x) \le f_{(1)}\left(x'\right), f_{(2)}(x) \le f_{(2)}\left(x'\right), \ldots, f_{(n)}(x) \le f_{(n)}\left(x'\right)\right),$$

and at least one of these inequalities is strict;

$$xP_Ex' \iff \left(f_{[1]}(x) \le f_{[1]}\left(x'\right), f_{[2]}(x) \le f_{[2]}\left(x'\right), \ldots, f_{[n]}(x) \le f_{[n]}\left(x'\right)\right),$$

and at least one of these inequalities is strict.

In this case, the pn-means (with respect to P_E) (elements of the set $G^E(X)$) are the points on the numerical axis which are nondominated under P_E.

Theorem 2 We have $G^E(X) \subseteq G^{\varnothing}(X) = \overline{X}$, and the set $G^E(X)$ is externally stable.

Note that, if the function φ is increasing on Re_+, then changing the original criteria $f_i(x)$ to criteria $\varphi(f_i(x))$ does not change the relation P_E and the set $G^E(X)$.

Let us consider examples of sets $G^E(X)$ constructed according to the methods described in Sect. 5.

Example 1 Let $n = 3$ and $X = \{1, 2, 5\}$. In this example, $G^E(X) = [1.5, 3]$.

In the above example, the set $G^E(X)$ is a single line segment. However, for large n, this set may be the union of several segments, excluding their endpoints.

Example 2 For $n = 6$ and different sets X, we have:

$$G^E(\{10, 11, 15, 61, 107, 110\}) = [10.5, 83) \cup (83.5, 85) \cup (106.5, 108)\,;$$

$$G^E(\{10, 11, 40, 55, 70, 110\}) = [10.5, 18) \cup (18; 67.5) \cup (68, 75)\,;$$

$$G^E(\{10, 57, 61, 64, 109, 110\}) = (56.5, 57.5) \cup (58.5, 88.5) \cup (108, 109.5]\,.$$

Examples 1 and 2 also illustrate the following result.

Theorem 3 Let the distance between two adjacent elements $x_{(i)}$ and $x_{(i+1)}$ of the variational series (2) be the smallest among all other pairs of adjacent elements of this series, and let these two elements be uniquely defined. Then the midpoint $x^c = ½(x_{(i)} + x_{(i+1)})$ is an element of $G^E(X)$. Moreover, if $x_{(i)}$ is $x_{(1)}$ or if $x_{(i+1)}$ is $x_{(n)}$, then x^c is the left or, respectively, right, endpoint of the set $G^E(X)$.

If $x_{(1)} \neq x_{(n)}$ and $(x_{(1)}, x_{(n)}) \not\subset G^E(X)$, for some values of parameter s, the power mean.

$$g^s(X) = \left(\frac{1}{n}\sum_{i=1}^{n}(x_i)^s\right)^{1\backslash s}, s \neq 0.$$

is not the mean with respect to P_E. This is because, as s increases on Re, the function $g^s(X)$, extended to preserve continuity, passes through all values from the interval $(x_{(1)}, x_{(n)})$ [8]. However, we have the following result:

Theorem 4 The arithmetic mean is a mean with respect to P_E, i.e., $g^1(X) \in G^E(X)$.

Example 3 According to Example 2, for $X = \{10, 57, 61, 64, 109, 110\}$ we have: $G^E(X) = (56.5, 57.5) \cup (58.5, 88.5) \cup (108, 109.5]$. In this example, the geometric mean $g^0(X) = 54.66 \notin G^E(X)$ and harmonic mean $g^{-1}(X) = 35.75 \notin G^E(X)$, and $g^1(X) = 68.5 \in G^E(X)$. In Example 1, for $X = \{1, 2, 5\}$, we have $G^E(X) = [1.5, 3]$. Here, the quadratic mean $g^2(X) = 3.162 \notin G^E(X)$, but $g^1(X) = 2.67 \in G^E(X)$.

Theorem 5 The median is a mean with respect to P_E, i.e., if n is an odd integer and the median is unique, we have $\mu(X) = x_{\left(\frac{n+1}{2}\right)} \in G^E(X)$. If n is an even number, the median is not unique and we have $\mu(X) = \left[x_{\left(\frac{n}{2}\right)}, x_{\left(\frac{n}{2}+1\right)}\right] \subseteq G^E(X)$.

Examples 1 and 2 provide illustrations to the above theorem.

It should be noted the following peculiarity of the means with respect to P_E: if the points $x_i \in X$ and $x_j \in X$, $x_i < x_j$, are included in $G^E(X)$, then the point $x_k \in X$, such that $x_i < x_k < x_j$, may not belong to $G^E(X)$!

Example 4 For $n = 7$ and different sets X, we have:

$$X' = \{1, 2, 3, 6, 8, 9, 11\}\,, G^E\left(X'\right) = [2; 3) \cup (3; 8.5]\,;$$

$$X'' = \{1, 2, 3, 7, 8, 10, 11\}, G^E\left(X''\right) = [2; 3) \cup (3.5; 9].$$

Here $x_{(2)}$, $x_{(4)} \in G^E(X)$, whereas $x_{(3)} \notin G^E(X)$ for both sets $X = X'$ and $X = X\prime\prime$. Moreover, the point $x_{(3)} = 3$ is a punctured point of the set $G^E(X')$ (it is dominated under P_E by the point $x_{(5)} = 8$, and in its arbitrarily small neighborhood there are points nondominated under P_E).

This feature clearly violates the very principle of constructing means as points closest to points from X, and is not consistent with the intuitive concept of a mean value. Therefore, the presence of this feature can be considered as a paradox of means with respect to P_E.

4 Mean Values for Equally Important Criteria Measured on the First Ordered Metric Scale

Let y be any vector estimate such that $y_i > y_j$. Consider any $\delta > 0$ such that $y_i - \delta \geq y_j + \delta$. Define the vector estimate z by replacing component y_i by $y_i - \delta$ and y_j by $y_j + \delta$, but $y_i - \delta \geq y_j + \delta$. Moving from y to z reduces the larger deviation y_i from one point in the sample and increases a smaller deviation y_j from a different point, by the same amounts δ. The resulting set of distances becomes closer to the ideal set of minimally possible equal deviations. Assume that, for any y and δ described above, the vector estimate z is preferred to the original vector estimate y, in the sense that z is "closer" to X than y and is therefore more suitable for the definition of the mean. Denote Δ the information about the described principle. Such approach is an analogue of Pigou-Dalton's principle of transfer for income distribution [2, 5]. This means that the equally important criteria have a common first ordered metric scale [4]. The preference relation $P_{E\Delta}$, generated on Re^n by the joint information E and Δ, is defined by the following decision rule [14, 18]:

$$\begin{aligned} xP_{E\Delta}x' \iff & f_{[1]}(x) \leq f_{[1]}\left(x'\right), f_{[1]}(x) + f_{[2]}(x) \leq f_{[1]}\left(x'\right) + f_{[2]}\left(x'\right), \ldots \\ & \ldots f_{[1]}(x) + f_{[2]}(x) + \cdots + f_{[n]}(x) \leq f_{[1]}\left(x'\right) + f_{[2]}\left(x'\right) + \ldots f_{[n]}\left(x'\right), \end{aligned}$$

and at least one of these inequalities is strict. In this case, the pn-means are the points that are nondominated under $P_{E\Delta}$. Because $P_{E\Delta} \supset P_E$, we have $G^E(X) \supseteq G^{E\Delta}(X)$.

Theorem 6 The arithmetic mean is a mean with respect to $P_{E\Delta}$, i.e., $g^1(X) \in G^{E\Delta}(X)$.

Theorem 7 If n is odd, the median (which is uniquely defined), is a mean with respect to $P_{E\Delta}$, i.e., $\mu(X) \in G^{E\Delta}(X)$. If n is even and the median is not uniquely defined, we only have $\mu(X) \cap G^{E\Delta}(X) \neq \varnothing$.

Example 5 If $n = 5$ and $X = \{1, 2, 3, 5, 11\}$, we have $G^{E\Delta}(X) = [3, 6]$, $\mu(X) = 3$ and $g^1(X) = 4.4$. If $n = 4$ and $X = \{10, 11, 12, 110\}$, we have $G^{E\Delta}(X) = [11.5,$

60], $\mu(X) = [11, 12]$ and $g^1(X) = 35.75$. If $X = \{10, 11, 20, 110\}$, we have $G^{E\Delta}(X) = [15.5, 60]$, $\mu(X) = [11, 20]$ and $g^1(X) = 37.75$.

Let us define the set $H = \{1, 2, \ldots, h\}$, where $h = \lfloor(n + 1)/2\rfloor$ is the integer part of $(n + 1)/2$.

Theorem 8 The set $G^{E\Delta}(X)$ is externally stable and coincides with the segment $[\alpha, \beta]$, where

$$\alpha = \frac{1}{2}\min\nolimits_{p\in H}\left(x_{(p)} + x_{(n+1-p)}\right), \beta = \frac{1}{2}\max\nolimits_{p\in H}\left(x_{(p)} + x_{(n+1-p)}\right) \tag{3}$$

Example 6 For $n = 5$, we have $h = \lfloor(n + 1)/2\rfloor = 3$ and $H = \{1, 2, 3\}$. For $X = \{1, 2, 7, 8, 11\}$, using Theorem 8, we have:

$$\begin{aligned}\alpha = \frac{1}{2}\ \min\left\{x_{(1)} + x_{(5)}, x_{(2)} + x_{(4)}, x_{(3)} + x_{(3)}\right\} &= \frac{1}{2}\ \min\{1 + 11, 2 + 8, 7 + 7\}\\ &= \frac{1}{2}\ \min\{12, 10, 14\} = 5;\end{aligned}$$

$$\beta = \frac{1}{2}\ \max\left\{x_{(1)} + x_{(5)}, x_{(2)} + x_{(4)}, x_{(3)} + x_{(3)}\right\} = \frac{1}{2}\ \max\{12, 10, 14\} = 7;$$

Therefore, $G^{E\Delta}(X) = [\alpha, \beta] = [5, 7]$.

5 On the Construction of Sets of Mean Values

For the construction of the set $G^E(X)$, we can use known methods of multicriteria optimization developed for the construction of the sets of nondominated variants [17]. Such methods utilize families of functions that are increasing (decreasing), or at least non-decreasing (non-increasing) with respect to P_E. For example, we can solve a parametric program which minimizes the function of single variable $\psi(f(x)|c) = \min_{\pi\in\Pi} \max_{i\in N} \{f_{\pi(i)}(x) - c_i\}$ on the set X, by varying the vector parameter $c \in f\left(\overline{X}\right)$. However, even if n is not very large, the number $n!$ of terms of this function (with respect to which the maximization is performed) turns out unacceptably large.

Taking into account that the set X is one-dimensional, we can utilize a different approach. Namely, we can consider a dense grid with the small step h which covers the set X, and identify the nondominated (with respect to P_E) points of this grid by simple enumeration [9]. The step h depends on the required precision and can decrease in the process of calculations of the set $G^E(X)$. We used this approach for the construction of the set $G^E(X)$ in Examples 1 и 2.

Example 7 Let us demonstrate the construction of the set $G^E(X)$ for $X = \{1, 2, 5, 9, 11\}$. Using computer for the calculations, while reducing the step length h, we obtain the following results:

$h = 1$:	$[2, 7] \cup [9, 9]$.
$h = 0.1$:	$[1.5, 7.4] \cup [8.6, 9.4]$.
$h = 0.01$:	$[1.50, 7.49] \cup [8.51, 9.49]$.
$h = 0.001$:	$[1.500, 7.499] \cup [8.501, 9.499]$.
$h = 0.0001$:	$[1.5000, 7.4999] \cup [8.5001, 9.4999]$

Using the enumeration approach with $h = 0.01$, we found out that the point 4.5 dominates the points 7.5 and 8.5. Similarly, the point 2.5 dominates the point 9.5. Therefore, by Theorem 2, we have $G^E(X) = [1.5, 7.5) \cup (8.5, 9.5)$.

Let us highlight another result that may be useful in the construction of the set $G^E(X)$.

Theorem 9 Let vector estimates of all $x \in X$ be located at the points of some uniform grid covering X. Then, in order to test if any grid point is a mean with respect to P_E, it suffices to compare its vector estimate only with the vector estimates of all the other points of the grid.

Let us note that the uniform grid required by the conditions of Theorem 9 can always be constructed if all points in X are rational numbers. In practical applications, these would typically be integer numbers or decimal fractions.

It is worth noting that it is easier to construct the set of means $G^{E\Delta}(X)$ than the set $G^E(X)$. According to Theorem 8, the set $G^{E\Delta}(X)$ is easily found by calculating the endpoints α and β of the segment $[\alpha, \beta]$ using formulae (3) – see Example 6.

6 On Comparing Multi-valued Means

In practice, it is important that we can compare the mean values measured on the same scale. For the means that are uniquely defined, this is a simple task of comparing the two numerical values. In the case of multi-valued means, in statistics, it is common to substitute such means by a single number, e.g., in the case of a median when n is an even number.

The set $G^\Gamma(X)$ consists of l intervals with the endpoints x^1, x^2; x^3, x^4; $\ldots$; x^{2l-1}, x^{2l}, and these intervals do not intersect with each other. Define the length $D^\Gamma(X)$ of the set $G^\Gamma(X)$ as the sum of the lengths of all these intervals: $D^\Gamma(X) = \sum_{k=1}^{l} \left|x^{2k} - x^{2k-1}\right|$. Furthermore, define $D_x^\Gamma(X)$ the length of the part of the set $G^\Gamma(X)$ that is located to the right of the point x. It includes the (part) of one interval and all the other intervals located to the right of x. The relative length $d_x^\Gamma(X)$ is defined as the ratio $d_x^\Gamma(X) = D_x^\Gamma(X)$: $D^\Gamma(X)$.

Because none of the points of the set $G^\Gamma(X)$ has any advantages (in the sense of representing the sample) compared to its other points, any of them may be regarded as an equally valid candidate for the choice of the mean. This is analogous to the principle of insufficient reason for decision making under ignorance [13]. Using first-order stochastic dominance [11], we say that the mean $G^\Gamma(X')$ is not less than the mean $G^\Gamma(X'')$ and state this as $G^\Gamma(X')) \succsim G^\Gamma(X'')$, if $d_x^\Gamma\left(X'\right) \geq d_x^\Gamma\left(X''\right)$ for each $x \in \mathrm{Re}$. If the latter inequality is strict for at least one $x \in \mathrm{Re}$, the former mean is greater than the latter. This relationship between the means ("is not less than") is a partial quasi-order. The corresponding relation "is greater than" is denoted $\succ$ and is a partial strict order (it is irreflexive and transitive). This strict relation is essentially a probabilistic dominance relation, or a strict first-order stochastic dominance relation [11]. Note that we have $d_x^\Gamma(X) = 1 - F(x)$, where $F(x)$ is the cumulative distribution function corresponding to the uniform distribution with the density equal to 1 / $D^\Gamma(X)$ on $G^\Gamma(X)$ and equal to zero outside $G^\Gamma(X)$.

It is clear that the relation $\succcurlyeq$ is weak in the sense that it would typically not result in a definitive comparison of the means. Relation $\succcurlyeq$ can be extended using the ideas of second-order stochastic dominance, but this approach does not appear to be sufficiently effective in practice either.

Another approach would be to "compress" the means that are not uniquely defined to single-valued means. However, this would lead to a loss of information, and the results of comparison would be approximate. For example, let the mean $G^\Gamma(X)$ consist of several not intersecting intervals defined by the endpoints x^1, x^2; x^3, x^4; …; $x^{2l\text{-}1}, x^{2l}$. We can represent this mean by its the centre of mass $x^\Gamma(X)$ and refer to it as the centroid mean.

Example 8 Let $G^E(X') = [1, 2) \cup (5, 8)$ and $G^E(X'') = [1.5, 4.5] \cup (8, 9]$. We have:

$$x^E\left(X'\right) = (1.5 \cdot 1 + 6.5 \cdot 3)/4 = 5.25;\ x^E\left(X''\right) = (3 \cdot 3 + 8.5 \cdot 1)/4 = 4.375.$$

Because 5.25 > 4.375, we can accept that the mean $G^E(X')$ is greater than $G^E(X'')$.

It is useful to note that, if $G^\Gamma(X') \succ G^\Gamma(X'')$, then $x^\Gamma(X') > x^\Gamma(X'')$ [11].

It is worth noting that it easier to compare the means $G^{E\Delta}(X')$ and $G^{E\Delta}(X'')$ than the means $G^E(X')$ and $G^E(X'')$, because the former are the segments $[\alpha', \beta']$ and $[\alpha'', \beta'']$ respectively. Because the graph of the function $d_x^{E\Delta}(X)$ is a broken line consisting of the single segment $[\alpha, \beta]$ on which it decreases from 1 to 0, $G^{E\Delta}(X')$ $G^{E\Delta}(X'')$ is true if and only if $\alpha' \geq \alpha''$ and $\beta' \geq \beta''$.

For the simplified application of the mean $G^{E\Delta}$, we can represent the segment $[\alpha, \beta]$ by its midpoint $\gamma = ½\,(\alpha + \beta)$, which can be referred to as the centroid mean (with respect to $P_{E\Delta}$).

Example 9 The means of the real GDP per capita in Europe calculated based on the data from Eurostat [3] are shown in Table 1 and Fig. 1.

Table 1 Mean real GDP per capita in Europe (in Euro)

Year	2012	2013	2014	2015	2016	2017	2018	2019
m	25,741	25,825	26,257	27,063	27,564	28,274	28,909	29,249
μ	21,780	20,400	20,250	21,020	22,270	23,200	24,120	24,570
α	19,720	19,745	20,250	21,020	21,995	22,840	23,245	23,485
β	40,840	41,310	42,015	42,970	43,850	43,750	44,335	44,545
γ	30,280	30,528	31,133	31,995	32,923	33,295	33,790	34,015

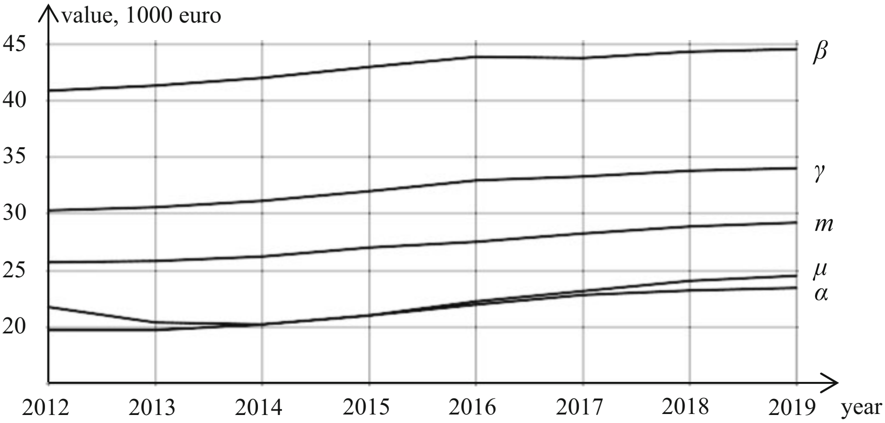


Fig. 1 Means of the GDP per capita in Europe

Table 1 shows that the GDP per capita m is increasing in the period from 2012 to 2019, but the median GDP per capita μ is decreasing until 2014 and is increasing afterwards. Therefore, it is impossible to make a definite conclusion about the GDP growth in the given period. However, the mean $G^{E\Delta}$ (defined by its boundaries α and β) is increasing (there is only an insignificant decrease of β in 2017), and the condensed mean γ is increasing over the whole period. This observation supports the conclusion that the GDP per capita in Europe has been increasing in the given period.

7 On Stability of Pn-Means

The question of stability of the means with respect to small perturbations of the data (1) is important from both theoretical and practical points of view.

Because $G^{\varnothing}(X) = \overline{X} = [x_{(1)}; x_{(n)}]$, a small change of the values x_i may lead only to small changes of $x_{(1)}$ and $x_{(n)}$. Therefore, the set of the means $G^{\varnothing}(X)$ is stable.

However, the mean with respect to P_E may not be stable in the sense that a very small perturbation of a single point in X may lead to a noticeable change of the set $G^E(X)$. The following examples illustrate this possibility.

Example 10 For $X = \{1, 2, 3\}$, we have $G^E(X) = [1.5, 2.5]$. However, for $X^\varepsilon = \{1, 2 - \varepsilon, 3\}$, where $\varepsilon > 0$ is very small, we have $G^E(X^\varepsilon) = [1.5 – 0.5\varepsilon, 2]$. The right endpoint of the set of the means with respect to P_E has changed by 0.5.

Example 11 For $X = \{10, 25, 40, 110\}$, we have $G^E(X) = [25, 60]$. However, for $X^\varepsilon = \{10, 25, 40 + \varepsilon, 110\}$, where $\varepsilon > 0$ is very small, we have $G^E(X^\varepsilon) = [17.5, 60]$. It is interesting that, although only one point in X has increased by a very small ε, the left endpoint of the set of the means (with respect to P_E) has decreased by 7.5.

Let us now consider the issue of stability of the mean with respect to $P_{E\Delta}$.

Example 12 In the setting of Example 10, we have $G^{E\Delta}(X) = \{2\}$ and $G^{E\Delta}(X^\varepsilon) = [2 - \varepsilon, 2]$. Here, a change of one of the data points in X by ε leads to the change of one of the endpoints of the set of the means with respect to $P_{E\Delta}$ by the same ε.

Example 13 Under the conditions of Example 11, we have $G^{E\Delta}(X) = [32.5, 60]$ и $G^{E\Delta}(X^\varepsilon) = [32.5 + 0.5\varepsilon, 60]$. In this example, a change of one of the data points in X by ε results in the change of one of the endpoints for the set of the means by 0.5ε.

Consider the general case. Suppose that the dataset X stated by (1) has changed to the set $X^\varepsilon = \{x_1 + \varepsilon_1, x_2 + \varepsilon_2, \ldots, x_n + \varepsilon_n\}$, where $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n$ are arbitrary numbers.

Theorem 10 The mean with respect to $P_{E\Delta}$ is stable in the following sense: If X is changed to X^ε, the endpoints of the set of the means $G^{E\Delta}(X) = [\alpha, \beta]$ do not change by more than the following value:

$\max\{|\varepsilon_1|, |\varepsilon_2|, \ldots, |\varepsilon_n|\}$.

Therefore, the means with respect to $P_\varnothing$ and $P_{E\Delta}$ are stable with respect to small perturbations of the dataset (1), while the means with respect to P_E may be noticeably unstable.

8 The Case of Data with Repetitions

Assume that the dataset allows repetitions, i.e., the point x_1 occurs β_1 times, x_2 occurs β_2 times, …, x_n occurs β_n times. In this case, the dataset (1) is replaced by Table 2.

Table 2 Data with repetitions

Value x_i	x_1	x_2	…	x_n
Weight β_i	β_1	β$_2$	…	β_n

In statistics, the numbers β_i are referred to as weights or (absolute) frequencies, and they are used for the calculation of weighted means.

All our results obtained above, starting with the definition of pn-means, are extended to the described more general case. For this, we consider the dataset consisting of the repeating values x_1 (β_1 times), x_2 (β_2 times) and so on, i.e., we restate the data in Table 2 as follows:

$$\left(\underbrace{x_1, \ldots, x_1}_{\beta_1}, \underbrace{x_2, \ldots, x_2}_{\beta_2}, \ldots, \underbrace{x_n, \ldots, x_n}_{\beta_n}\right).$$

The use of the described methods of construction of pn-means in this case may be computationally demanding as the dimension of the problem becomes very large for large values β_i. To overcome this problem, we may use decision rules developed in theory of qualitative criteria importance measured on continuous scale [18]. In this approach, we treat the integer numbers β_i as quantitative coefficients reflecting the importance of criteria and use notation P^β and $P^{\beta\Delta}$ to denote the corresponding relations instead of P^E and $P^{E\Delta}$.

To state the relevant decision rules for the vector estimates y and z, define the following set and values:

$$\begin{aligned} W(y, z) &= \{y_1\} \cup \{y_2\} \cup \cdots \cup \{y_m\} \cup \{z_1\} \cup \{z_2\} \cup \cdots \cup \{z_m\} \\ &= \left\{w_1, w_2, \ldots, w_q\right\}, w_1 > w_2 > \cdots > w_q; \end{aligned}$$

$$\begin{aligned} &= \sum\nolimits_{i:y_i \geq w_k} \beta_i, \;\; b_k(z) = \sum\nolimits_{i:z_i \geq w_k} \beta_i \; b_k(y) = \sum\nolimits_{i:y_i \geq w_k} \beta_i \\ &= \sum\nolimits_{i:z_i \geq w_k} \beta_i, k = 1, 2, \ldots, q-1; \end{aligned}$$

$$d_k(y) = \sum_{j=1}^{k} b_j(y)\left(w_j - w_{j+1}\right), k = 1, 2, \ldots, q-1.$$

Decision rule for P^β:

$$yP^\beta z \iff b_k(y) \leq b_k(z), k = 1, 2, \ldots, q-1, \tag{4}$$

and at least one of these inequalities is strict.

Decision rule for $P^{\beta\Delta}$:

$$yP^{\beta\Delta}z \iff d_k(y) \leq d_k(z), k = 1, 2, \ldots, q–1, \tag{5}$$

Table 3 Data in Example 14

Value x_i	1	2	4	5	7	9	11
Weight β_i	2	1	4	1	2	3	1

Table 4 Data for Example 15

Number i	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Value $x_{(i)}$	1	1	2	4	4	4	4	5	7	7	9	9	9	11

and at least one of these inequalities is strict.

Example 14 Consider the dataset in Table 3.

Using decision rules (4) and (5), let us compare points 5 and 3 which have the following vector estimates: $y = f(5) = (4, 3, 1, 0, 2, 4, 6)$ and $z = f(3) = (2, 1, 1, 2, 4, 6, 8)$. In this case, $W = (8, 6, 4, 3, 2, 1, 0)$. Therefore, $q = 7$. We have:

$b(y) = (b_1(y), b_2(y), \ldots, b_6(y)) = (0, 1, 6, 7, 9, 13);$
$b(z) = (b_1(z), b_2(z), \ldots, b_6(z)) = (1, 4, 6, 6, 9, 14);$
$d(y) = (d_1(y), d_2(y), \ldots, d_6(y)) = (0, 2, 8, 15, 24, 37);$
$d(z) = (d_1(z), d_2(z), \ldots, d_6(z)) = (2, 10, 16, 22, 31, 45).$

Note that $b_1(y) = 0 < b_1(z) = 1$ but $b_4(y) = 7 > b_1(z) = 6$. According to (4), neither $yP^{\beta}z$ nor $zP^{\beta}y$ is true. However, because all 6 inequalities (5) are true and at least one of them is strict, we have $yP^{\beta\Delta}z$.

Note that formula (3) is easier to use if we first rearrange data with repetitions in the form (1).

Example 15 Consider the data from Table 3 of Example 14. We can rearrange these data as in Table 4 in which we specify the ordinal number i for each point and the corresponding value $x_{(i)}$.

Using formulae (3), we consecutively calculate:

$$\begin{aligned}\alpha = \tfrac{1}{2}\ \min \big\{&x_{(1)} + x_{(14)}, x_{(2)} + x_{(13)}, x_{(3)} + x_{(12)}, x_{(4)} + x_{(11)}, x_{(5)} + x_{(10)},\\ &x_{(6)} + x_{(9)}, x_{(7)} + x_{(8)}\big\} =\\ = \tfrac{1}{2}\ &\min\{1+11, 1+9, 2+9, 4+9, 4+7, 4+7, 4+5\}\\ = \tfrac{1}{2}\ &\min\{12, 10, 11, 13, 11, 9\} = 4.5;\end{aligned}$$

$$\begin{aligned}\beta = \tfrac{1}{2}\ \max \big\{&x_{(1)} + x_{(14)}, x_{(2)} + x_{(13)}, x_{(3)} + x_{(12)}, x_{(4)} + x_{(11)}, x_{(5)} + x_{(10)},\\ &x_{(6)} + x_{(9)}, x_{(7)} + x_{(8)}\big\} =\\ = \tfrac{1}{2}\ &\max\{12, 10, 11, 13, 11, 11, 9\} = 6.5.\end{aligned}$$

Therefore, $G^{\beta\Delta}(X) = [4.5, 6.5]$.

9 Conclusion

In this paper, we introduced new notions of the means based on unifying ideas of multicriteria optimization. These notions do not require certain properties of the means, which are typically assumed by the conventional approaches in statistics and which can sometimes complicate the choice of a suitable mean in some problems [8]. Instead, our approach utilizes the distance from a current point to each point of the dataset. The proximity from a current point to all points in the dataset is characterized by the vector components of which are the distances between the current point and each point of the dataset. The means are defined as the points which are nondominated with respect to the preference relation among the vectors of distances characterized by scale properties, such as equal importance or ordinality, and/or transfer principles.

It turns out that such means are typically not unique and that their sets may have a complex structure. This potentially complicates the calculation of such means for large samples. However, the advances in computer and software technologies make this computational issue less problematic.

The suggested means allow two different interpretations, either as the range of possible mean values in some specific situations characterized by scale properties, or as whole sets that characterize the chosen sample.

Among the new means introduced in this paper, the means defined with respect to relation $P_{E\Delta}$ should be of the most practical interest. The set $G^{E\Delta}(X)$ of such means has a simple structure (it is a segment $[\alpha, \beta]$), and it is stable with respect to small perturbations of the dataset. Furthermore, there exists a simple exact method for the calculation of the set $G^{E\Delta}(X)$. Namely, we have suggested analytical formulae for the calculation of the endpoints α и β.

In applications, the comparison of different multi-valued means developed in our paper may be uninteresting because they usually turn out to be incomparable under the corresponding partial preference relation. However, in some problems, the described multi-valued approach has advantages over the use of known means (see, e.g., Example 9). If, instead of the set of pn-means, we consider their corresponding centres of mass, then such centroid means are uniquely defined. The latter are equally operational as the conventional means and but are less informative than the original pn-means. For example, instead of the mean $G^{E\Delta}(X) = [\alpha, \beta]$, we may use the corresponding centroid mean (with respect to $P_{E\Delta}$) $\gamma = ½\,(\alpha + \beta)$.

The suggested new means are a useful complement to the range of conventional means used in statistics. Among further research avenues arising from our paper, let us note development of new pn-means under different assumptions about the properties of the scales of measurement and corresponding computational methods.

Acknowledgments This work is an output of a research project implemented as part of the Basic Research Program at the National Research University Higher School of Economics (HSE University).

References

1. Beliakov, G., Pradera, A., Calvo, T.: Aggregation functions: a Guide for Practitioners. Springer, Berlin (2007)
2. Dalton, H.: The measurement of the inequality of incomes. Economic J. **30**, 348–361 (1920)
3. Eurostat: Real GDP per capita. https://ec.europa.eu/eurostat/databrowser/view/sdg_08_10/default/table?lang=en. (2020).
4. Fishburn, P.C.: Decision and Value Theory. Wiley, New York (1964)
5. Fishburn, P.C., Willig, R.D.: Transfer principles in income redistribution. J. Public Econom. **25**, 323–328 (1984)
6. Foster, C.: Being mean about the mean. Math. Sch. **43**, 32–33 (2014)
7. Grabisch, M., Marichal, J.-L., Mesiar, R., Pap, E.: Aggregation functions: means. Inf. Sci. **181**, 1–22 (2011)
8. Gini, C.: Le Medie. Ulet, Torino (1957)
9. Knuth, D.E.: The Art of Computer Programming: Vol. 3: Sorting and Searching, 2nd edn. Addison-Wesley, New York (1998)
10. Kricheff, R.S.: Means move – analyze the averages. In: That Doesn't Work Anymore – Retooling Investment Economics in the Age of Discruption, pp. 37–42. De Gruyter, Berlin (2018)
11. Levy, H.: Stochastic dominance and expected utility: survey and analysis. Manag. Sci. **38**, 555–593 (1992)
12. Lewontin, R., Levins, R.: The politics of averages. Capital. Nat. Social. **11**, 111–114 (2000)
13. Luce, R.D., Raiffa, H.: Games and Decisions. Wiley, New York (1957)
14. Marshall, A.W., Olkin, I.: Inequalities: Theory of Majorization and its Applications. Academic, New York (1979)
15. Mirkin, B.G.: Group choice. Wiley, New York (1979)
16. Nelson, L.S.: Some notes on averages. J. Quality Technology. **30**, 100–101 (1998)
17. Podinovskii, V.V.: Multicriterial problems with uniform equivalent criteria. USSR Comp. Math. Math. Physics. **15**, 47–60 (1975)
18. Podinovski, V.V.: On the use of importance information in MCDA problems with criteria measured on the first ordered metric scale. J. of Multi-Crit. Decis. Anal. **15**, 163–174 (2009)
19. Podinovski, V.V., Nelyubin, A.P.: Mean quantities: a multicriteria approach. Control Sciences. #. **5**, 3–16 (2020) (In Russian)
20. Podinovski, V.V., Nelyubin, A.P.: Mean quantities: a multicriteria approach. II. Control Sciences. **2**, 33–41 (2021) (In Russian)
21. Roberts, F.S.: Measurement Theory: with Applications to Decisionmaking, Utility, and Social Sciences (Encyclopedia of Mathematics and its Applications). Cambridge University Press, Cambridge (1984)
22. Smith, M.J.: Statistical Analysis. Handbook. A Comprehensive Handbook of Statistical: Concepts, Techniques and Software Tools. The Winchelsea Press, Edinburgh (2018)
23. Stevens, S.S.: On the theory of scales of measurement. Sci. New Series. **103**, 677–680 (1946)

Data and Text Interpretation in Social Media: Urban Planning Conflicts

Maria Pilgun and Nailia Gabdrakhmanova

1 Introduction

The article presents the results of the development of an algorithm for constructing mathematical and neural network models for the study of digital content generated by various actors, as well as for the rapid detection, prevention and resolution of urban conflicts, which are necessary for the effective management of urban systems.

The analysis of social tension in a metropolis is a vital task, and the speed of conflict detection is of great importance, as well as predictive analytics, which enables prediction of emerging conflict situations in order to develop effective measures to prevent them. It is the analysis of digital data generated by users that makes it possible to analyze the situation in real time and quickly detect social tension in the urban environment. Real-time data analysis is in demand in various fields: in the field of medical laboratory diagnostics [1]; for observations of coupled biological, chemical, and physical processes in the ocean from the macro to micro scale [2]; in real time intelligent systems [3]; for High-speed 3d railroad tie deflection mapping in real-time using an array of air-coupled non-contact transducers [4], etc. The objectives of the study were to develop and test an algorithm that includes the integration of neural network and mathematical models for analyzing digital content to identify semantic accents that characterize the dissatisfaction of residents, as well as to assess the positioning features of the project in the media space, segments of the greatest informational attention,

M. Pilgun (✉)
Russian State Social University, Moscow, Russia
e-mail: pilgunm@yandex.ru

N. Gabdrakhmanova
Peoples' Friendship University of Russia, Moscow, Russia
e-mail: gabd-nelli@yandex.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_18

social tension among residents of the metropolis during the urban planning project implementation, as well as to predict the development of the situation. In particular, the study was aimed at analyzing the dynamics of information activity on digital resources to track changes in the mood of the active citizens of the city and certain districts of Moscow involved in the discussion of the construction of the Metro Great Ring Line (Southern Section) (GRL), identify key content topics triggering the involvement of users in the discussion of the project and timely predict emerging and/or developing conflict situations. The analysis was conducted during the active stage of the GRL construction (Southern Section), which included the construction of three new GRL metro stations (Novatorskaya, Vorontsovskaya and Zyuzino), as well as the reconstruction of the Kakhovskaya station. Research questions were as follows: Will the results obtained with the use of neural network and mathematical models match? Does the proposed model make it possible to identify and analyze social tension in the metropolis, as well as predict the situation?

Sections of manuscript: Introduction. 2. Materials and Methods. 2.1. Text Analysis Methods. 2.2. Approaches to Mathematical Modeling. 3. Results. 3.1. General Description of the Content. 3.2. Neural Network Semantic Model 3.2.1. Content sentiment. 3.2.2. Key Negative Semantic Accents Presented in the User-Generated Content. 3.2.3. Social Tension Level. 3.3. Simulation Data. 4. Discussion. 5. Conclusions. Author Contributions. Funding. Institutional Review Board Statement. Informed Consent Statement. Data Availability Statement. Conflicts of Interest. References.

2 Materials and Methods

The material for the study was the verbal content generated by users on digital platforms, dedicated to the implementation of the GRL project (Southern Section), as well as digital footprints. The data were collected between October 1, 2020 and June 10, 2021 (Table 1).

Audience coverage is ensured mainly by microblogs, instant messengers, social net-works and videos (Fig. 1). Twitter, Telegram, VKontakte and YouTube are the top coverage-ensuring sources (Fig. 2).

Table 1 Quantitative characteristics of data

Parameter name	Values
Number of tokens	62 657 289e
Number of messages	7063
Maximum number of messages per day	614
Number of active authors	933
Activity (posts per author)	7.57
Number od sources	213

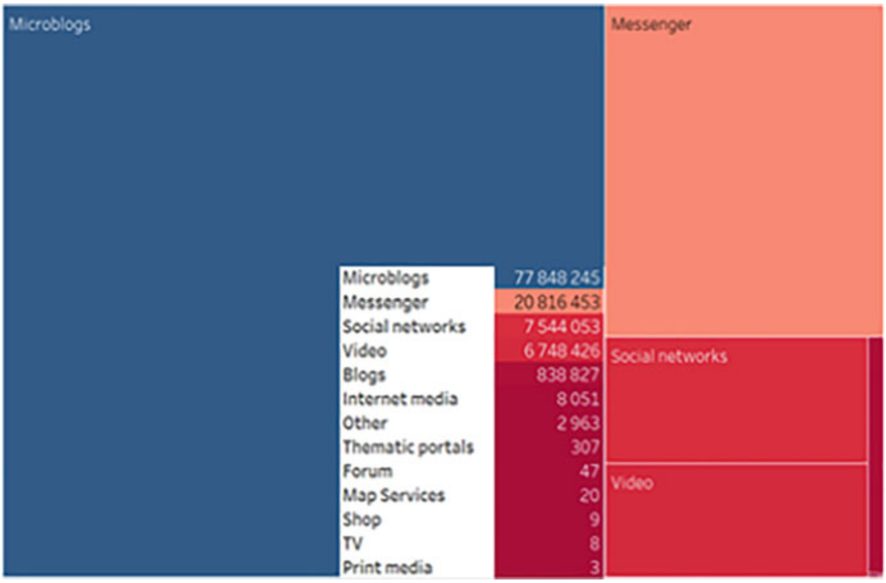


Fig. 1 Digital sources ranked by audience reach

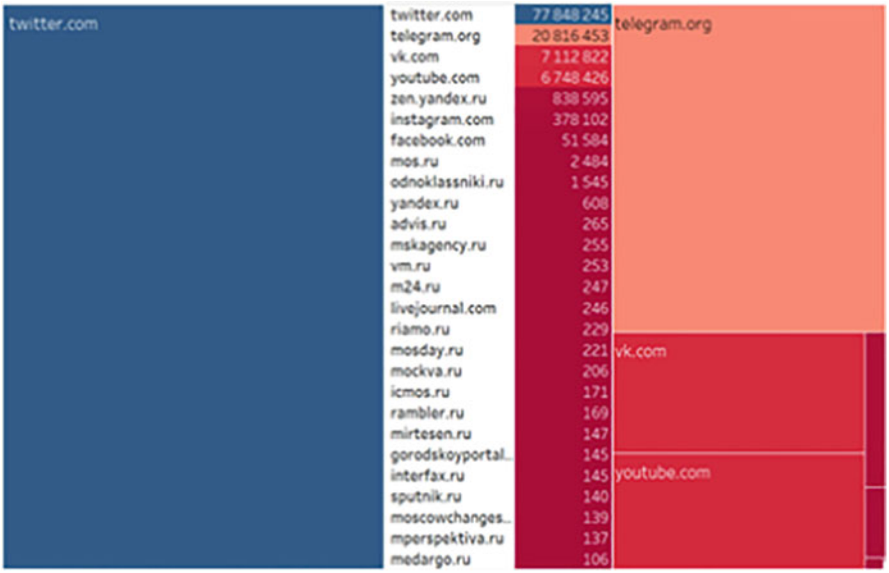


Fig. 2 Types of sources ranked by audience reach

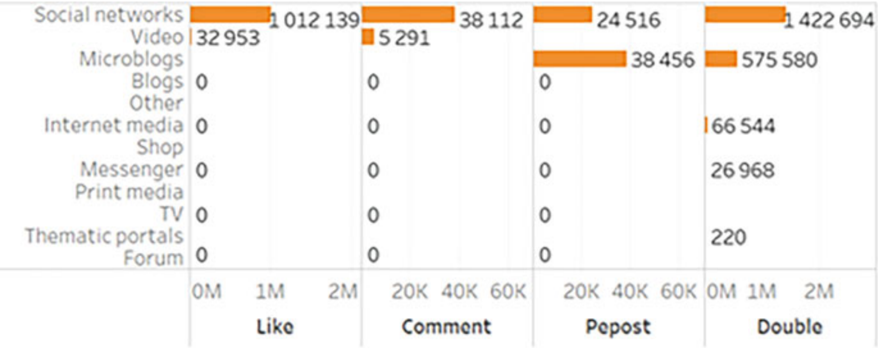


Fig. 3 Distribution of actors' digital footprints by types of sources

To generate content dedicated to the implementation of the GRL project (Southern Sec-tion), the actors preferred to use social networks, video hosting and microblogs (Fig. 3).

Among social networks, the undisputed leader was the VKontate platform. Instagram, YouTube and Facebook were also popular with users (Fig. 4).

2.1 *Text Analysis Methods*

The article presents a methodology for determining the perception of a certain situation by actors, which is implemented based on the results of the analysis of the content generated by users and their digital footprints. The study involved an interdisciplinary approach. To interpret the data, neural network text analysis, analysis of lexical associations using the TextAnalyst 2.3 technology, a detailed description of which is presented in [5, 6]; content analysis [7–9] using the AutoMap service; and Sentiment analysis using the Eureka Engine sentiment module were performed. For visual analytics, the Tableau platform was used.

2.2 *Approaches to Mathematical Modeling*

It follows from the objective formulated above that the mathematical model shall be dynamic. Dynamic mathematical models make it possible to analyze the situation and make predictions several steps ahead. In this article, the authors have developed several different types of mathematical models to study the potential of a conflict situation based on digital data. At the first stage, the problem was formalized and a time series was constructed based on pre-processed digital data. Currently, quite a lot of methods have been developed for the analysis of time series [10–13]. The

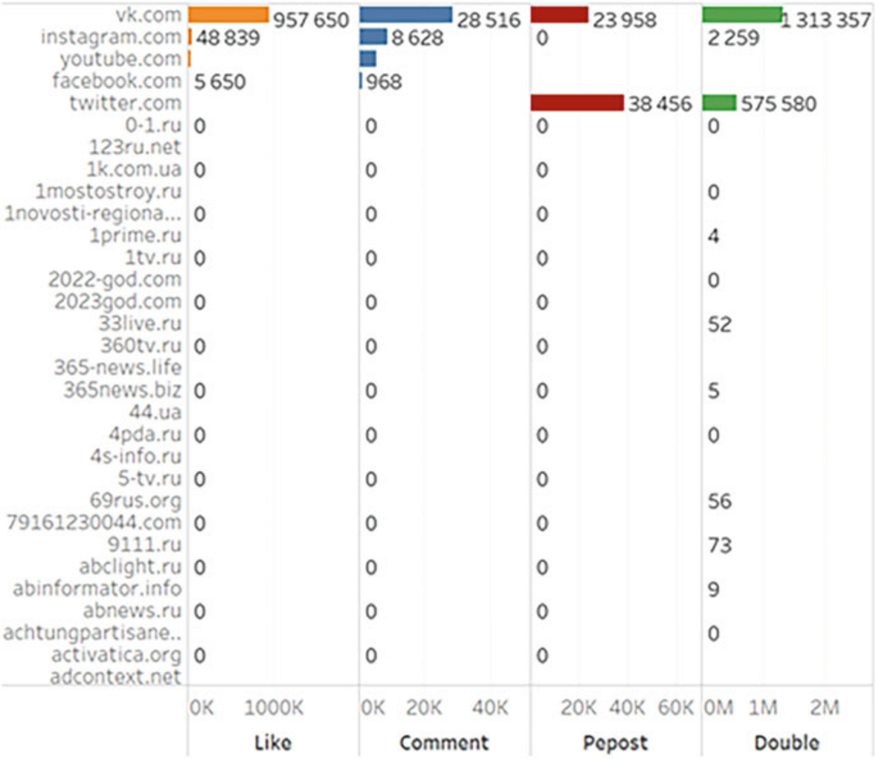


Fig. 4 Distribution of actors' digital footprints by sources

article presents the results of developing models according to observational data using deterministic and stochastic models; and a comparative analysis of models in terms of the prediction accuracy was conducted. When developing models, ordinary differential equations (ODEs) [14, 15], stochastic differential equations (SDE) [15], regression analysis [16] and convolutional neural networks (CNN) [17] were used.

3 Results

3.1 General Description of the Content

The dynamics of the total number of messages (Fig. 5) shows two growth peaks: on March 24, 2021 (2241) and April 1, 2021 (1699). A similar situation is shown by the dynamics of the number of unique messages on April 1, 2021 (958) (Fig. 6).

The peak of growth in the total number of messages on March 24, 2021 (2241) is deter-mined by the information that Muscovites are invited to vote for the name

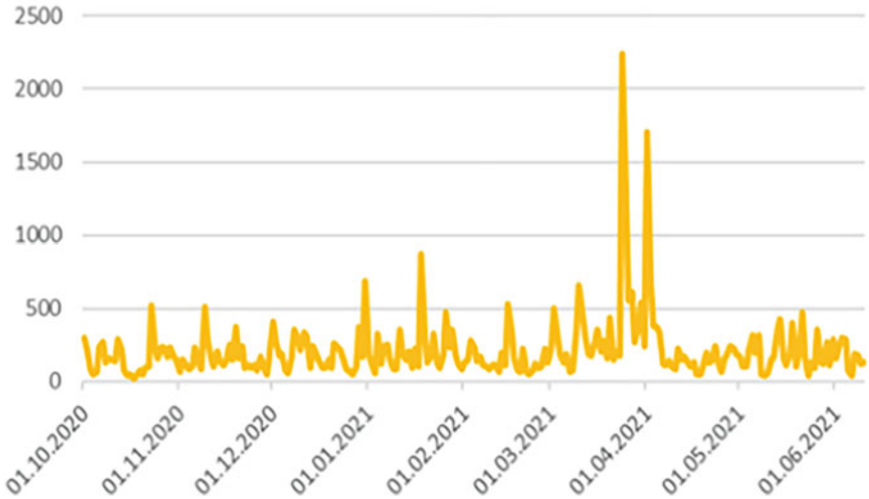


Fig. 5 Dynamics of the total number of messages

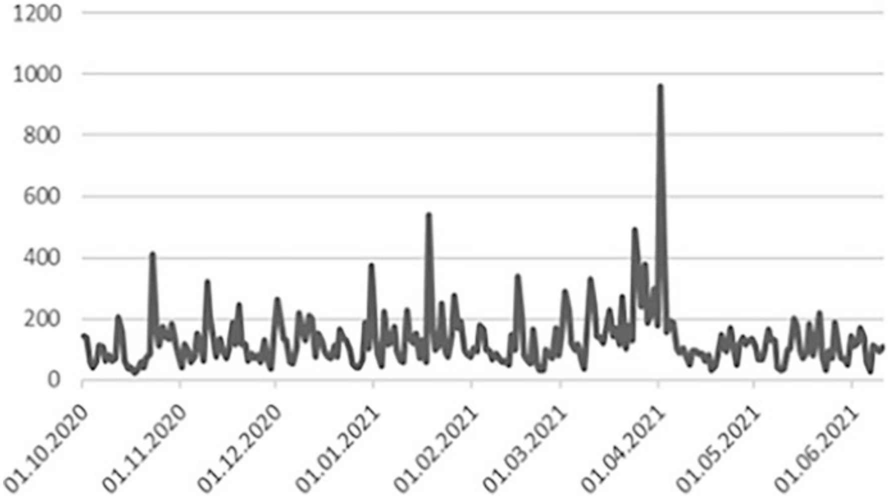


Fig. 6 Dynamics of the number of unique messages

for the new GRL station. The peak of the total number of messages (1699) and the number of unique messages (958) falls on April 1, 2021 and is associated with the opening of the Narodnoye Opolchenie and Mnevniki metro stations of the Great Ring Line. The peak of growth in the number of views falls on April 1, 2021 (11,050,725) (Fig. 7) and in the activity of authors on March 24, 2021 (1237); data for April 1, 2021 (599) (Fig. 8) shows similar results.

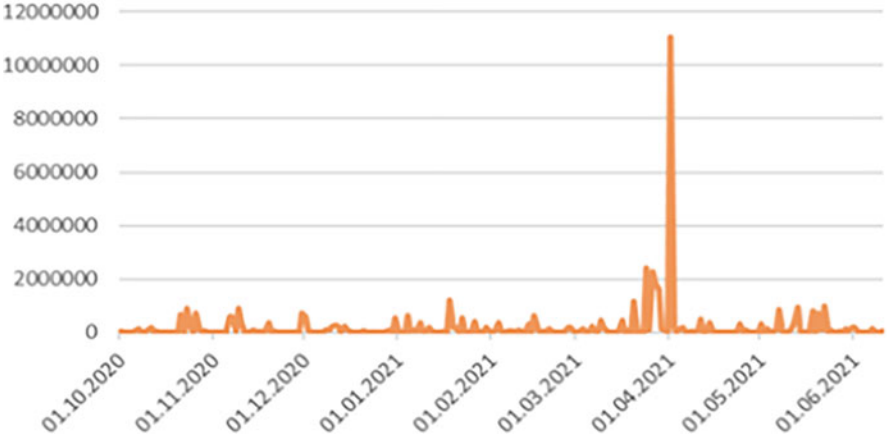


Fig. 7 Dynamics of the number of views

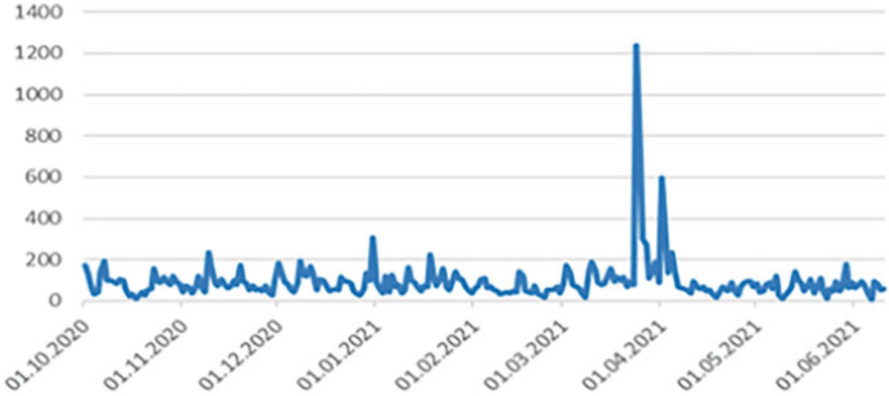
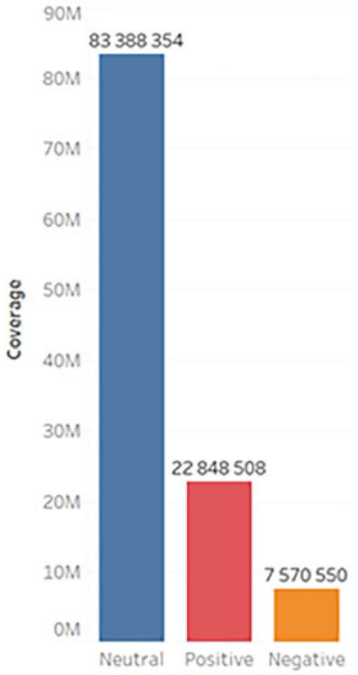


Fig. 8 Dynamics of the authors' activity

In the geolocation of authors' digital footprints, users on the territory of Russia logically prevail. Meanwhile, there are actors from different countries who are interested in the problems of building the GRL (Southern Section). Analysis of the actors' digital footprints geolocation by regions shows that the Central Federal District (CFD) represents the maxi-mum number of actors, and the actors of the Nizhny Novgorod Region and the North-Western Federal District (NWF D) are also highly active. Among Moscow actors, the most active are actors from Khoroshevo-Mnevniki, as well as residents of the following districts: Kosino-Ukhtomsky, Maryina Roshcha, Basmanny District, Sokolinaya Gora, Sokolniki, Zyuzino.

Fig. 9 Content sentiment by coverage



3.2 Neural Network Semantic Model

3.2.1 Content Sentiment

The sentiment feature of the vast majority of messages and digital footprints in the context of references to the project is neutral (Figs. 9 and 10).

Messages with negative and neutral posts were generated on the following platforms (in descending order): VKontakte, Telegram, Twitter; and positive messages were posted on VKontakte, as well as on mos.ru and Telegram. Data analysis showed that relevant content in 2021 increased significantly compared to 2020 (Fig. 11). Of course, it should be borne in mind that the large portion of content is gen-erated by biased media. Meanwhile, it is important to note that the number of negative messages increased by 2.1 times, neutral—by 2.2 times, and positive— by 3.1 times (Fig. 12).

The top sources ranked by the sentiment types are shown in Fig. 13, 14, and 15.

3.2.2 Key Negative Semantic Accents Presented in the User-Generated Content

- Complaints against project designers and builders who make town planning mistakes while designing and do not take the needs of citizens into account.

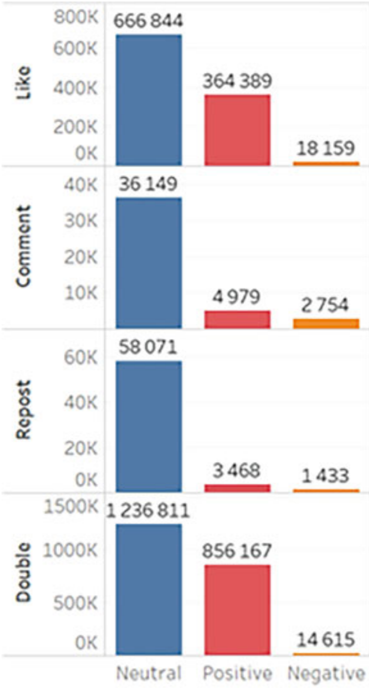


Fig. 10 Users' digital footprints sentiment

- Residents are outraged by the lack of dialogue with the authorities and the quality of public hearings.
- Poor location of new stations.
- Decreased living standards of residents, the emergence of a structural risk for residential buildings; residents demand to sign contracts for insurance of residential buildings against possible destruction and want to reduce fees for housing and utilities services or receive compensation for utility bills.
- Unsuccessful (according to residents) location of exits and the transition (removal) of surface pedestrian crossings.
- Increased noise levels in residential buildings during the construction; in particular, by-pass roads will be laid under the windows of the buildings.
- Construction works performed at night.
- Sidewalks become narrower; pedestrian areas are transferred to new places under the windows of residential buildings.
- Risk of technological catastrophes during the construction.
- Deterioration of the transport situation in the area, since the usual transport routes of the residents of the area will be blocked during the construction period.
- Destruction of the green zone of the city.
- Residents fear that the construction of new metro lines will significantly impair the movement of residents of older districts, as they will not be able to enter the metro cars, which will be overcrowded at the previous stations.

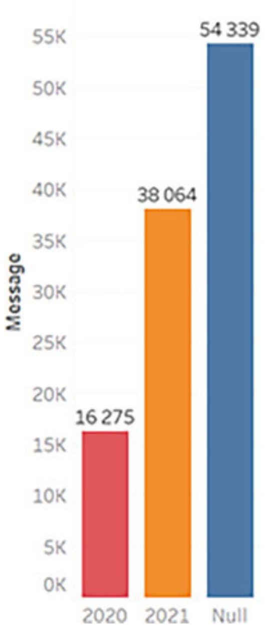


Fig. 11 Dynamics of the total number of messages

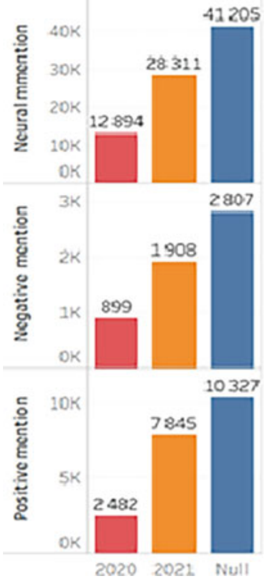


Fig. 12 Dynamics of messages with various sentiment types

- The construction will change the usual ground public transport routes.
- Complication of the system of transfers, inefficiency of transport connections, deterioration in the design of transfers in comparison with the previous stages of construction of the metro system:

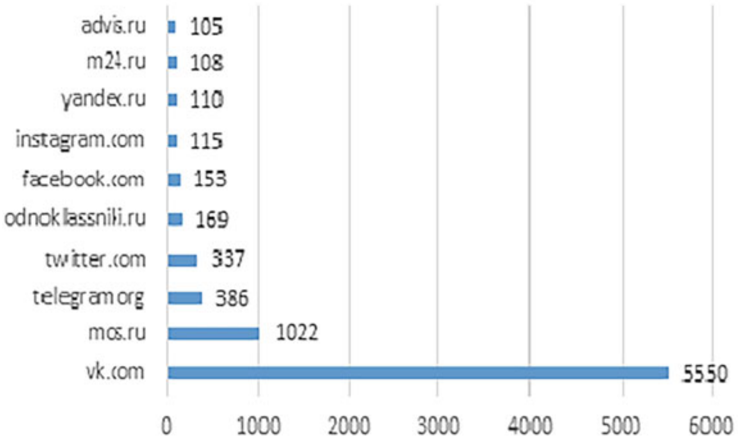


Fig. 13 Top-sources with a positive sentiment

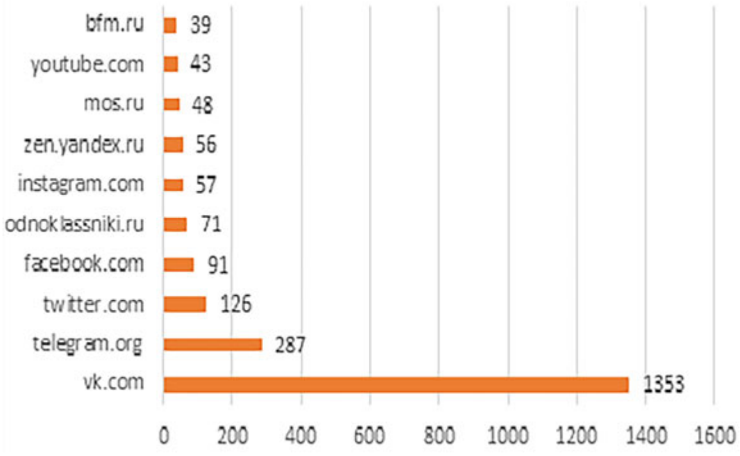


Fig. 14 Top-sources with a negative sentiment

- Low quality of design of new stations in comparison with the Soviet period of construction and foreign international experience:
- Pits excavated for the construction cause flooding.
- Elimination of parking lots for Muscovites' private vehicles.
- The needs of people with limited mobility, for whom the urban environment becomes unsurmountable, are not taken into account.
- Negative aspects of the GRL construction are superimposed on old conflicts between residents and builders:

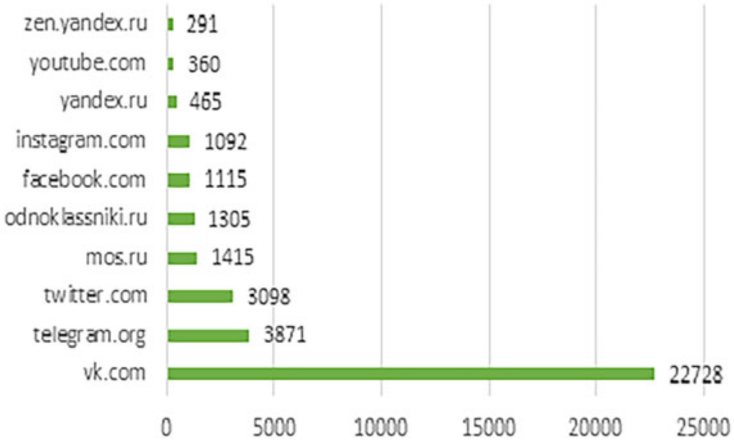


Fig. 15 Top-sources with a neutral sentiment

Table 2 Social stress and social well-being indices

Facility name	Social stress index	Social well-being indes
GRL (Southern Section)	3.39	12.75

- – confrontation between residents and builders from the Novatorov Street station to the Sevastopolsky Prospekt station.
- – residents' protests against the construction of the Biryulyovskaya metro line.
- – dissatisfaction of residents caused by frequent replacements of the pavements.

3.2.3 Social Tension Level

The result of the analysis of the consolidated database showed a low level of social stress of 5.39 and a moderate value of the social well-being index of 12.75 (Table 2)

Analysis of the data showed a low level of social tension caused by the GRL (Southern section) construction. In terms of the number of negative messages, the Vorontsovskaya station is slightly ahead of other stations; the minimum number of negative reactions is observed for the Novatorskaya station. In the positive cluster, the Kakhovskaya station is the leader; the minimum number of positive messages is related to the Novatorskaya station. In the neutral cluster, the Zyuzino station takes the first place; the minimum number of neutral messages is related to the construction of the Novatorskaya station (Fig. 16).

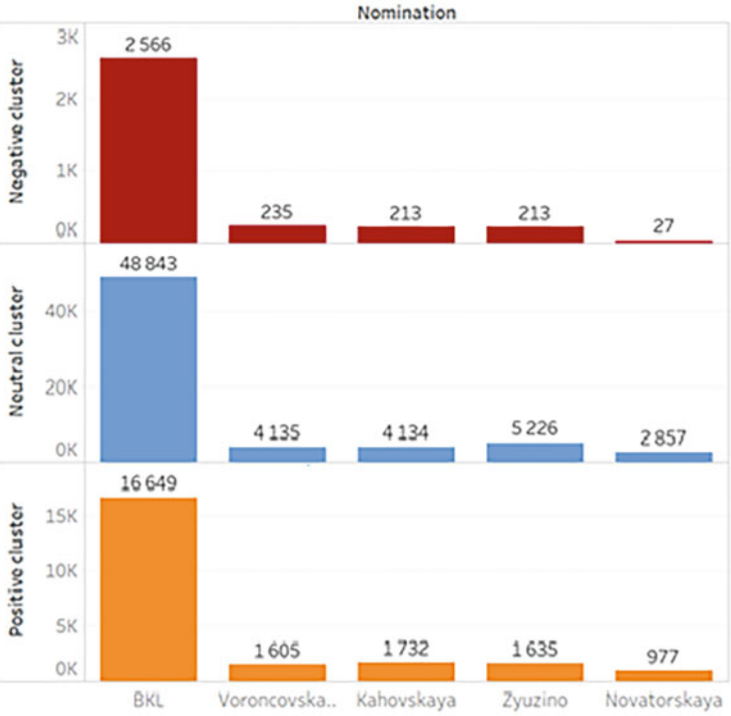


Fig. 16 Social tension rating

3.3 Simulation Data

The problem under consideration belongs to the class of problems that are difficult to formalize. Based on the above analysis of the situation according to digital data, the following formalization of the problem was adopted. The following groups have been introduced:

P is a group of actors with a positive attitude towards the project; pos(t) is the number of comments of the group P at the moment t,

N is a group of actors with a negative attitude towards the project; neg(t) is the number of comments of the group N at the moment t;

U is a group of actors with a neutral attitude towards the project; u(t) is the number of comments of the group U at the moment t.

In view of the problem under consideration, we assumed that the percentage of all three groups of the part society that is not involved in social networks is the same as for the part of the society that is involved. Due to the fact that we are primarily interested in the dynamics of the number of actors with a positive and negative attitude, and given that the activity of posts varies depending on the day of the week, holidays, etc. the following normalization is adopted.

Table 3 Fragment of the initial data

t	p(t)	n(t)
1	0.0537	0.0107
2	0.1024	0.0097
3	0.0357	0.0476
4	0.0217	0.1086
5	0.0645	0.0967
6	0.0845	0.0234

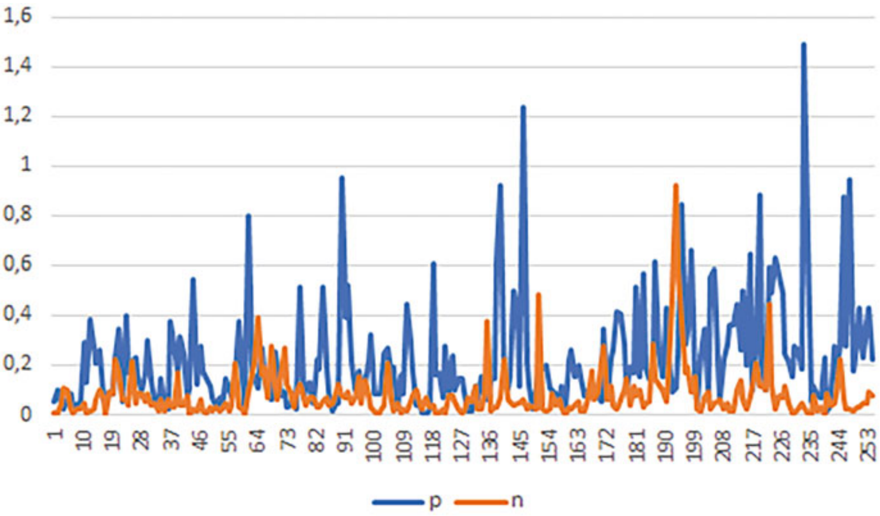


Fig. 17 Dynamics of the initial data p(t) and n(t)

Two new variables have been introduced: p(t)=pos(t)/u(t), n(t)=neg(t)/u(t).

Table 3 contains a fragment of the initial data for p(t), n(t). In Fig. 17, the dynamics of the initial data p(t) and n(t) is presented.

General Formulation

Time series {p(t)}, {n(t)} are given, where t=1,.., N, which characterize the level of positive and negative attitude towards the project, respectively. The sampling step is constant. It is necessary to build dynamic models for predicting and analyzing the emergence of a conflict situation according to these data.

Mathematical Models with Regression Analysis

The best known and widely used models for time series analysis are various autoregressive models. Trends of the time series {p(t)}, {n(t)} are built using regression analysis methods. As a result of estimating the model parameters, the following equations are obtained:

$$p(t) = 0,1 + 0,001t;\ n(t) = 0.06 + 0.0001t$$

The obtained estimates of the trend coefficients preceding t indicate that the series {p(t)} grows faster than the series {n(t)}.

Mathematical Models Based on Ordinary Differential Equations (ODEs)
To analyze the behavior of a dynamic system, it is important to have continuous mathematical models. The most suitable continuous models are differential equations. In this section, we present the results of building mathematical models based on ODEs. When choosing the general form of the differential equation, the problem under consideration can be represented as a competition problem: two-species struggle in populations, an arms race, military operations, etc. [10]. We have looked at various ODE models for describing competition. After comparing the results of the built models, systems of autonomous differential equations were chosen to describe the processes. The chosen equation is as follows:

$$\begin{cases} \frac{dp}{dt} = \alpha_1 \cdot n + \beta_1 \cdot p + \gamma_1 \\ \frac{dn}{dt} = \alpha_2 \cdot n + \beta_2 \cdot p + \gamma_2 \end{cases} \tag{1}$$

Model parameter estimates (1) were found using the multiple regression method and the Nelder-Mead method. At the first step, using regression analysis, we found coefficient estimates using the regression analysis method for each equation (1) separately; then this solution was adjusted using the Nelder-Mead method for the system. The process of coefficient approximation resulted in the following values:

$\alpha_1 = 0, 07; \beta_1 = 0.8; \gamma_1 = -0.18; \alpha_2 = -0.05; \beta_2 = 0.5; \gamma_2 = -0.03$. For ODE models, it is important that the estimates of parameters α_1 and α_2 are within the following range: $0 < \alpha_1 < 1$; $0 < \alpha_2 < 1$.

The model obtained (1) was studied in the nOp plane in order to determine the qualitative behavior of the p(t),n(t) functions in time. Equation (1) has the following equilibrium position:

$$\begin{cases} \frac{dp}{dt} = 0 \\ \frac{dn}{dt} = 0 \end{cases}$$

The equilibrium values of n and p are obviously found from the following conditions:

$$\begin{cases} \alpha_1 \cdot n + \beta_1 \cdot p + \gamma_1 = 0 \\ \alpha_2 \cdot n + \beta_2 \cdot p + \gamma_2 = 0 \end{cases}$$

The following values of equilibrium states are calculated: $p^0 = 0.228, n^0 = 0.08$

The values of the equilibrium states match the estimates of the mathematical expectation of the p(t), n(t) time series. We used the result obtained when choosing a stochastic differential equation (SDE) model. This result can also indicate the adequacy of the built model. Analyzing the estimates of the coefficients for the system of differential equations (1), it can be noted that the series {p(t)} grows faster than the series {n(t)}.

Mathematical Models Based on Stochastic Differential Equations (SDEs)
To describe the evolution of systems with interacting elements, there are two approach-es: the construction of deterministic or stochastic models. Stochastic calculus is a powerful tool for describing many processes. A stochastic differential equation (SDE) is a differential equation in which one or more terms represent a stochastic process. The SDE solution is also a stochastic process. Unlike deterministic ones, stochastic models make it possible to take into account the probabilistic nature of birth-and-death processes, as well as the effects of the external environment, which cause random fluctuations in the model parameters. The most commonly used example of an SDE is an equation with a white noise term. For each time series, a mathematical model was built in the form of the Ornstein-Uhlenbeck process:

$$dp = \alpha_1 \cdot (\beta_1 - p(t))dt + \sigma_1 \cdot \delta W_1 \tag{2}$$

$$dn = \alpha_2 \cdot (\beta_2 - n(t))dt + \sigma_2 \cdot \delta W_2 \tag{3}$$

where σW is the Wiener process. To estimate the parameters of models (2) and (3), the Vasicek model construction algorithm was used [18]. As a result of building the model, the following parameters were obtained:

$$\alpha_1 = 0.79; \beta_1 = 0.23; \sigma_1 = 0.21; \alpha_2 = 0.46; \beta_2 = 0.08; \sigma_2 = 0.08$$

For SDE models, for an equilibrium to exist, the following condition must be met:

$$\beta_1 \cdot \beta_2 > \alpha_1 \cdot \alpha_2.$$

This condition is met for the built models.

Mathematical Model with CNNs
Traditionally designed for 2D image data, CNNs can be used to model univariate and multivariate time series prediction problems. In our problem, the data are considered in the form of a multivariate time series. The input of the CNN was data with an interval of 2. The choice of the size of the number of input time steps has an important impact on how much data will be used for training. An estimate of the size of the number of input time steps was obtained using estimates of time series autocorrelations. When training the neural net-work, the input of the CNN was the vector ([x(k−2),y(k−2)], ([x(k−1),y(k−1)])), the vector [x(k),y(k)] was taken as a response vector. The process was implemented in the Google Colaboratory environment (Google Research product used for code writing); the models were implemented and trained using machine learning libraries such as: tensorflow, keras, sklearn. Additionally, the mathematical libraries numpy, pandas and matplotlib were used for calculations and plotting. When training the CNN, the ADAMAX optimizer was used, and ReLU was used as an activation function. The following training results were obtained:

Table 4 Fragment of the simulation results

t	p	$\hat{p}$	e
1	0.179104	0.079104	0.1
2	0.290323	0.290323	3.33E−16
3	0.429688	0.36875	0.060938
4	0.233577	0.157664	0.075912
5	0.87218	0.5	0.37218
6	0.27897	0.1	0.17897

Table 5 Mean retro prediction error for all models

b	ARMA	ODE	SDE	CNN
p	0.04	0.1624	0.084	0.025
n	0.009	0.016	0.009	0.003

Epoch 98/100: 164/164-0s-loss: 0.0140-405ms/epoch-2ms/step; Epoch 99/100: 164/164-0s-loss: 0.0141-404ms/epoch-2ms/step; Epoch 100/100: 164/164-0s-loss: 0.0139-396ms/epoch-2ms/step; Train Score: 0.18 RMSE; Test Score: 0.30 RMSE.

Table 4 shows a fragment of the simulation results for p(t) on the test set. The following designations are agreed for the table: t for time, p for a real value, $\hat{p}$ for a model value, e for an absolute calculation error.

Comparative Analysis of the Models Obtained

To assess the quality of the mathematical models built, we used the average retro prediction error for 10 steps ahead for the time series {p(t)} and {n(t)}. The estimate of the mean error is found by the formula:

$$E = \frac{1}{n}\left(\sum_{i=1}^{n}(x_i - \hat{x_i})\right)^2$$

where x_i is the value of the time series at the i-th point of the retro prediction; $\hat{x_i}$ is the model value of the time series at the i-th point of the retro prediction. Table 5 shows the calculated values of the mean prediction error for all models.

It follows from the table that the models built using convolutional neural networks yield the smallest prediction error. However, the built analytical models are important for the analysis of the situation. Such models are models built on the basis of ODEs and SDEs. Comparing the average retro prediction errors of the ODE and SDE models, we can conclude that it is important to take into account the stochastic nature of the processes.

4 Discussion

The article presents a solution to the problem of identifying social tension using mathematical models based on the interpretation of digital data generated by various types of users. The task was divided into subtasks: (1) collect and filter data;

(2) build a neural network semantic model; (3) build a mathematical model; and (4) make a management decision. Studies undertaken in the course of building mathematical models have shown that the models for the problem posed must meet the following requirements. Mathematical models must have the properties of dynamism and adaptability; it is necessary to take in-to account the stochastic component. When solving the problem, four models of different types were built. The built models can be used as parallel models. However, in addition to this, each model has its own characteristics, which makes it possible to more accurately analyze and predict the situation in terms of a potential conflict. The smallest retro prediction error corresponds to convolutional neural networks. To analyze the situation for the presence of a conflict situation, SDEs and ODEs are more suitable.

5 Conclusions

The main (key) tool in the study undertaken is neural network models and differential equations. At the first stage, neural networks were used for text analysis and data filtering. At the next stage, on the basis of the obtained solutions, the problem was formalized and mathematically formulated. It is shown that it is possible to analyze and monitor the situation based on the forecast of time series using CNNs. As a result of solving the problem, the following specific conclusions were made, regarding this situation. The study showed that the integration of neural network and mathematical models for the analysis of digital content made it possible to identify real points of dissatisfaction and positioning of the project in the media space, segments of the greatest informational attention, social tension among residents of Moscow and its districts around the GRL (Southern Section) construction place, as well as to predict the development of the situation.

The level of social stress and digital aggression determined in this study showed how sensitive residents of Moscow and residents of construction areas are to the information activity and involvement of network users in the discussion of the GRL construction and ongoing construction work on the Southern sector: and also made it possible to draw conclusions about the level of social tension and approval of the current situation around the object of study. Certain potential risks of conflict with residents may arise during the construction of the Vorontsovskaya station. Meanwhile, data analysis and calculation of social stress and social well-being indices made it possible to predict the absence of strong opposition to the construction on the part of the residents of the district. The course of events confirmed the correctness of the results obtained.

Author Contributions: Conceptualization. N.G. and M.P.; methodology. N.G. and M.P.; software. N.G. and M.P.; validation. N.G. and M.P.; formal analysis. N.G.; investigation. M.P.; writ-ing—original draft preparation. M.P.; writing—review and editing. N.G.; visualization. M.P.; supervision. N.G. All authors have read and agreed to the published version of the manuscript. Funding: This research was funded by RFBR. Grant 19-08-000261. Institutional Review Board Statement:

Not applicable. Informed Consent Statement: Not applicable. Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: (ttps://vk.com) (https://www.livejournal.com/) (https://twitter.com). (https://www.youtube.com). Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Gressner, A.M., Arndt, T. (eds.): Lexikon der Medizinischen Laboratoriumsdiagnostik. Direct Analysis in Real Time. Springer Reference Medizin. Springer, Berlin (2019). Available at: https://doi.org/10.1007/978-3-662-48986-4-311065
2. Venkatesan, R., Tandon, A., D'Asaro, E., Atmanand, M.A. (eds.): Observing the Oceans in Real Time, 323 pp. Springer, Cham (2018)
3. Mizera-Pietraszko, J., Pichappan, P., Mohamed, L. (eds.) Lecture Notes in Real-Time Intelligent Systems, 526 pp. Springer, Cham (2019)
4. Datta, D., Hosseinzadeh, A.Z., Cui, R., di Scalea, F.L.: High-Speed 3D Railroad Tie Deflection Mapping in Real-Time Using an Array of Air-Coupled Non-contact Transducers, vol. 254. Springer, Cham (2023)
5. Kharlamov, A., Pilgun, M. (eds.): Neuroinformatics and Semantic Representations. Theory and Applications, 317 pp. Cambridge Scholars Publishing, Newcastle upon Tyne (2020)
6. Pilgun, M., Raskhodchikov, A.N., Koreneva Antonova, O.: Effects of Covid-19 on multilingual communication. Front. Psychol. **12** (2022)
7. Krippendorff, K.: Content Analysis. An Introduction to Its Methodology, 3rd. edn., 441 pp. SAGE Publications, Los Angeles (2012)
8. Mayring, P.: Qualitative content analysis. In: A Companion to Qualitative Research. No. 1, pp. 159–176 (2004)
9. White, M.D., Marsh, E.: Content Analysis: A Flexible Methodology. Library Trends. No. 1, pp. 22–45
10. Time series analysis. In: Schintler, L.A., McNeely, C.L. (eds.) Encyclopedia of Big Data. Springer, Cham (2022)
11. Nokeri, T.C.: Time-series analysis. In: Data Science Revealed. Apress, Berkeley (2021). https://doi.org/10.1007/978-1-4842-6870-4-3
12. Onder, I., Wei, W.: Time series analysis. In: Egger, R. (ed.) Applied Data Science in Tourism. Tourism on the Verge. Springer, Cham (2022). Available at: https://doi.org/10.1007/978-3-030-88389-8-22
13. Gabdrakhmanova, N., Fedin, V., Matsuta, B.: The modeling of forecasting new situations in the dynamics of the economic system on the example of several financial indicators. In: International Symposium Intelligent System, Moscow (2020)
14. Arnold, V.: Geometric methods in the theory of ordinary differential equations. In: MTSNMO, Moscow (2012)
15. Samarsky, A., Mikhailov, A.: Mathematical Modeling: Ideas. Methods. Examples. Fizmatlit, Moscow (2001)
16. Aivazyan, S.: Methods of Econometrics: Textbook. M., INFRA-M (2010)
17. Scholle, F.: Deep Learning in Python. SPb, Peter (2021)
18. Shiryaev, A.: Fundamentals of Stochastic Financial Mathematics. Fasis, Moscow (1998)

Visual Explainable Machine Learning for High-Stakes Decision-Making with Worst Case Estimates

Charles Recaido and Boris Kovalerchuk

1 Introduction

1.1 Motivation and Approach Overview

Many Machine Learning (ML) models are complex *black boxes* for the end users, which creates difficulties for trusted decision making using them [11, 15] due to various and often hidden assumptions made in the model discovery process. Decision-making can be a life critical or high-risk process. One dataset this work considers is the Wisconsin Breast Cancer (WBC) dataset [6, 19]. This is a benchmark dataset used to classify tumors as benign or malignant. A misdiagnosis of a malignant tumor as benign could be a fatal decision for a patient [19]. Other high-risk applications include self-driving cars, missile launches, certain investment strategies and others. A large part of the current impressive successes of deep learning models is not in the high-risk decisions. For many high-risk decisions, the user still reluctant to rely on machine learning models due multiple reason including the *lack of model interpretability* and *uncertainty on model accuracy*.

ML algorithms often rely on *random splitting* available data into training and validation data. The accuracy of each conducted split as well as the *average model accuracy* for splits like ten-fold cross validation can be high and considered appropriate dependent on application. However, the accuracy of the *worst-case* split can be significantly lower than the average, where most difficult cases are put to the validation set to test the model in the most stressful situations. For life-critical or high-risk decision-making, models with lower average accuracy among many

C. Recaido · B. Kovalerchuk (✉)
Department of Computer Science, Central Washington University, Ellensburg, WA, USA
e-mail: Charles.Recaido@cwu.edu; Boris.Kovalerchuk@cwu.edu

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_19

random splits but higher accuracy using the estimate of *worst-case* split may be preferable [9]. This approach is known as a *minmax strategy* based on the *Shannon function* [9] with the goal of finding the model that produces the highest accuracy on most difficult validation data. While finding the exact solution of this problem is computationally challenging the visual knowledge discovery approach allows successfully to find an *upper estimate* of it, as this chapter shows below.

This work considers two main problems: (1) the **interpretability** of ML models where many advanced ML models are difficult to understand, respectively called *black boxes*, and (2) the **reliability of accuracy** of ML models, where the model accuracy can be exaggerated, which is unacceptable in applications with high cost of individual errors.

Many techniques to increase human interpretability of machine learning models exist including data visualization, e.g., [15–18, 24]. The combination of data visualization and machine learning can provide self-service models for the end-user [18]. Self-service visualization models allow an end-user to apply their domain knowledge to tweak the model rules for better decision-making. Human interpretability also provides benefits in data transparency, data fairness, and the development of new models.

Multiple approaches can be found in the literature to address model interpretability and reliability of accuracy problems. This chapter follows the approaches presented in [9, 11] based on the visual means. Practically all interpretability studies one way or another use visualization, but mostly to present the results of model explainability studies. The approach that we follow in this chapter has a fundamental advantage of having *lossless visual means* as a major part of both ML model discovery and explanation. It provides a basis for a *self-service model* discovery and interpretation by the end-user/domain expert.

Unfortunately, not every known visualization technique fit well this goal. Machine learning data are fundamentally multidimensional, but popular existing methods to visualize multidimensional data like principal component analysis (PCA), multidimensional scaling (MDS), t-distributed stochastic neighbor embedding (t-SNE) and RadVis are lossy one way or another in representing multidimensional data. It means that attempts to discover multidimensional pattern in these visualizations have a fundamental flaw. In addition, artificially created new coordinates like PCA and t-SNE components do not have natural domain interpretations and can be viewed as black-box *dimensionality reduction* (DR) techniques.

The patterns which exist in n-D space can be lost or corrupted in the process of conversion of n-D data to these visualizations in 2-D/3-D before we will try to discover patterns in these visualization spaces. For instance, two different n-D points can be mapped to a single point in the visualization space due to much smaller size of 2-D/3-D space than n-D space. It is well illustrated by the *Johnson-Lindenstrauss lemma* [22].

The corruption of n-D patterns is illustrated in Fig. 1 for t-SNE for Wisconsin Breast Cancer (WBC) data. Each n-D rectangle (hyperblock, HB) is a continuous part of the n-D space, but in t-SNE some of those hyperblocks are not continuous.

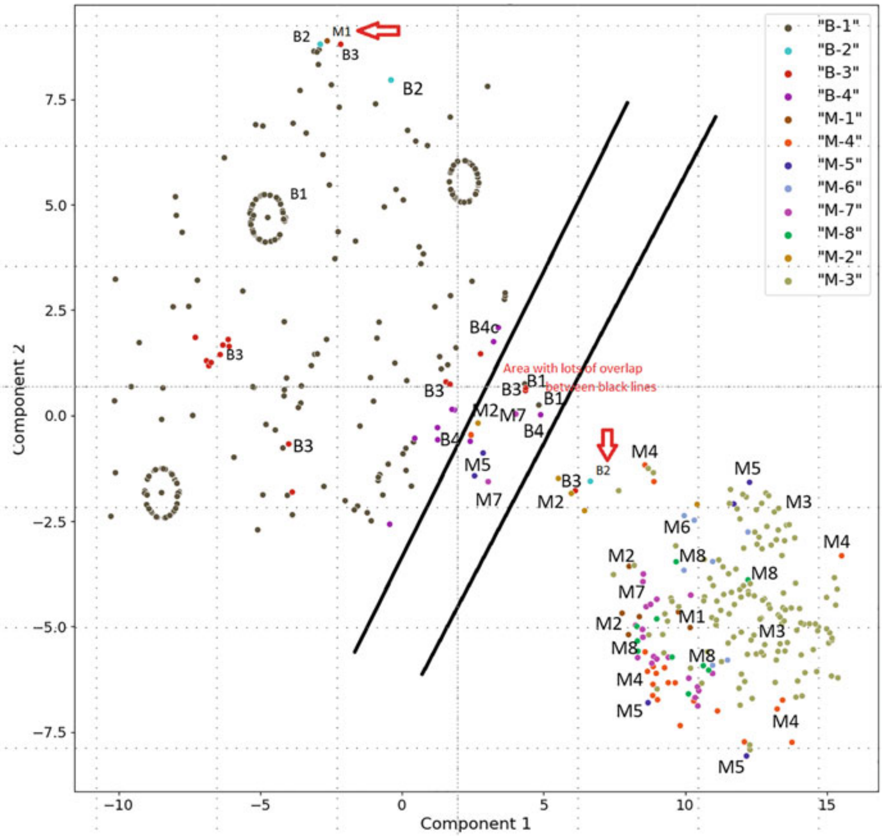


Fig. 1 Corruption of interpretable patterns of 9-D points in t-SNE in WBC data

Some cases of the same hyperblock are far away in t-SNE visualization with multiple cases of other hyperblocks in between.

Next, all n-D points in the hyperblock are within the distances d_1,d_2,. . . ,d_n from the center n-D point of the HB in the respective coordinates. This is an interpretable similarity in contrast with, say, similarity in Euclidian distance often used in multiple kernels, that can be non-interpretable for heterogenous data with heterogeneous attributes like, blood pressure, pulse, temperature and so on. Fig. 1 shows 648 9-D cases mapped to the first two t-SNE components with 8 malignant hyperblocks and 4 benign hyperblocks. The red arrow on the top points to the blue case from benign HB B2 and the red arrow in the center of this picture points to another blue case from the same benign hyperblock B2, which are far away from the first blue case with multiple cases on other HBs in between. This picture shows other cases with the same issue. It also shows multiple cases from benign and malignant classes that are near each other. They are good candidates to be checked for the worst-case validation set. See cases between two black lines as an example.

Therefore, we rely on General Line Coordinates (GLC), which losslessly visualize n-D data with unique graph constructing algorithms [10]. Generally, multidimensional GLC are mapped to graphs in 3-D/2-D space to enhance visual aids. In this chapter to form a powerful tool with high level visual clarity GLC we combine them with non-overlapping hyperblocks. The main advantage of HBs over other methods like many kernels is that HBs *are interpretable*, and all samples of the dataset can be grouped forming pure HBs known as *atomic* HBs [12]. Therefore, complex datasets can be considered unions of the atomic HBs.

While PCA, t-SNE and other lossy methods can corrupt data as was shown above for t-SNE, if the corruption is minor or can be controlled, these methods are very useful. t-SNE is a non-linear unsupervised dimensionality reduction technique that has been applied to high dimensional datasets such as MNIST handwritten digits (784 dimensions) [3] and genomic sampling [8]. Therefore, we explore capabilities of lossy methods in this chapter in combination with General Line Coordinates for visual class separation.

This work explores a new visualization method through the combination of a novel General Line Coordinates system called Dynamic Scaffolding Coordinates (DSC) along with hyperblocks, and dimensional reduction techniques such as principal component analysis and t-distributed stochastic neighbor embedding. Visualizations of high clarity are produced for multiple benchmark datasets where *worst-case* data splits can be discovered that impact model performance. These new methods are called **Dynamic Scaffolding Coordinates** (**DSC1**, and **DSC2**). DSC1 has basis in parallel coordinates whilst DSC2 has basis in Shitted Paired Coordinates [10]. DSC1 and DSC2 are lossless data visualization methods, which compact multiple input coordinates to only two axes.

One immediate consequence of our visual approach is that large datasets hamper visual knowledge discovery (VKD) due to line occlusion and permutation complexity [10]. We adjusted our dynamic scaffolding approach by incorporating hyperblocks and/or methods like PCA and t-SNE to reduce the number of lines and lay a foundation for our dynamic scaffolds to grow from.

We packaged DSC1 and DSC2 into a software toolkit DSCViz [25]. DSCViz grants the user interactive capabilities such as changing attribute order, emphasizing or deemphasizing classes and/or attributes, visualizing the coordinate system via polylines, markers, or both, zoom and drag capabilities, and clipping/bounding algorithms for sample selection. In addition, DSCViz includes parallel coordinate and shifted paired coordinate plots with the same functionalities listed above. All functionalities are purposefully built to enhance visual knowledge discovery for the end user. DSCViz was developed using Python as the default programming language, QTCreator for the GUI interface, and OpenGL for plot rendering.

We applied the DSCViz software to a real problem, which is using visual knowledge discovery to find the upper estimate of the worst dataset split to training and validation sets (also referred to as the most difficult or **worst-case split**) and/or reduce false predictions to improve critical decision-making such as tumor diagnosis. This is achieved by analyzing visually the DSC1 and DSC2 plots for regions of class overlap and selecting these samples for a validation set. This validation set is expected to reduce general model accuracy when compared to standard k-fold cross validation. One challenge of this VKD approach we found was that not all overlapped regions are equal when discriminating classes from each other, some overlapped regions practically do not impact model performance whereas other overlapped regions have a great impact when used in the validation set.

Several *worst-case* split experiments were conducted on benchmark datasets such as Seeds, Iris, and Wisconsin Breast Cancer [6]. We further investigated our visualization tool on very large datasets such as MNIST handwritten digits [3].

1.2 Forms of General Line Coordinates

There are many forms of GLC visualizations which are dependent on the graph constructing algorithm. GLC generalize multidimensional coordinate systems with multiple axes such as parallel coordinates (PC), radial (star) coordinates, shifted paired coordinates (SPC) and others, by allowing more flexibility in location of coordinates [10]. This work uses PC and SPC to construct two new GLC visualizations on a 2-D plane.

Parallel line coordinates (PC) represent multidimensional data using several one-dimensional axes. Parallel line coordinates have been used extensively for visual knowledge discovery and decision-making [21]. Each axis in a PC plot is independent and represents the informational space of one attribute. For instance, a 4-D point $\mathbf{x} = (1,3,2,5)$ is represented by a polyline (directed graph) with 4 nodes located on respective 4 vertical axes at the respective positions (1,3,2,5). Figure 2a shows the Iris dataset on a parallel coordinate plot. Each axis is responsible for one of four attributes: petal width, petal length, sepal width, and sepal length. The minimum and maximum values of each axis correspond to the range of each attribute. Increasing the range of one attribute will have no effect on the other attributes. Figure 2a shows a clear classification area for the red class on the bottom right-hand corner across two attributes. Any new samples that would plot in the bottom right-hand corner would be classified as red class. Establishing a rule to classify blue and green cases from each other is more difficult as they exhibit overlap on all four attributes. Figure 2b shows the Iris dataset in another form of General Line Coordinates.

Shifted paired coordinates represent multidimensional data using several two-dimensional axes [10, 18]. Each two-dimensional axis holds the informational space of two attributes. SPC show relationships between pairs of attributes. For instance, a 4-D point $\mathbf{x} = (1,3,2,5)$ is represented by a polyline (directed graph) which connects node/point (1,3) of the first pair of coordinates in one Cartesian plot with node/point (2,5) of the second pair of coordinates in another Cartesian plot. This leads to a limitation of the SPC plot in which an even number of attributes are required. A dataset with an odd number of attributes will require engineering, duplicating,

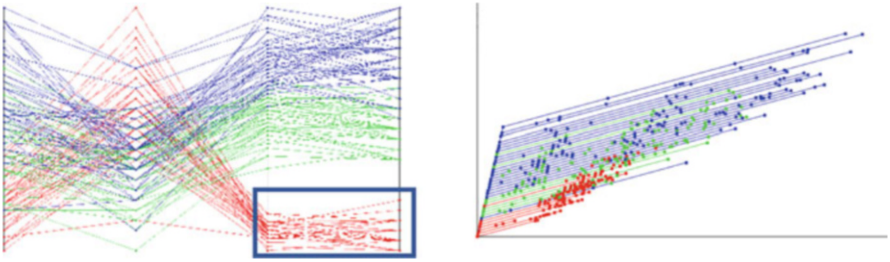


Fig. 2 Transformation of a 4-D PC plot to a 2-D GLC plot. (**a**) Iris dataset in Parallel Coordinates. (**b**) Iris dataset in one form of General Line Coordinates

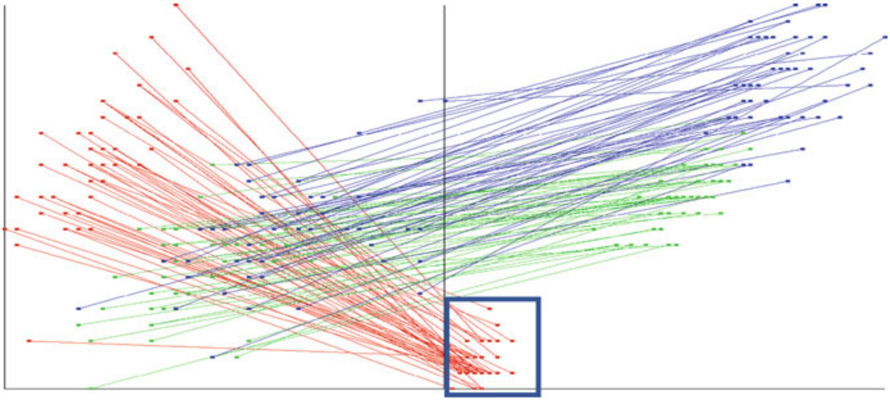


Fig. 3 Iris dataset on a shifted paired coordinate plot

or removing an attribute. Duplicating of an attribute is most desirable because it preserves information of the dataset.

Figure 3 shows the lossless Iris dataset on a shifted paired coordinates plot. The first attribute-pair shows a relationship between sepal width (vertical) and sepal length (horizontal), and the second attribute-pair shows a relationship between petal width (vertical) and petal length (horizontal). Separation of the red class is shown in the blue rectangle. Green and blue classes are *highly overlapped*. The order of attributes is important when making a SPC plot due to the unique relationship between any two attributes, which leads to visually different plots. This introduces a new hurdle in developing multidimensional visualizations, which is the exponential time complexity of plotting every attribute order permutation and choosing the best one.

Figure 4 is a visually appealing representation of class separation of the Iris dataset in SPC constructed using an alternative pairing of the same 4 coordinates. The red class is completely separated from the red and green class. In Fig. 3 the green class and blue class had large amounts of overlap, but for this permutation of attributes the blue and green classes overlap in a *much smaller area*. The difference

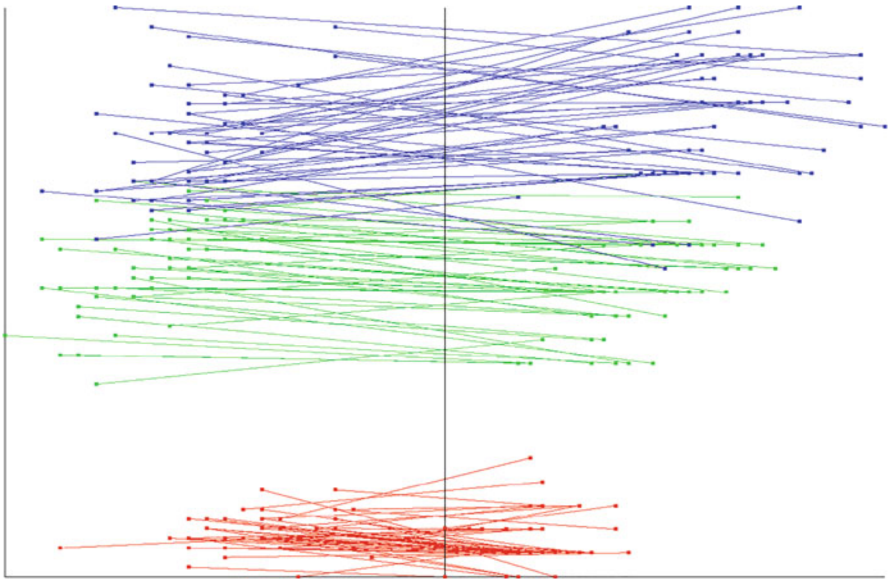


Fig. 4 Iris dataset on SPC with high visual classification clarity

between Figs. 3 and 4 is quite drastic in readability to the user. The time to produce every attribute permutation of the Iris dataset and choose the plot for Fig. 4 was only a few seconds, however, computing permutations of larger multidimensional datasets may not be feasible.

A **hyperblock** (HB) is a multidimensional "rectangle" (n-orthotope) with set of multidimensional points $\{\boldsymbol{x} = (x_1, x_2, \ldots, x_n)\}$ with center $\boldsymbol{c} = (c_1, c_2, \ldots, c_n)$ and side lengths $\boldsymbol{L} = (L_1, L_2, \ldots, L_n)$ [12, 20] such that,

$$\forall i \in N, \mid x_i - c_i \mid \leq \frac{L_i}{2} \tag{1}$$

If a HB has equal length sides, then it is known as a *hypercube*. HBs can represent a group of samples with similar attribute values and can be used to condense the number of lines needed for visualization. HBs are highly interpretable data units as a combination of individual coordinates without imposing non-interpretable operations between coordinates. This allows for the separation of different data units that exist among multiple attributes during model creation. In contrast, some other kernels known in machine learning are not interpretable. For example, in k-nearest neighbors, a Euclidean distance search requires a summation of squared differences for all attributes and computing square roots of them. This may have no meaning in the domain, like cancer diagnostics, and not appropriate to the domain expert. In a previous study hyperblocks were used to generalize decision tree rules [12]. Class

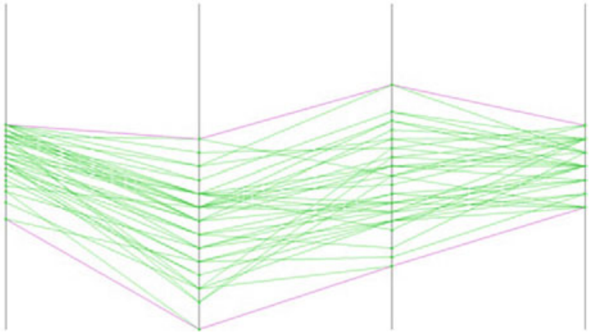


Fig. 5 Hyperblock in 4-D space with boundary lines (pink) on a PC plot

purity ratings are given to HBs, with the purest HBs containing samples from only one class, and the *least pure* HBs contain equal number of samples from all classes.

Figure 5 shows a HB in 4-D space. The boundary lines (pink) contain several green samples. The green lines can be omitted, and the pink boundary lines will still contain the information space of all those samples. Methods like principal component analysis can produce HBs but those HBs can be non-interpretable. Non-overlapping interpretable hyperblocks from a specific dataset can be found by using decision trees, Merger Hyperblock algorithm [12], and other methods. For this research we considered the decision tree method for creating hyperblocks due to the smaller time complexity of running a decision tree compared to Merger Hyperblock algorithm.

1.3 Related Studies

In this section we discuss prior studies involving General Line Coordinates systems and other approaches like Local Interpretable Model-agnostic Explanations (LIME) and their applications.

The LIME method was proposed as means to approximate less interpretable models such as neural networks [14, 23]. It approximates a learning model at the local level by performing perturbation around a particular input and mapping the output through linear models. LIME has been used in many applications from natural language processing [4] to computer vision [14]. However, LIME and similar methods have a *limited interpretability* for heterogeneous data [11] in contrast with decision trees and logical models, which we exploit in this study with methods based on the GLC.

For GLC these studies include interactive visual knowledge discovery in shifted paired coordinates, Pareto optimization in GLC, GLC-L coordinate system, and more [10]. In [18] interpretability of machine learning models was increased by developing interactive shifted paired coordinates system in the SPCVis software. SPVis introduces a plethora of features that adapt the shifted paired coordinate system such as non-linear scaling, non-orthogonal displays, and serpent parallel coordinates. In addition, in [18] a genetic algorithm and coordinate order optimizer are used to find strong attribute permutations that led to class separability, which is especially important for high-dimensional datasets. The SPCVis optimization algorithms allow to produce a multi-stage visual classifier. Accuracy results using SPCVis on Seeds, Wisconsin Breast Cancer, and Iris datasets are comparable to ML algorithms from the literature. Our study also uses SPC, not for discovering classification models directly, but by using SPC to design new lossless visualizations DSC2.

In [5, 10] General Line Coordinates are used to transform non-image data (vectors filled with general numeric attribute information) into an image. One important factor of it is that GLC is a **lossless** projection from multidimensional data into two dimensions, and the produced image contains all the original information of the dataset. This approach allowed to apply to any dataset that is not image such as Wisconsin Breast Cancer dataset and input that data into a Convolutional Neural Network. While our research uses a GLC system with dynamic scaffolding, [5] used an alternative GLC system referred to as GLC-L where angles are calculated by optimizing a linear discrimination function. There is a close link between DCS1 and GLC-L in design of the visualization, but with a significant difference in their use to discover visual patterns.

The first difference between GLC-L and DSC2 is in the ways of assigning angles and ordering of attributes. In GLC-L the order of the attributes (segments of the polyline) is not critical, but in scaffolding it is optimized. In DSC we derive the angles and order of attributes from external sources like location of values of the attributes in parallel coordinates, shifted paired coordinates, decision trees and other possible sources. It is based on the idea that if some attributes separate classes better than other in external sources, then it can be exploited to assign angles and order of attributes in DSC to improve visualization. Next, DSC allows adding/engineering additional attributes to improve visualization too.

In [1, 10] GLC-L and Pareto optimization are used to find a best-case scenario in certain datasets like student performance, and local weather. As an example, one might want to know which month is best to go hiking, or which pre-major classes lead to better students in preparation for getting accepted into a Computer Science program. It is demonstrated in [1, 10] how to pin down key attributes that would form a basis for the Pareto subsets, and those subsets would lead to a Pareto Frontier. This work was able to condense several parallel coordinate plots into a single GLC-L plot. This work also uses GLC-L visualization method, which is similar DSC methods we develop in this study, but for setting up priorities among alternatives in the Pareto set, not for discovering classification models.

In [12] Parallel Coordinates are used as a basis for discovering interpretable classification rules. In this work we propose to use the parallel coordinates to design an alternative dynamic visualization DSC1 for better visual discrimination of classes. Parallel coordinates are static, where the location of the next coordinate does not depend on the location of the previous coordinates. The dependence using in DSC allows to capture more complex patters than in static parallel coordinates.

2 Dynamic Scaffold Coordinates with Hyperblocks

2.1 Dynamic Scaffolding Coordinates Based on Parallel Coordinates

Dynamic Scaffolding Coordinates based on Parallel Coordinates (DSC1) generalize the parallel coordinate plot by creating a series of origin-to-attribute scaffolds. Each attribute axis is given a certain angle and the scaffolds connected tip-to-tail to form a polyline to represent losslessly n-D data. The axis tilt can be user-defined or found analytically through optimization. The axis tilt is required to better visualize data trends across two dimensions. Without the axis tilt the line components would stack vertically in one dimension.

The DSC1 graph construction algorithm (Fig. 6) as follows:

1. Set up dataset sample coordinates in the same manner as a Parallel Coordinates plot.
2. Apply a rotation transformation for each individual attribute axis with pre-defined angles.
3. Create a vector from the origin to the attribute point for each attribute. Each of these vectors is called a **scaffold**. The scaffolds are created for all samples.
4. The first attribute scaffold position is left untouched; however, the tail of the first attribute scaffold is removed, making the tips of the first attribute the "origin" of the polyline.
5. Translate the remaining scaffolds to the tip of the preceding scaffold.

Mathematically this process can be described as follows. Consider n-D point $X = (x_1,x_2,..,x_n)$, then we produce rotated vectors $\mathbf{x}_1,\mathbf{x}_2,..,\mathbf{x}_n$ from X, such as shown in Fig. 6b, with lengths equal to respective values of x_i, $||\mathbf{x}_\mathrm{i}|| = x_i$. $i = 1{:}n$. Then we add these vectors to the origin point O: $V_n = O + \mathbf{x}_1 + \mathbf{x}_2 + \ \ldots \ + \mathbf{x}_n$ to produce a *directed graph* like shown in Fig. 6c. Here, in contrast with classical summation of vectors with the n-D point, which will produce a single n-D point V_n, we preserve results of all consecutive sums $V_i = O + \mathbf{x}_1 + \mathbf{x}_2 + \ \ldots \mathbf{x}_i$. These points ($V_1$,$V_2$, ..., V_n) are vertices of the directed graph like shown in Fig. 6c.

The angles in the DSC1 graph construction algorithm are chosen to visually show separation of classes that separate on one attribute called the *attribute of separation*. The attribute of separation is placed first in the order of attributes and given the steepest angle to emphasize its importance and the order for the remaining attributes sharing the same angle (Fig. 7b), however the possibility exists to change the angle of each attribute as shown in Fig. 8. The comparison of Fig. 7a with Iris data in parallel coordinates with the same data in DSC1 in Figs. 7b and 8 shows the

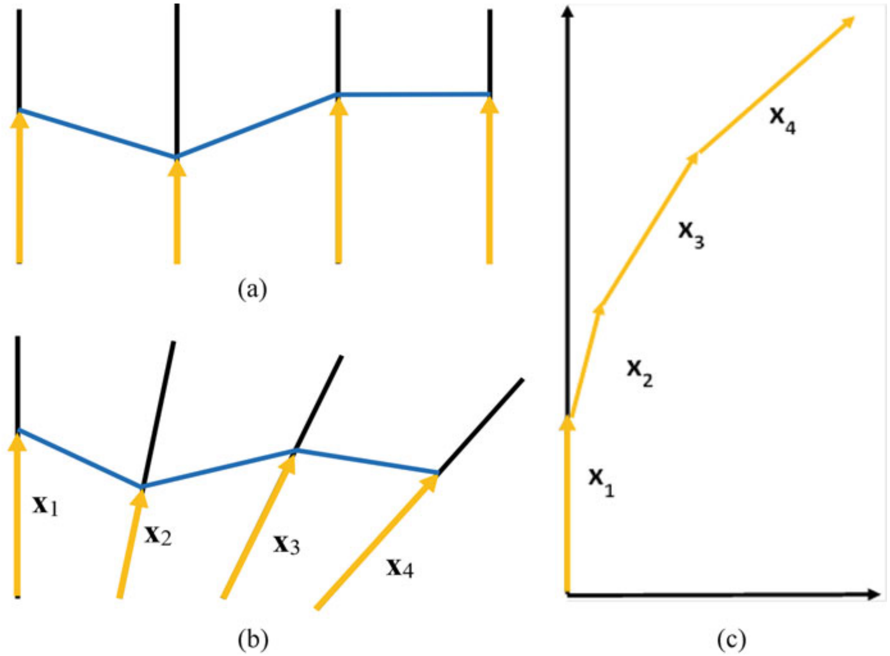


Fig. 6 Visual steps for construction of the DSC1 plot. (**a**) One sample on parallel coordinates. (**b**) Rotating the axes. (**c**) Rotating the axes

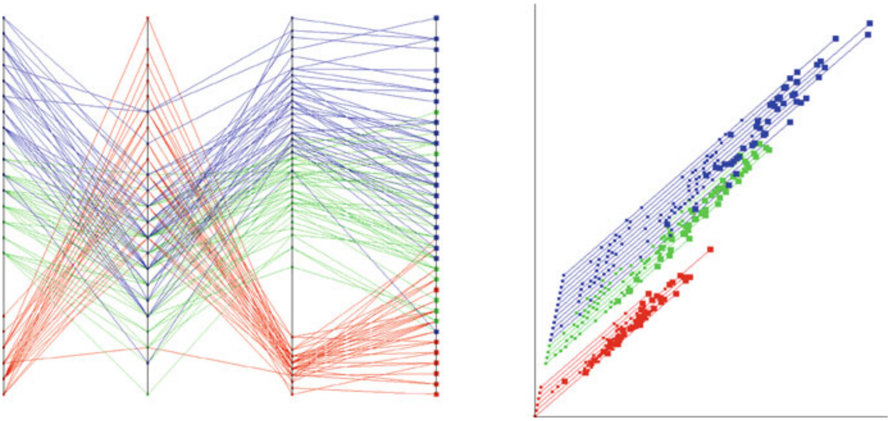


Fig. 7 Side-by-side comparison of Parallel Coordinates and DSC1 of Iris dataset. (**a**) Iris dataset on Parallel Coordinates. (**b**) Iris dataset on DSC1

pure advantages of DSC1 over the parallel coordinates for this dataset, while both methods losslessly represent all these four-dimensional data in 2-D visualization space.

Fig. 8 DSC1 with a different rotation for each attribute

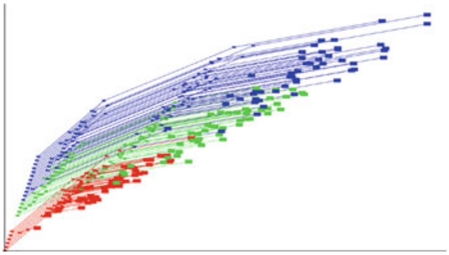
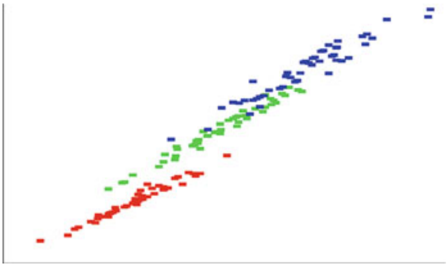


Fig. 9 Hiding polylines and certain attribute markers on DSC1



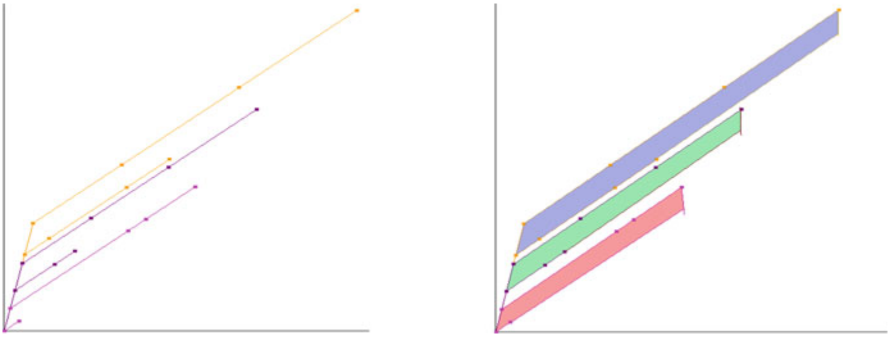


Fig. 10 DSC1 visualized using only HBs. (**a**) HB boundaries on DSC1. (**b**) Shaded HBs on DSC1

Other techniques may be applied to DSC1 such as hiding certain attributes markers and hiding the polylines. Figure 9 is another representation of Fig. 7b where the polylines are hidden as well as the first three attributes. These self-service techniques can be deployed by the end-user to highlight certain attributes or regions of the dataset that may be of interest.

Figures 7b, 8 and 9 clearly shows that blue and green classes overlap. We can find subsets of those classes in the form of hyperblocks, which do not overlap as Fig. 10 shows. Next, DSC1 software grants the user the ability to reduce graphs of samples to an upper and lower hyperblock boundary line as shown in Fig. 10. This alternative visualization reduces line occlusion, which can enhance visual knowledge discovery in sample dense but highly separable datasets.

Dynamic scaffolding coordinates can be used to create visual classifiers. Particularly in DSC1 this can be done via a series of DSC1 plots using graphical linear

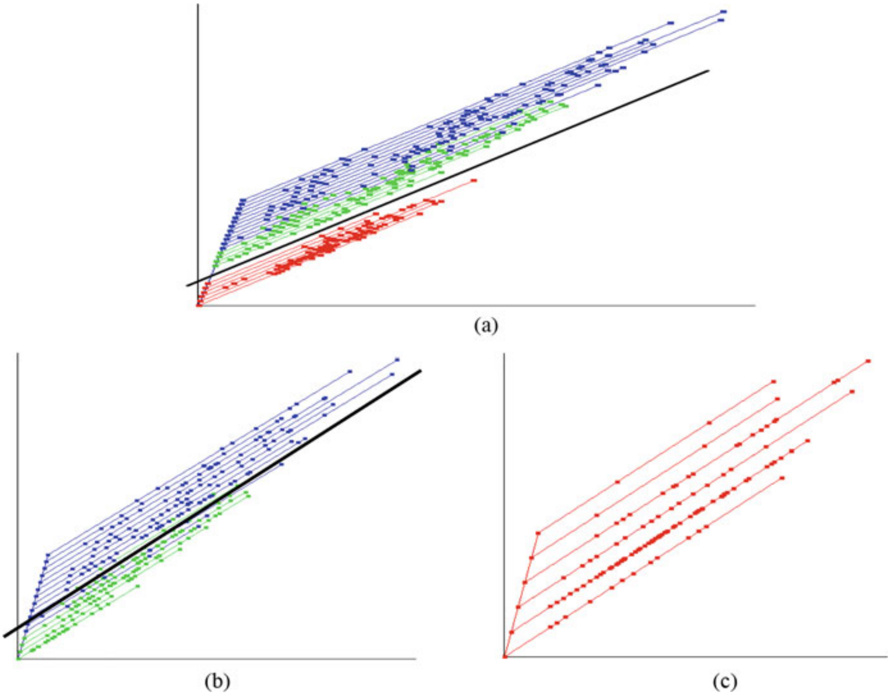


Fig. 11 DSC1 plot series classifier. (**a**) Complete Iris dataset with graphically linear separator (black line). (**b**) Top split. (**c**) Bottom split

separators. Figure 11a shows the graphical linear separator that separates the entire Setosa (red) class from the Virginica (blue) and Versicolor (green) classes. The next step of the classifier is separating the Virginica and Versicolor classes. There does not exist a spot that can completely divide the two classes without misclassifying some samples as shown in Fig. 11b, thus one must analyze the best spot for a graphically linear separator. This methodology is very similar to rule establishment in a decision tree where clear divides between classes may not exist.

2.2 Dynamic Scaffolding Coordinates Based on Shifted Paired Coordinates

Dynamic Scaffolding Coordinates (DSC2) based on Shifted Paired Coordinates generalize the Shifted Paired Coordinates by creating a series of origin-to-pair scaffolds. Each scaffold is connected tip-to-tail; however, the tail of the first scaffold line is removed as the first attribute pair is the starting point of the multidimensional line.

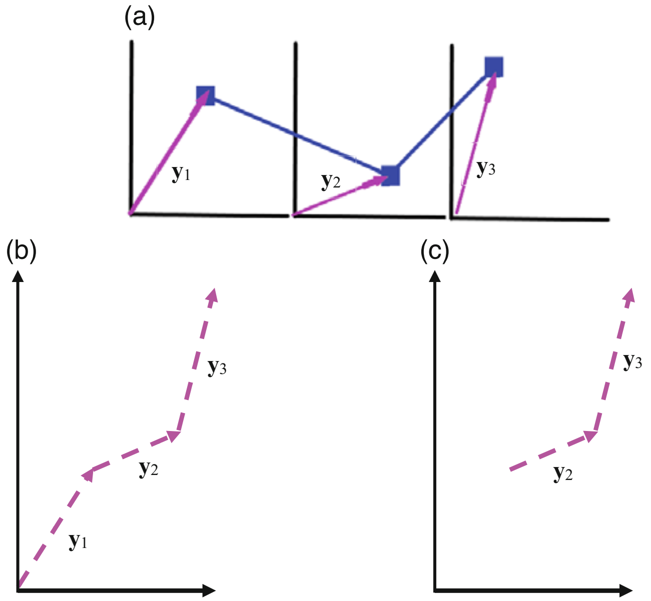


Fig. 12 Visual steps for construction of the DSC2 plot. (**a**) One sample with scaffolds on SPC. (**b**) Connecting the scaffolds. (**c**) Removing the first scaffold

The DSC2 graph construction algorithm illustrated in Fig. 12 is as follows:

1. Set up dataset sample coordinates in the same manner as a SPC plot.
2. Create a scaffold from the origin to the attribute-pair point for each attribute-pair and for all samples.
3. The first attribute-pair scaffold position is left untouched; however, the tail of the first scaffold is removed, making the tips of the first attribute-pair the "origin" of the polyline.
4. Translate the remaining scaffolds, to the tip of the preceding scaffold

The general mathematical description of the DSC2 process is the same as for DSC1. The only difference is that vectors are produced by using SPC not by using parallel coordinates. Here we create a set of 2-D vectors $\mathbf{y}_i$ from n-D point $X = (x_1,x_2,..,x_n)$ as follows, $\mathbf{y}_1 = (x_1,x_2)$, $\mathbf{y}_2 = (x_3,x_4), \ldots$ $\mathbf{y}_{\mathrm{n/2}} = (x_{n\text{-}1},x_n)$ for even n. If the dimension n is odd then we repeat the last x_n getting $\mathbf{y}_{(n+1)/2} = (x_n,x_n)$. Then we add these vectors $\mathbf{y}_i$ to the origin point O: $V_n = O + \mathbf{y}_1 + \mathbf{y}_2 + \ldots + \mathbf{y}_{n/2}$ to produce a *directed graph* like shown in Fig. 12. Here, again, in contrast with classical summation of vectors with the n-D point, which will produce a single n-D point V_n, we preserve results of all consecutive sums $V_i = O + \mathbf{y}_1 + \mathbf{y}_2 + \ldots \mathbf{y}_i$. These points $(V_1,V_2,\ldots,V_n)$ are vertices of the directed graph. The DSC2 process is also lossless in the same way as DSC1 process without any loss of n-dimensional information. In contrast, with DSC1 it contains two times less nodes for even n. Figure 13 allows to compare

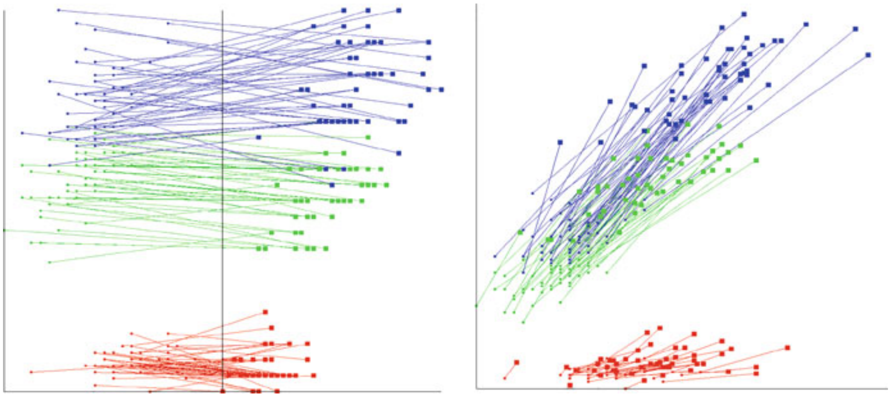


Fig. 13 Side-by-side comparison of SPC and DSC2 of Iris Dataset. (**a**) Iris dataset on SPC. (**b**) Iris dataset on DSC2

Iris data in SPC and DSC2. Here in contrast with DSC1 vs. parallel coordinates, DSC2 did not produced better separation of the classes than SPC. In the next section we will show a way how DSC2 is enhanced to make more appealing on these data for finding the area where the classes overlap.

3 Methods and Experimental Results

In this section we describe the methods for visualizing hyperblocks and finding the worst-case splits of data to training and validation sets in DSC. These methods are illustrated on Iris, Seeds, and Wisconsin Breast Cancer (WBC) datasets. In addition, we explored how our visualization performs on large datasets such as MNIST Handwritten Digits. Attributes of interest are developed to produce class separation on the DSC2 plot. Attributes of interest are important because testing every attribute permutation on a DSC2 plot, a factorial time complexity problem, is not feasible for large datasets of a high dimensionality.

3.1 Overview of Methods

We employed three methods to find the **attributes of interest**: (1) hyperblock analysis from decision trees, (2) principal component analysis, and (3) t-distributed stochastic neighbor embedding (t-SNE).

As was pointed out above **hyperblocks** play an important role in visualization of multidimensional data of multiple classes. While in general, the distribution of data of multiple classes can be very complex in multidimensional space, hyperblocks are quite simple and interpretable. Therefore, there is a great interest to visualize hyperblocks, which do not overlap in the n-D space, also non-overlapping in the 2-D visualization space. If this goal will be reached, then data with more complex distributions in a high-dimensional space can be represented as combinations of several hyperblocks. This is especially beneficial when those hyperblocks are pure, i.e., contain only cases of given class.

More formally and specifically this task can be formulated as follows. Consider m non-overlapping HB$_i$, $i = 1{:}m$ in the n-D space. It means that for each pair HB$_i$ HB$_j$ exists a coordinate X$_{k(i,j)}$ where these HBs do not overlap. The goal is finding and using those separating coordinates X$_{k(i,j)}$ to visualize HBs without-overlap in 2-D.

If the number of these separating coordinates is relatively small, then the chances to visualize those hyperblocks non-overlapping in the 2-D visualization space is much higher. However, it is possible that all X$_{k(i,j)}$ differ for all pairs of these HBs. The total number of these pairs is the number of pairs combinations. Thus, the attempts to visualize many non-overlapping hyperblocks in DSC1 without overlap can fail.

For three HBs it can be done successfully as the Iris example above demonstrates. Therefore, a sequential process is feasible, where first data are split to three hyperblocks. These initial HBs can be impure in a high-dimensional space. Then for each of these HBs, the process continues to split them to 2-3 hyperblocks. With more iterations, the HBs can be made purer and purer. As we see, this is very similar to the decision tree process, while it does not require a binary split. It can operate with three HBs at each iteration. In the actual examples below, we used the decision trees to find the hyperblocks as a computationally efficient process.

Additional attributes can be very beneficial in the situation when hyperblocks are not sufficient to resolve the issue being too simple relative to the actual distribution of the cases, which may require too many hyperblocks. The main idea of using additional attributes is as follows. We compute first two principal components of the dataset, add them to the dataset as two additional attributes and use in visualization along with original attributes. Similarly, we can use t-SNE or Multidimensional Scaling (MDS) components.

As was shown in Sect. 2.1 for DSC1 and below for DSC2 for the Iris dataset hyperblock analysis was successful for class separation. On the other hand, to produce quality class separation of a more complex WBC dataset on DSC2 using hyperblock analysis was not sufficient. To visualize WBC class separation, we escalated to using two principal components in addition to the real dataset attributes. Likewise, for MNIST we were unable to produce class separation using hyperblock analysis or principal components and escalated to using t-distributed stochastic neighbor embedding components in addition to principal components.

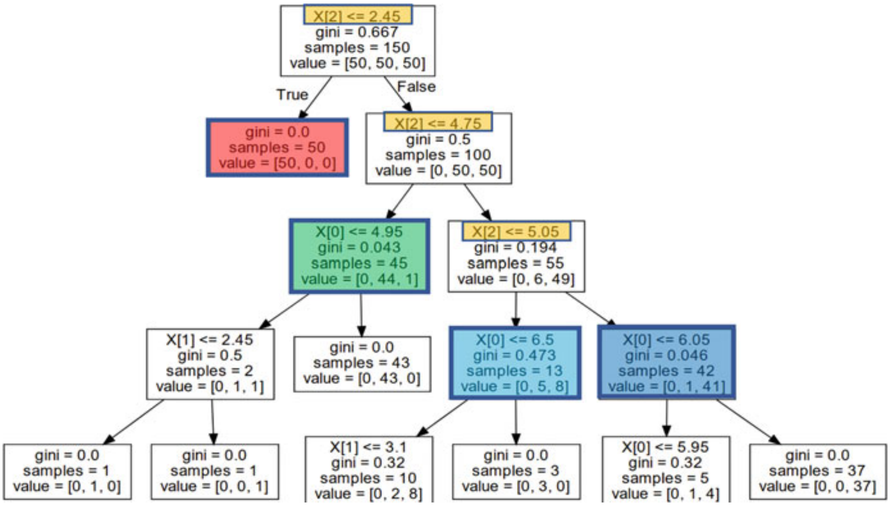


Fig. 14 Four levels below the root (level 0) of the Iris decision tree

3.2 DT Hyperblocks with Iris Dataset

The Iris dataset contains 150 samples of three types of Iris flower (Setosa, Virginica, and Versicolor) [6, 7]. The number of samples are balanced between the classes meaning there is 50 samples per Iris flower. The dataset has four attributes relating to petal width, petal length, sepal length, and sepal width.

The decision tree (DT) analysis is a simple way to develop HBs for this dataset. Figure 14 shows a decision tree model built using the Classification and Regression Tree algorithm (CART) with Gini impurity criterion, and greedy approach on the best split. This figure forms HBs of the Iris dataset. Each sub-branch of the DT represents one HB. To produce non-overlapping HBs from a DT a parent and child node cannot simultaneously be selected because the parent node contains the information of the child, essentially a child is a sub-hyperblock contained inside a parent hyperblock. Going deep into a decision tree a branch can form 100% pure HBs, which contain cases of only one class. However, it runs the risk of overfitting as HBs with very few or a single sample will be present, as some full branches ended in the terminal nodes of the DT show in Fig. 14 with a single case.

The HB ended in the red node has 100% Setosa purity, the HB ended in the green node has 97.78% versicolor purity, and the HB ended in blue node has 97.61% virginica purity. These HBs have high purity and contain many samples to reduce the possibility of overfitting. The HB ended in cyan node has only 61.53% virginica purity. It is highly impure. The nodes with the orange highlighted text illustrate the decision tree rules which create such HBs. These DT rules can be applied to multidimensional visualizations when ordering attributes as shown in Fig. 15. Figure 16 is a modified Fig. 15b where connecting lines are omitted. In Fig. 16

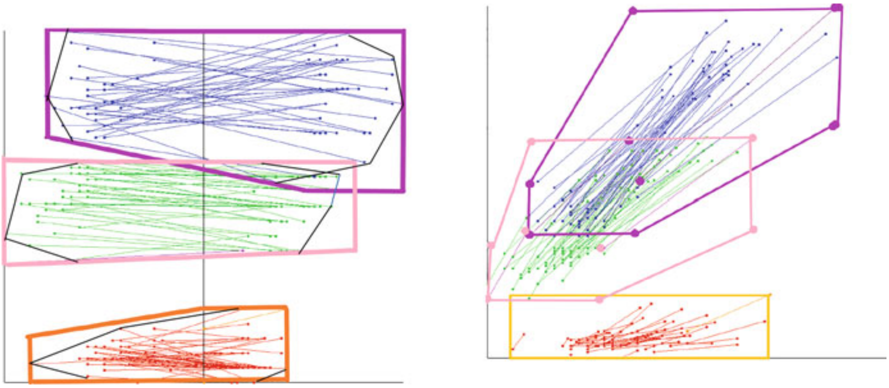
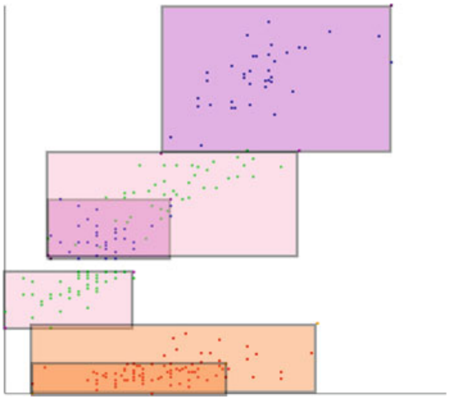


Fig. 15 Side-by-side comparison of SPC and DSC2 of three Iris HBs with outlined classes. (**a**) Iris dataset on SPC. (**b**) Iris dataset on DSC2

Fig. 16 Three main class hyperblocks chosen from the decision tree



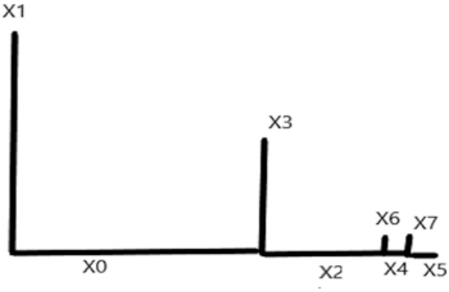
the rectangle 1.1 includes the start points of the cases of Setosa class (class 1), the rectangle 1.2 includes the end points of cases of this class. Similarly, rectangles 2.1, 2.2, 3.1 and 3.2 show in DSC2 these points of cases from remaining two classes.

One caveat to the transformation of SPC to DSC2 is that the succeeding attribute-pairs can condense onto the preceding attribute-pair as shown in Fig. 16 where the tips of the green class are nearby the tail of the blue class for some samples. See rectangles 2.2 and 3.1 in Fig. 16. It is particularly noticeable when an attribute-pair consist of zeros or very small values. One method we deployed to combat this phenomenon is to scale certain attributes-pairs to be larger than the succeeding attribute-pairs as shown in Fig. 17.

This effect has diminishing returns on attribute-pairs that come last due to polyline growth always being in the positive up and right directions, thus it is beneficial to carefully select the first couple of attribute-pairs.

By decreasing the size of the succeeding attribute pairs or increasing the size of the preceding attribute pairs we were able to highlight better separability between

Fig. 17 Downscaling attribute-pair axes



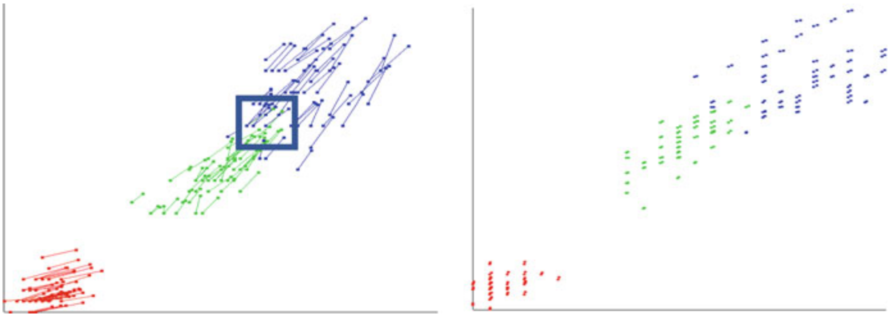


Fig. 18 Emphasizing key attribute-pairs on DSC2. (**a**) Scaling second attribute pair to 10%. (**b**) Scaling second attribute pair to 99%

classes. In Fig. 15b there was higher visual class overlap, and it was difficult to select samples for a validation set that would result in a *worst-case* split, however in Fig. 18 it is immediately noticeable which samples to select for a *worst-case* split.

The Setosa (red) class is completely separated and can be omitted from the worst-case decision analysis as ML models would not struggle to classify Setosa from the other two classes. However, Virginica (blue) and Versicolor (green) have overlap. This leaves 100 samples between the two classes and a 90%–10% test-validation split would require 10 samples to be selected which is shown in Fig. 18a. DSCViz features a clipping function that employs the Cohen-Sutherland line clipping algorithm to find samples that are clipped by a user defined rectangle. There also exists vertex bounding where samples are selected if any vertex of the sample's polyline is contained within the box. This can be useful when viewing the dataset using marker points rather than polylines.

Figure 19 is used to explain the reason why we need to reduce class overlap as much as possible before choosing samples for a *worst-case* validation split. Whilst the orange rectangle is certainly over an area that appears to be highly overlapped, picking those samples may not lead to a bad split because those attributes are likely to not be used by the classification algorithms. Clearly a model such as DT or SVM would make a classification decision using the third or fourth attributes and ignore the first two attributes. By reducing overlap as much as possible we can determine which areas of a dataset that a ML model might use in the decision-making process

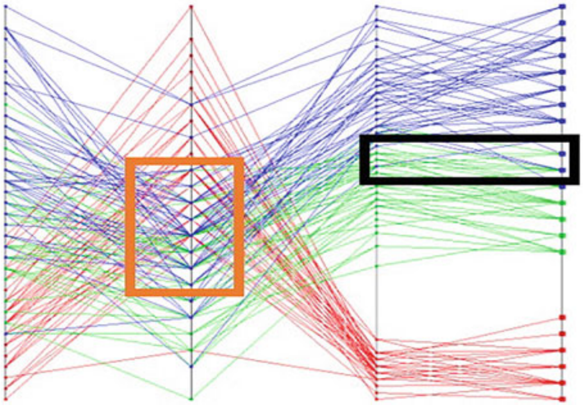


Fig. 19 Two areas of overlap on the Iris dataset on PC

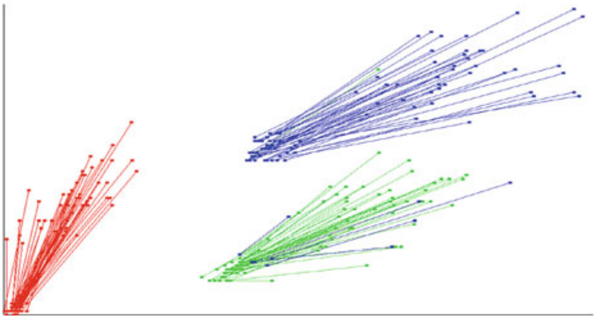


Fig. 20 Non-linear scaling on certain attributes of the Iris dataset in DSC2 after non-linear rescaling

rather than ignore. The black rectangle is an excellent spot to choose samples for a bad split. However, the Iris dataset is a simple dataset, and it is clear where ML models may decide to create rules. On larger datasets it is more difficult to determine what areas of overlap are forced into a ML model's rule creation rather than ignored.

This analysis highlights the close relations between (1) building a visualization with the *smallest overlap* of cases of different classes and (2) finding the *worst validation* set in this visualization space. In essence, while these problems may seem unrelated, but they are really two sides of the same coin. We want to get visualization where the classes are separated as much as possible and then pick up the area where they still overlap to be used as a worst validation set.

Another technique known as non-linear scaling [10, 18] can be used to separate Iris data as shown in Fig. 20. The decision tree in Fig. 14 for the Iris dataset made a rule that separated majority of the Setosa class (red) at a normalized value of 0.17 for petal length. Another decision tree was used to pick up separation using the petal width attribute. The Virginica class (blue) was separated from the Versicolor class (green) 0.67 for petal length. Samples with a petal length attribute greater than 0.17 scaled closer to a normalized value of one, while values less than 0.17 were scaled closer to a normalized value of zero. Likewise for petal width we used 0.67 as the indicator to push attributes values towards 0 or 1. Non-linear scaling exaggerations

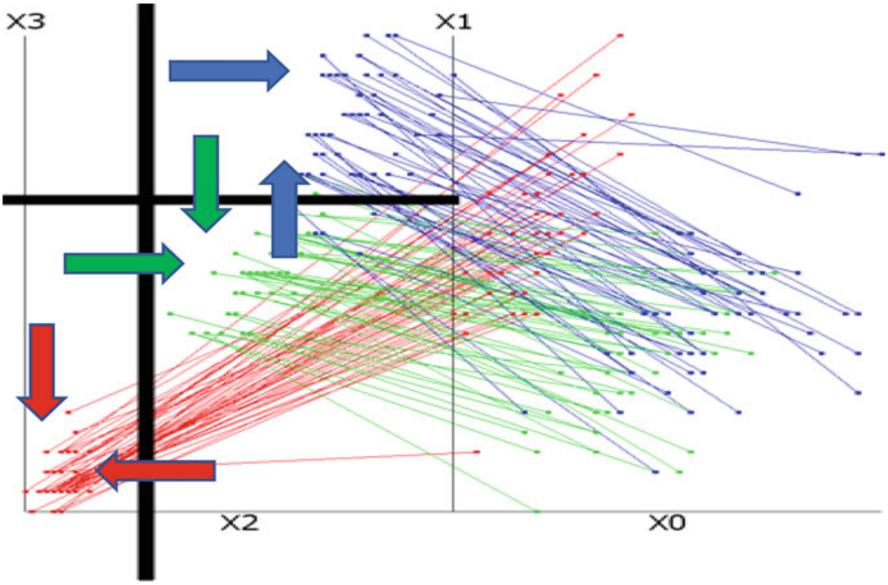


Fig. 21 Non-linear scaling technique on SPC

is also applied. A high exaggeration would squish attribute values at the limits of 0 and 1 whilst a low exaggeration will retain more of the original data.

Figure 21 demonstrates how non-linear scaling was applied on the first attribute-pair to create Fig. 20. The classes are pushed in the direction of the corresponding color arrows. The Virginica class (blue) is pushed up because it is above the black horizontal line and the Versicolor class (green) and Setosa class (red) are pushed down as they are below the black horizontal line. The red class is pushed to the left because it is on the left side of the black vertical line whilst the green and blue classes are pushed right as they are right of the black vertical line.

3.3 DT Hyperblocks with Seeds Dataset

The Seeds dataset contains 210 samples of three types of wheat seeds (Kama, Rosa, and Canadian) [2, 6]. The number of samples are balanced between the classes meaning there is 70 samples per wheat seed. The dataset has seven attributes relating to area, perimeter, compactness, length of kernel, width of kernel, asymmetry coefficient, and length of kernel groove.

Immediate separation of classes is not apparent using a PC or SPC as shown in Fig. 22. Note that the Kernel Groove attribute was duplicated to create an even number of attributes for the SPC plot. We used a decision tree analysis to find attributes of interest for DSC2 in the process like what described above for the

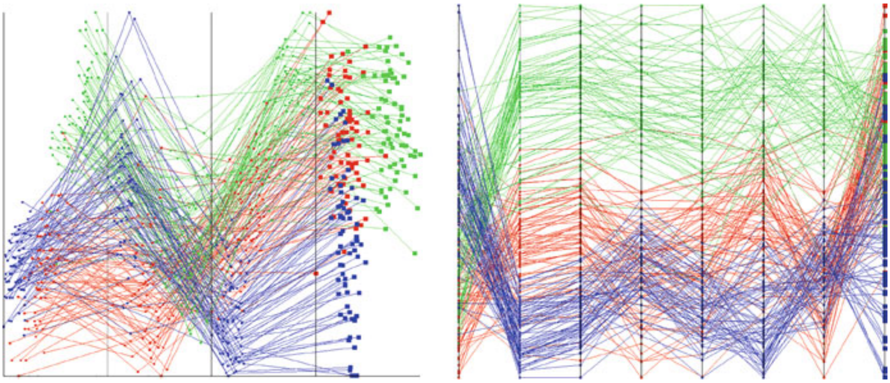
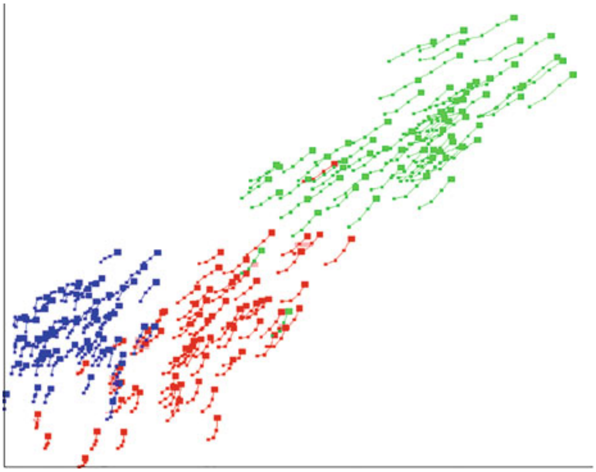


Fig. 22 Seeds dataset on SPC and PC. (**a**) Seeds dataset on Shifted Paired Coordinates. (**b**) Seeds dataset on Parallel Coordinates

Fig. 23 Seeds on DSC2 after scaling and HB analysis



iris dataset. This decision tree is in Appendix A. The attributes of interest happened to be area and kernel groove. The resulting visualization in DSC2 is shown in Fig. 23, which demonstrates its advantages over visualization of the same Seeds data in Shifted Paired Coordinates and Parallel Coordinates shown in Fig. 22. The three classes of seeds are much better separated in Fig. 23.

Analyzing the decision tree in Appendix A, we established three hyperblocks with relatively high purity: the green HB contained 68 samples of the green class and 1 sample from the red class, whilst the blue HB contained 70 samples of the red class and 14 samples of the red class, and finally, the red HB contained 55 samples of the red class and 2 samples of the green class. The green HB was separated from the red and blue HBs at a kernel groove value of 5.576 whilst the red and blue HBs were further separated from each other at an area value of 13.410. This decision tree

Fig. 24 Enhanced Version of the Seeds DSC2 Plot

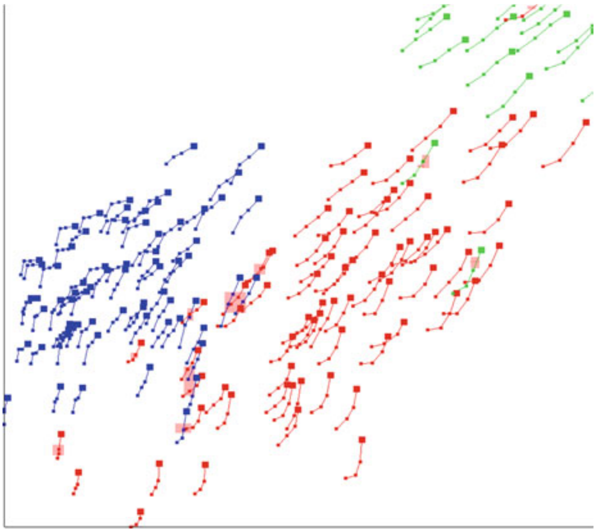


Table 1 Standard ten-fold cross validation model accuracy on Seeds dataset

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	AVG
DT	95.2	95.2	90.5	95.2	100.0	81.0	100.0	95.2	76.2	81.0	91.0
SVM	100.0	95.2	90.5	100.0	100.0	85.7	100.0	90.5	71.4	66.7	90.0
RF	85.7	95.2	95.2	95.2	100.0	95.2	100.0	95.2	71.4	85.7	91.9
KNN	95.2	95.2	90.5	85.7	100.0	81.0	95.2	90.5	76.2	76.2	88.6
LR	100.0	95.2	95.2	100.0	100.0	81.0	95.2	90.5	81.0	76.2	91.4
NB	90.5	90.5	95.2	90.5	100.0	90.5	100.0	95.2	61.9	76.2	89.0
SGD	81.0	85.7	81.0	90.5	81.0	85.7	95.2	81.0	71.4	90.5	84.3
MLP	95.2	90.5	81.0	95.2	100.0	81.0	95.2	90.5	85.7	81.0	89.5

model is also a CART model with Gini impurity criterion, and greedy approach on the best split.

Using Fig. 23 we employed a box clipping algorithm on 21 samples of the Seeds dataset which appeared to be overlapped. Figure 23 is the full dataset and is difficult to see the boxes we used to clip samples as we plucked individual samples at a time. Figure 24 offers an enhanced view of Fig. 23 which shows the samples we clipped into the validation set.

After selecting 21 samples (10% of the Seeds dataset) and removing the duplicated kernel groove attribute we compared our validation split to a standard ten-Fold Cross Validation (CV) package in the scikit-learn library [13]. The 10 splits were reused for each model of the eight contained in Table 1. For each ML model all parameters were kept as default. The classifiers are as follows: Decision Tree (DT) using CART algorithm and greedy approach, Support Vector Machine (SVM) with linear approach and l2 penalty, Random Forests (RF) with 100 estimators, K-Nearest Neighbors (KNN) with 5 neighbors and uniform weights, Logistic Regression (LR)

Table 2 Estimate of the worst-case split on Seeds dataset

Model	Accuracy
DT	57.14%
SVM	38.10%
RF	61.90%
KNN	23.81%
LR	38.10%
NB	57.14%
SGD	42.86%
MLP	23.81%

with l2 penalty, Gaussian Naïve Bayes (NB), Stochastic Gradient Descent (SGD) linear classifier, and Multilayer Perceptron (MLP) with one hidden layer of 100 hidden units.

Table 1 shows that standard ML algorithms can classify the Seeds dataset within 84.3–91.9% average accuracy without any additional processing, dimensional reduction, or feature engineering. The lowest split was 66.7% in SVM, and the highest split was 100% across seven models. The best performing model was Random Forest, and the worst performing model was Stochastic Gradient Descent.

Table 2 shows that despite the strong model accuracies obtained in Table 1, all eight classifiers had an accuracy between 23.81% and 61.9% in the upper estimate of the worst-case split. Again Random Forest was the best performing algorithm whilst KNN had the lowest performance. Intuitively this makes sense that KNN would be the lowest because the samples we took from the DSC2 plot had neighbors from a different class.

3.4 DT Hyperblocks and Principal Components with Wisconsin Breast Cancer Dataset

The Wisconsin Breast Cancer dataset contains 699 samples using nine descriptive attributes: clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses [6, 19]. We removed 16 samples which have missing values leaving a total of 683 samples. Those 683 samples include 444 benign cases and 239 malignant cases. We chose the WBC dataset due the high-risk nature of cancer tumor diagnosis. A misdiagnosis of a malignant tumor as a benign tumor could prove fatal for the patient [19].

Figure 25 shows the WBC dataset on PC and SPC plots. In both types of plots the dataset samples are heavily overlapped, however, the benign class (green) is concentrated towards the bottom of both plots. The benign class has 1.86 times more samples than the malignant class (red), but the malignant class consumes more area of the plot.

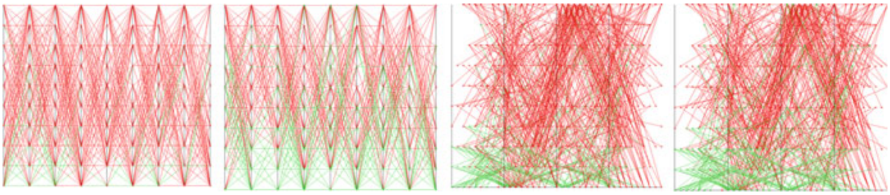


Fig. 25 Wisconsin Breast Cancer dataset on PC and SPC. (**a**) Malignant class on top on PC. (**b**) Benign class on top on PC. (**c**) Malignant class on top on SPC. (**d**) Benign class on top on SPC

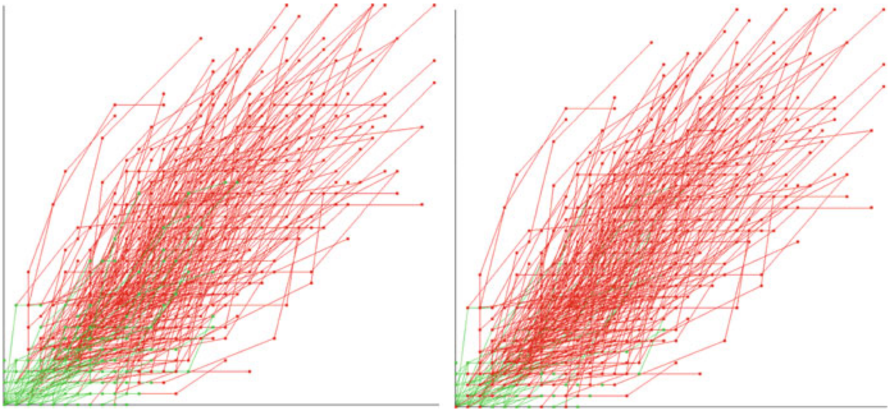


Fig. 26 DSC2 of the WBC dataset. (**a**) Benign class on top on DSC2. (**b**) Malignant class on top on DSC2

We remove the mitosis attribute for the SPC plot as it requires an even number of attributes. This decision was made from analysis of DT (see Appendix B), that showed the mitosis feature was not used for complete set separation in DT. As in the previous datasets the decision tree model also used CART with Gini impurity criterion, and greedy approach on the best split.

Figure 26 shows the WBC dataset on DSC2. It is shown that the benign class (green) exists mostly in the bottom left corner and has short polyline growth whilst the malignant class (red) also starts in the bottom left corner but has long polyline growth. To reduce the area of overlap to a manageable level we employed attribute scaling to find regions of overlap that would have meaning to a ML model.

Using the DT analysis in Appendix B, it was difficult to decide attributes of interest because of the many branches and deep level of the decision tree. Unlike the Iris dataset the WBC dataset requires multiple attributes to separate the two classes. It was noticed that the uniformity of cell size and uniformity of cell shape attributes were a reoccurring attribute that the decision tree used to create rules. Thus, we made these two attributes the attribute of interest and placed them as the first attribute-pair on DSC2 (Fig. 27). After upscaling, the first attribute-pair we were able to develop a clearer picture of where proper overlap can exist for a *worst-*

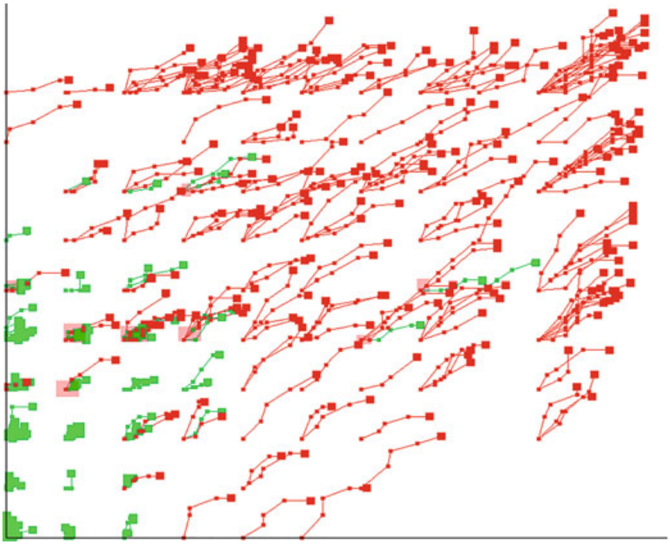
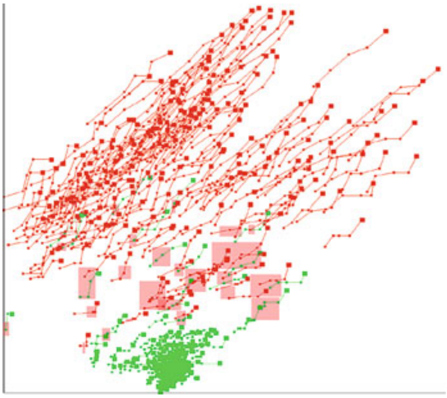


Fig. 27 WBC dataset on DSC2 after upscaling the first attribute-pair

Fig. 28 WBC dataset with two Principal Components



case split. Proper overlap refers to areas of overlap that likely will not be ignored by the classier algorithms. The red squares in Fig. 27 represent areas that would clip samples for a validation set.

Next we further escalated WBC data visualization by utilizing principal component analysis as an attempt to preserve global structure in only two dimensions. These two principal components would become our attributes of interest and accordingly placed as the first attribute-pair (Fig. 28). Compared to Fig. 27 it became easier to identify which samples would result in a *worst-case* split. To have enough samples to fill a validation set we used overlapped samples and samples that were near the boundaries of the two classes.

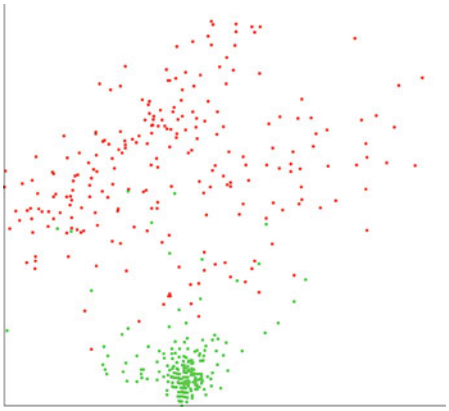


Fig. 29 WBC Principal Components

For some datasets dynamic scaffolding coordinates may not be necessary when looking for worst case splits. The two attributes of interest (in this case the principal components) in the form of a scatterplot would be just as effective. However, when combining the WBC attributes with the two principal components we were able to capture the general structure of the samples. One obvious trend in Fig. 28 that does not show up in Fig. 29 is that malignant cases tend to have large values across many attributes whereas benign cases tend to have attribute values close to zero. The end-user can gain a better insight into which samples are likely malignant and which ones are benign. It could also be useful in determining edge cases of benign tumors that may transform into a malignant tumor later. Of course, this line of logic would require going through the scientific process to determine if benign cases with long polylines are more likely to turn into malignant cases. With Fig. 29 these higher-level insights would not be possible as the points themselves have no differentiation besides location and class color.

3.5 MNIST Handwritten Digits with Autoencoder and T-SNE Components

The datasets we experimented in previous sections were relatively small in both dimensions, and the number of cases. The goal of this section is to explore capabilities of DSC visualization for a much larger dataset. For this purpose, we selected the MNIST Handwritten digits dataset (MNIST) [3], which contains 60,000 samples of digits on a 28 × 28-pixel grid in a training set. There is an additional test set containing 10,000 samples. The dataset is available from Kaggle.com. The pixel values are represented in grayscale between 0 and 255. The 28 × 28-pixel grid was converted to a 784-dimensional dataset. In the previous sections, we were able to visualize smaller datasets in SPC and parallel coordinates and have shown advantages of DSC over them. For larger MNIST dataset, a SPC or PC our

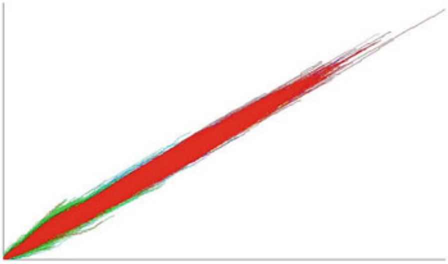


Fig. 30 784 Attributes of MNIST on DSC2

implementations did not handle the 784 attributes. These methods require special treatment to deal with heavy occlusion. The decision three approach, which was successful for smaller datasets above, has a limited applicability to the MNIST dataset, because the produced DT was very large, deep and with accuracy only around 79%. Respectively, finding attributes of interest in the MNIST DT was difficult.

Unlike SPC and PC we were able to render a plot that fit inside the application window using DSC2. The plot in Fig. 30 gave lack of insight into the dataset as all ten digits were stacked on top of each other. The dimension reduction technique using Autoencoder was more successful. In prior work [5] two digits of MNIST were analyzed together using 32 autoencoder features with a high model accuracy. Therefore, we reduced the MNIST dataset to 32 autoencoder attributes and plot them on DSC2. Unlike [5] that compared only two digits, we chose to compare a single digit to all other digits.

Figure 31 provided interesting digit patterns of the MNIST dataset on DSC2. Each digit had a slightly different polyline growth pattern and density. However, there is large amounts of overlap between the ten-digit classes, and it was difficult to visually separate the classes from each other. The next attempt was to use t-SNE components as attributes of interest (Fig. 33), considering that t-SNE has been used to visualize MNIST digits with high clarity (Fig. 32).

We reduced the MNIST dataset from 784 attributes to only 50 truncated Singular Value Decomposition (SVD) components. SVD describes a linear transformation through rotations and scaling. It is essentially a map between different sized vector spaces. We used Truncated SVD which is beneficial when data is sparse. MNIST can be considered sparse as many pixels in a single digit pixel-grid are without ink. With these 50 SVD components we generated two t-SNE components. However, t-SNE has several drawbacks. t-SNE components cannot be applied to unseen data without applying them simultaneously with the training data due to the lack of a parametric embedding [16], which leads to t-SNE algorithm being considered as a difficult to interpret or *black box* dimension reduction technique. Next, t-SNE does not preserve distance between clusters, nor does it preserve cluster density [13]. It is also very visible in our Fig. 1 for WBC data as we discussed in Sect. 1. t-SNE can be simplified to a dimensional reduction technique that does not preserve global structure but preserves local neighborhoods [8]. t-SNE uses a perplexity parameter

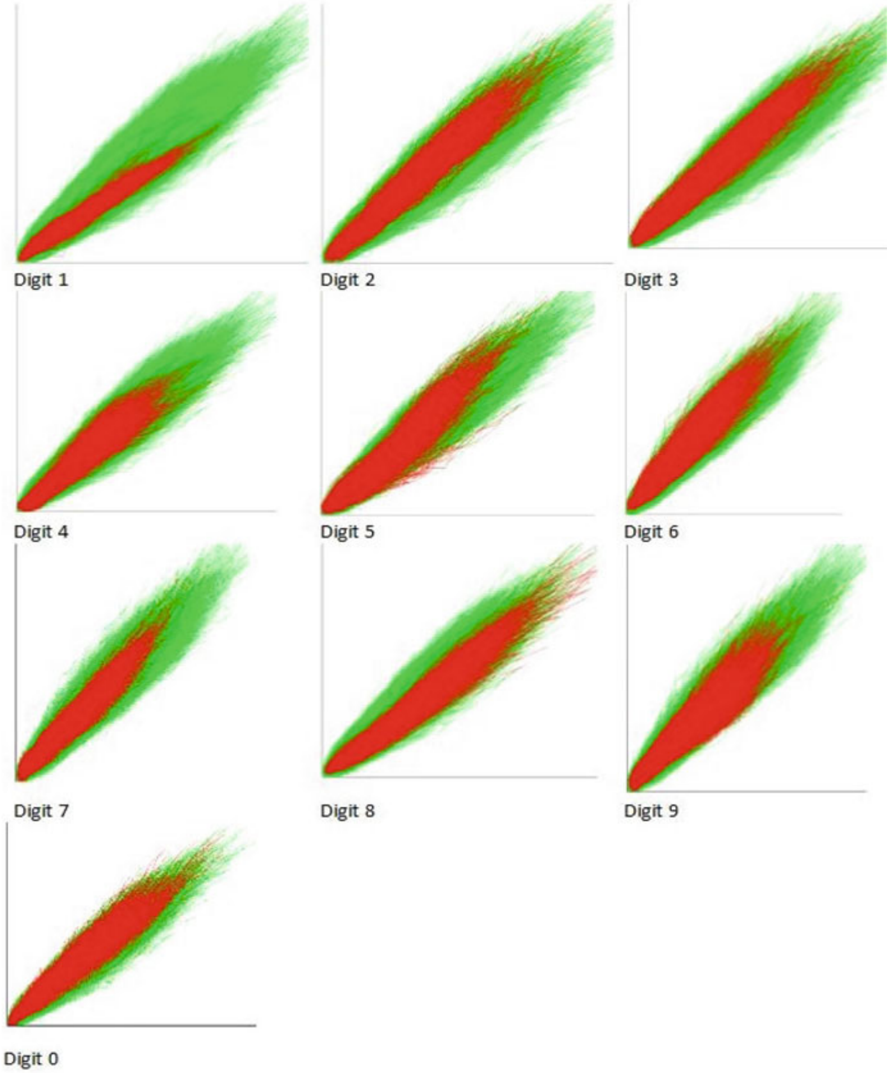


Fig. 31 Each MNIST Digit using autoencoder features against all other digits on DSC2

which we set to 30. The creator of t-SNE recommends a perplexity value between 5 and 50 [13]. This parameter is related to how many nearby neighbors any point may have.

Like the WBC with PCA in DSC2 plot from the previous section, we grew our sample polylines from the two t-SNE components which became our attributes of interest. The MNIST SVD components were downscaled by a factor of 0.003 to keep the polylines from growing on top of other clusters. Figures 32 and 33 look

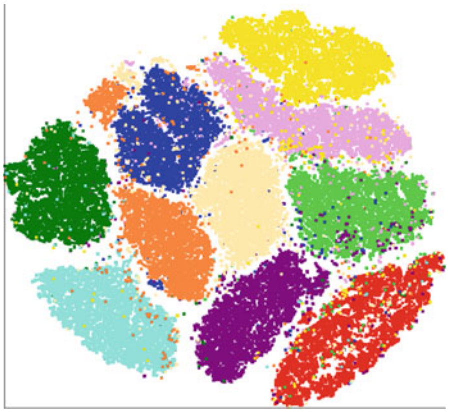


Fig. 32 t-SNE components for MNIST

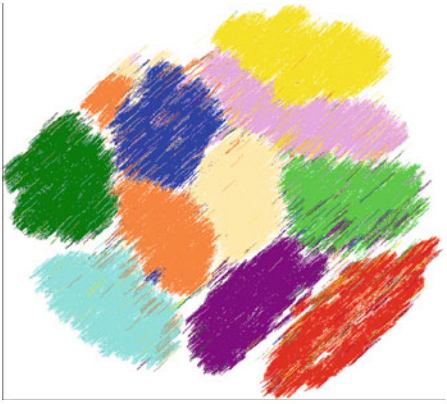


Fig. 33 t-SNE + MNIST attributes on DSC2

very similar when the entire plot is in frame. However, zooming in on the DSC2 plot can reveal important differences between Figs. 32 and 33.

Figure 34 reveals that certain points can end up growing into the main cluster of their class when using DSC2 scaffolding on top of t-SNE. There is a risk to this analysis however, since t-SNE does not preserve distances, whilst the scaffolds from the SVD components do preserve distance in SVD. In future work we will consider other ways of visualizing MNIST dataset without having to rely on black box techniques such as t-SNE, but for our current research we found t-SNE and scaffolding to produce class clusters with high clarity. While MNIST dataset is not a high-risk application, if we were to choose samples for *worst-case* analysis, we would pick samples that lie in a class cluster but are not part of that class.

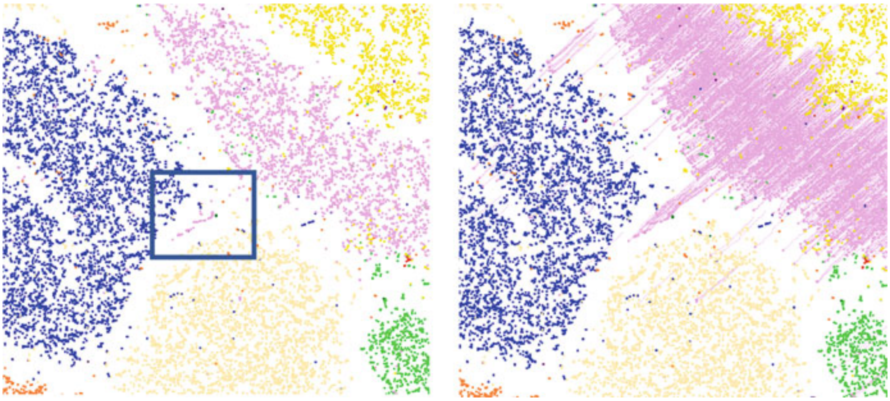


Fig. 34 t-SNE scatterplot vs. DSC2 scaffolds. (**a**) t-SNE only scatterplot. (**b**) t-SNE + DSC2 scaffolds

3.6 Comparison of Worst and Average Validation Sets for Different Classifiers

This section is devoted to comparison of performance on average and worst validation sets, where average validation sets are obtained by using k-fold cross validation and the worst validation sets are obtained by visual methods described in the previous sections. It is important to know how big the difference is between accuracy for the average and worst validation sets for different classifiers to be able to select better ones.

Classification results were obtained from **eight standard ML classifiers** in the sci-kit learn Python library [13] using ten-fold cross validation. The single ten-fold cross validation tested against eight models was compared to the validation split found using several box-clipping areas in Fig. 29. The samples in the multiple boxes were clipped using Cohen Sutherland algorithm into a validation set of 67 samples which is 9.81% of the entire WBC dataset. The PCA components were removed, and the mitosis attribute was added back to both the training and validation set. The goal of our DSC2 visualization was to select samples from the original dataset that may lead to an upper estimate of the *worst-case* split. We refer to this validation split as an *upper estimate*, as this methodology does not guarantee the *absolute worst-case* split.

For each ML model all parameters were kept as default, except for the multilayer perceptron which was given an additional hidden layer from the default of one hidden layer. The classifiers are as follows: Decision Tree (DT) using CART algorithm and greedy approach, Support Vector Machine (SVM) with linear approach and l2 penalty, Random Forests (RF) with 100 estimators, K-Nearest Neighbors (KNN) with 5 neighbors and uniform weights, Logistic Regression (LR) with l2 penalty,

Table 3 Standard tenfold Cross Validation model accuracy for WBC dataset

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	AVG
DT	95.7	91.3	95.7	92.6	95.6	91.1	94.1	98.5	97.1	95.6	94.7
SVM	92.8	98.6	95.7	94.1	98.5	97.1	97.1	100	98.5	98.5	97.1
RF	92.8	94.2	95.7	94.1	98.5	97.1	98.5	98.5	98.5	98.5	96.6
KNN	91.3	98.6	95.6	94.1	100	97.1	98.5	100	98.5	98.5	97.2
LR	92.7	97.1	94.2	94.1	100	97.1	95.5	100	97.1	100	96.8
NB	92.7	95.6	94.2	94.1	98.5	95.6	97.1	97.1	98.5	97.1	96.1
SGD	95.7	92.8	95.7	94.1	100	91.2	95.6	98.5	98.5	100	96.2
MLP	89.8	89.9	94.2	92.6	100	94.1	97.1	100	98.5	98.5	95.5

Table 4 Upper estimate of the worst split from PCA-DSC2 analysis for WBC dataset

Model	Accuracy (%)
DT	62.12
SVM	62.12
RF	65.15
KNN	60.60
LR	63.63
NB	72.72
SGD	57.58
MLP	60.60

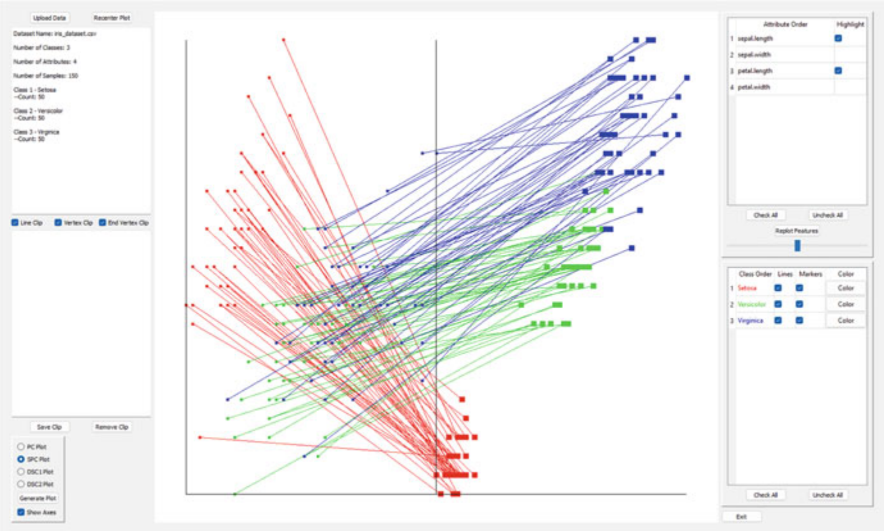
Gaussian Naïve Bayes (NB), Stochastic Gradient Descent (SGD), and Multilayer Perceptron (MLP) with one hidden layer of 100 hidden units.

Table 3 shows that standard ML algorithms can classify the WBC dataset within 94–97% average accuracy without any additional processing, dimensional reduction, or feature engineering. The lowest split is resulted in 89.8% and the highest split is resulted in 100%. The best performing model was KNN and SVM, and the worst performing model was DT classifier.

Table 4 shows that despite the strong model accuracies obtained in Table 3, all eight classifiers had an accuracy between 57% and 72% in the upper estimate of the **worst-case scenario**. In life-critical and other high-risk applications knowing the worst performance of a model can influence reliance on the model and the possibilities of incorporating additional models into decision-making as a safeguard. In this case ten-fold cross validation accuracy suggests using KNN or SVM classifiers, but the model that performed the best on the estimate of the most difficult split was Naïve Bayes by a margin of 7.57% of the next best model Random Forest.

4 DSCViz Software

DSCViz provides a GUI application to control dynamic scaffolding coordinates, parallel coordinates, and shifted paired coordinates plots. It is available upon request for not commercial use. After the dataset is uploaded information about the dataset



Dataset Name: iris_dataset.csv
Number of Classes: 3
Number of Attributes: 4
Number of Samples: 150
Class 1 - Setosa
--Count: 50
Class 2 - Versicolor
--Count: 50
Class 3 - Virginica
--Count: 50

☑ Line Clip ☑ Vertex Clip ☑ End Vertex Clip

○ PC Plot
● SPC Plot
○ DSC1 Plot
○ DSC2 Plot
☑ Show Axes

	Attribute Order	Highlight
1	sepal.length	☑
2	sepal.width	
3	petal.length	☑
4	petal.width	

	Class Order	Lines	Markers	Color
1	Setosa	☑	☑	Color
2	Versicolor	☑	☑	Color
3	Virginica	☑	☑	Color

Fig. 35 DSCViz application running SPC plot

will populate in the top left corner of the window. The user then selects which of the four plots to create using the radio button and clicking generate plot.

From here the user has the capabilities to drag and zoom in real time using GPU rendering by making various OpenGL calls. The dataset vertices are stored in the GPU memory for fast access when doing matrix operations.

DSCViz has been used on datasets such as MNIST, which can include 40 million data points, of course many are overlapped and do not render. If the dragging and zooming is slow a user may use hiding all the classes and markers, dragging the plot, and then reactivating. There are no immediate slowdowns for small datasets such as WBC and Iris, or MNIST after dimensional reduction techniques are applied.

A user can establishes a specific order of features in the UI. Similarly, classes can also be reordered. Individual class markers, class polylines, as well as the plot axes all have toggles to hide. The attribute markers can be controlled at the attribute level by toggling them in the highlight column. There is a general slider that controls the transparency of unselected attributes. The slider set at 0 will completely hide the attributes.

DSCViz gives the user the ability to clip samples from the dataset. The line clipping algorithm is Cohen-Sutherland. There is also the option to clip samples by any vertex or end vertices. Unlike the line clipping method, the vertex methods only clip samples that have a vertex inside the clipping area. Sequential clips can be added and saved any time. The user may remove and reset the clip entirely. All clipping samples are moved into the validation set. Figures 35, 36 and 37 illustrate DSCViz.

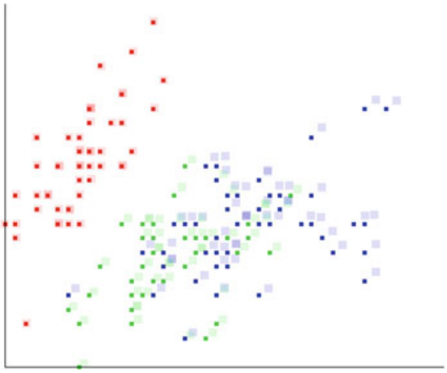
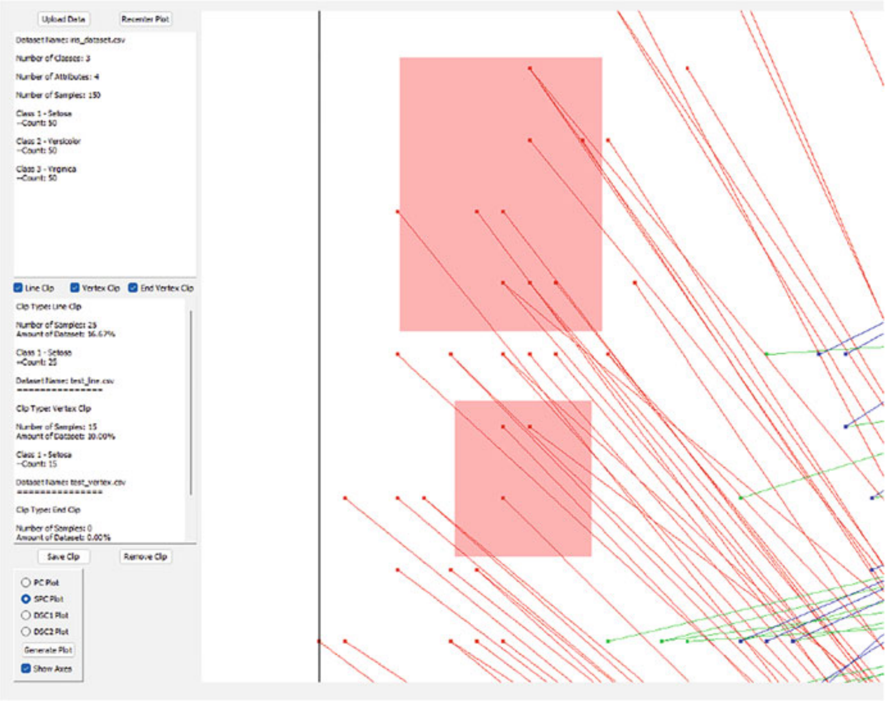


Fig. 36 Lowering attribute transparency



```
Dataset Name: iris_dataset.csv

Number of Classes: 3

Number of Attributes: 4

Number of Samples: 150

Class 1 - Setosa
--Count: 50

Class 2 - Versicolor
--Count: 50

Class 3 - Virginica
--Count: 50
```

```
Clip Type: Line Clip

Number of Samples: 25
Amount of Dataset: 16.67%

Class 1 - Setosa
--Count: 25

Dataset Name: test_line.csv
================

Clip Type: Vertex Clip

Number of Samples: 15
Amount of Dataset: 10.00%

Class 1 - Setosa
--Count: 15

Dataset Name: test_vertex.csv
================

Clip Type: End Clip

Number of Samples: 0
Amount of Dataset: 0.00%
```

Fig. 37 Two clipping areas on DSCViz

5 Conclusions

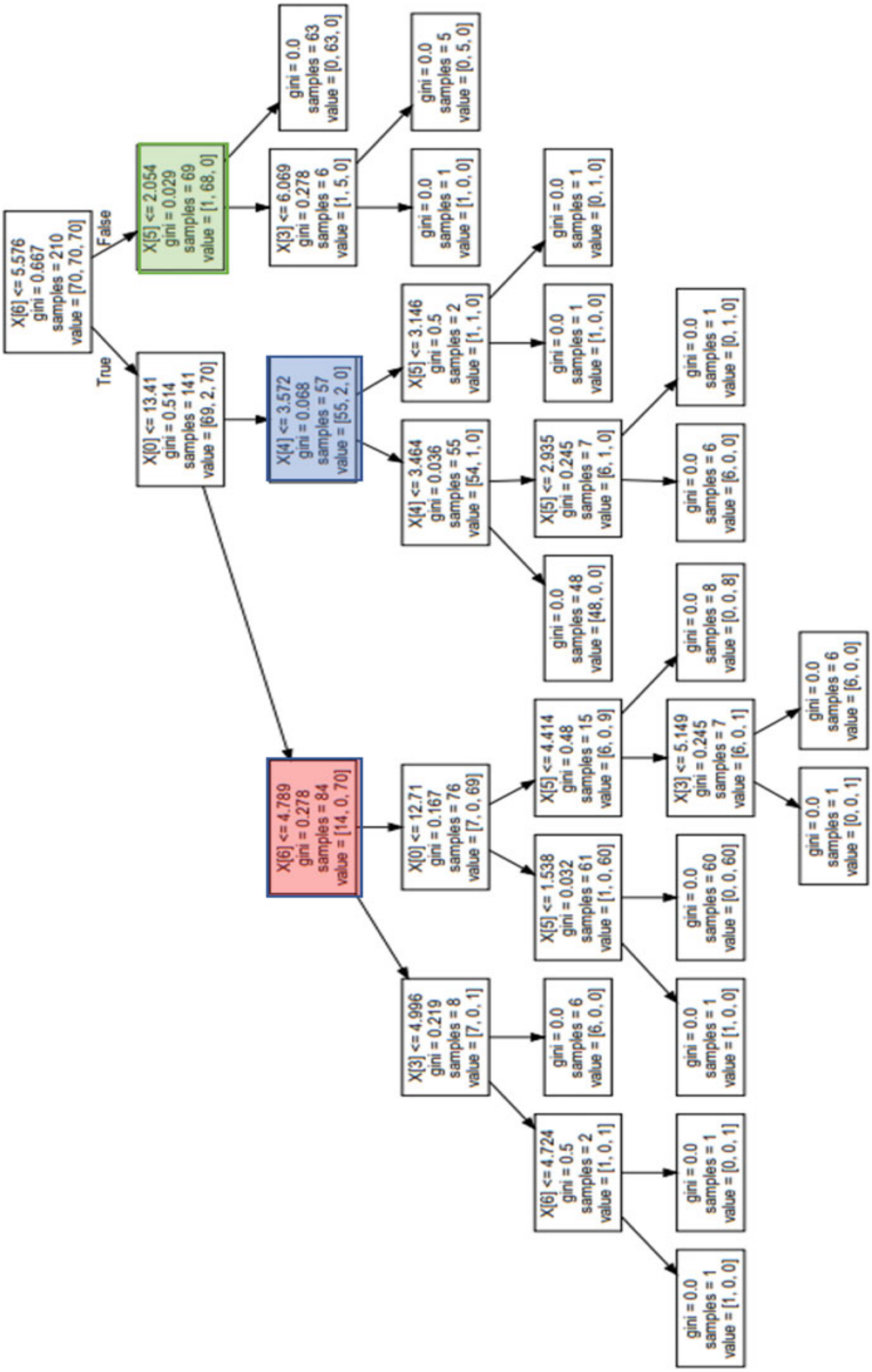
This study contributes to visual and interpretable machine learning methods by developing DSC1 and DSC2 systems that can be used for multidimensional visualization, analysis, and classification. DSC1 and DSC2 have self-service components that allow domain experts to change, add, or remove attributes and select regions of highly condensed samples for model selection.

Hyperblocks as interpretable data units are used to highlight attributes of separation within a dataset as a computationally efficient alternative to genetic or brute force algorithms for attribute order permutation selection. When HBs are unable to provide adequate attributes of separation, we escalated to various dimension reduction techniques that allowed for visualizing dimensionally rich datasets like MNIST.

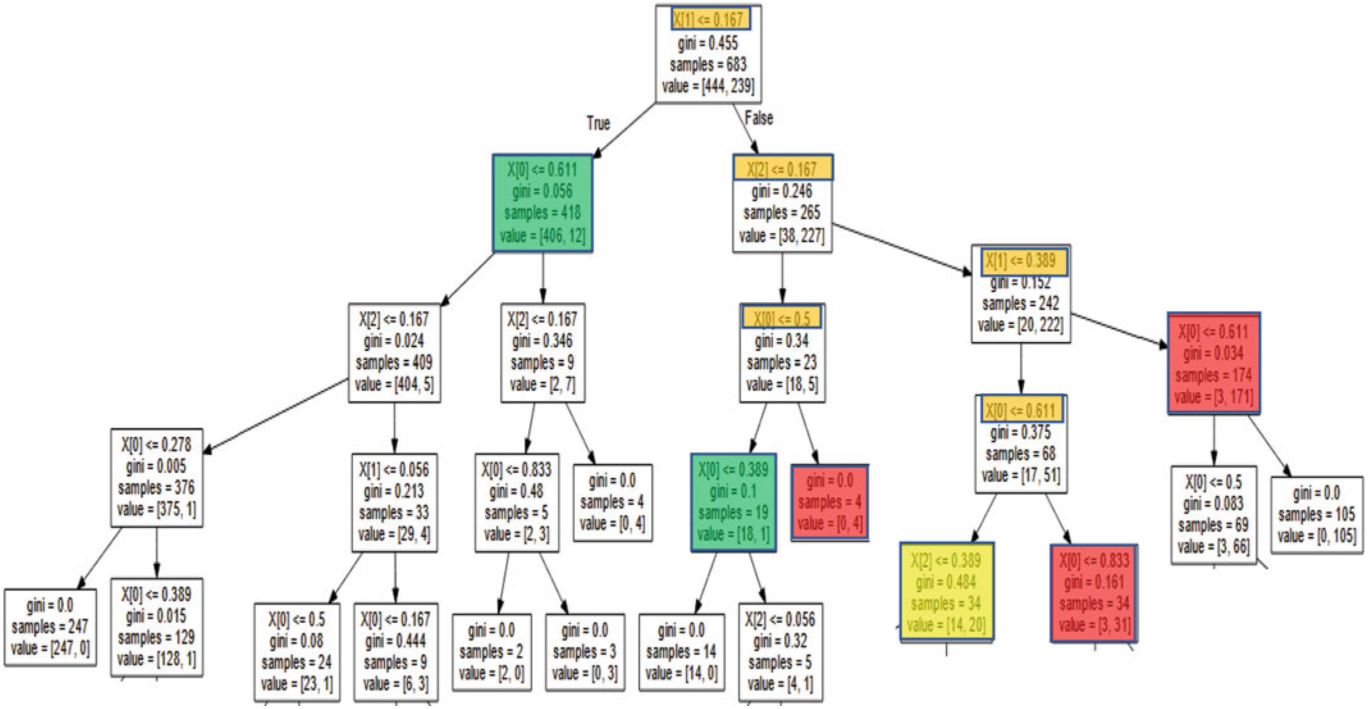
The results in Tables 2 and 4 show the lower accuracy of standard ML models for benchmark datasets when looking for difficult dataset splits. Section 3 provides a framework of visualizing these difficult splits on DSC2. One area that we would like to improve on is visualizing difficult splits in a dataset without reliance on dimensional reduction techniques to provide plot clarity. In future work we look towards developing dimension reduction techniques that preserve hyperblock structure or adapting other dimension reduction techniques.

A.1 Appendices

A.1.1 Appendix A – Seeds Dataset Decision Tree


X[6] <= 5.576
gini = 0.667
samples = 210
value = [70, 70, 70]
True
False
X[5] <= 2.054
gini = 0.029
samples = 69
value = [1, 68, 0]
gini = 0.0
samples = 63
value = [0, 63, 0]
X[3] <= 6.069
gini = 0.278
samples = 6
value = [1, 5, 0]
gini = 0.0
samples = 5
value = [0, 5, 0]
gini = 0.0
samples = 1
value = [1, 0, 0]
X[0] <= 13.41
gini = 0.514
samples = 141
value = [69, 2, 70]
X[4] <= 3.572
gini = 0.068
samples = 57
value = [55, 2, 0]
X[5] <= 3.146
gini = 0.5
samples = 2
value = [1, 1, 0]
gini = 0.0
samples = 1
value = [0, 1, 0]
gini = 0.0
samples = 1
value = [1, 0, 0]
X[4] <= 3.464
gini = 0.036
samples = 55
value = [54, 1, 0]
X[5] <= 2.935
gini = 0.245
samples = 7
value = [6, 1, 0]
gini = 0.0
samples = 1
value = [0, 1, 0]
gini = 0.0
samples = 6
value = [6, 0, 0]
gini = 0.0
samples = 48
value = [48, 0, 0]
X[6] <= 4.789
gini = 0.278
samples = 84
value = [14, 0, 70]
X[0] <= 12.71
gini = 0.167
samples = 76
value = [7, 0, 69]
X[5] <= 4.414
gini = 0.48
samples = 15
value = [6, 0, 9]
gini = 0.0
samples = 8
value = [0, 0, 8]
X[3] <= 5.149
gini = 0.245
samples = 7
value = [6, 0, 1]
gini = 0.0
samples = 6
value = [6, 0, 0]
gini = 0.0
samples = 1
value = [0, 0, 1]
X[5] <= 1.538
gini = 0.032
samples = 61
value = [1, 0, 60]
gini = 0.0
samples = 60
value = [0, 0, 60]
gini = 0.0
samples = 1
value = [1, 0, 0]
X[3] <= 4.996
gini = 0.219
samples = 8
value = [7, 0, 1]
gini = 0.0
samples = 6
value = [6, 0, 0]
X[6] <= 4.724
gini = 0.5
samples = 2
value = [1, 0, 1]
gini = 0.0
samples = 1
value = [0, 0, 1]
gini = 0.0
samples = 1
value = [1, 0, 0]

A.1.2 Appendix B – Wisconsin Breast Cancer Dataset Decision Tree



References

1. Brown, J.: Visualizing multidimensional data with general line coordinates and pareto optimization. Central Washington University, All Master's Theses. 898. Available at https://digitalcommons.cwu.edu/etd/898. (2017).
2. Charytanowicz, M., Niewczas, J., Kulczycki, P., Kowalski, P.A., Łukasik, S., Żak, S.: Complete gradient clustering algorithm for features analysis of x-ray images. In: Information technologies in biomedicine, pp. 15–24. Springer, Berlin (2010)
3. Deng, L.: The MNIST database of handwritten digit images for machine learning research. IEEE Signal Process. Mag. **29**(6), 141–142 (2012)
4. Di Cicco, V., Firmani, D., Kouras, N., Merialdo, P., Srivastava, D.: Interpreting deep learning models for entity resolution: an experience report using LIME. In: Proceedings of the second international workshop on exploiting artificial intelligence techniques for data management, pp. 1–4, New York (2019)
5. Dovhalets, D., Kovalerchuk, B., Vajda, S., Andonie, R.: Deep learning of 2-D images representing n-D data in general line coordinates. In: International symposium on affective science and engineering ISASE2018, pp. 1–6. Japan Society of Kansei Engineering, Chuo-ku (2018) https://www.jstage.jst.go.jp/article/isase/ISASE2018/0/ISASE2018_1_18/_pdf
6. Dua, D., Graff, C.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine (2019) http://archive.ics.uci.edu/ml
7. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Ann. Eugenics. **7**(2), 179–188 (1936)
8. Kobak, D., Berens, P.: The art of using t-SNE for single-cell transcriptomics. Nat. Commun. **10**(1), 1–4 (2019)
9. Kovalerchuk, B.: Enhancement of cross validation using hybrid visual and analytical means with Shannon function. In: Beyond Traditional Probabilistic Data Processing Techniques: Interval, Fuzzy Etc. Methods and their Applications, pp. 517–543. Springer, Cham (2020)
10. Kovalerchuk, B.: Visual Knowledge Discovery and Machine Learning. Springer, Cham (2018)
11. Kovalerchuk, B., Ahmad, M.A., Teredesai, A.: Survey of Explainable Machine Learning with Visual and Granular Methods beyond Quasi-Explanations, pp. 217–267. A perspective of granular computing, Interpretable artificial intelligence (2021)
12. Kovalerchuk, B., Hayes, D.: Discovering interpretable machine learning models in parallel coordinates. In: 2021 25th International Conference Information Visualisation (IV), pp. 181–188. IEEE, Piscataway (2021)
13. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **1**(12), 2825–2830 (2011)
14. Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should i trust you?" explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144. Association for Computing Machinery, New York (2016)
15. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat. Mach. Intell. **1**(5), 206–215 (2019)
16. Van Der Maaten, L.: Learning a parametric embedding by preserving local structure. In: Artificial Intelligence and Statistics, pp. 384–391. PMLR, New York (2009)
17. Van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(11), 2579–2605 (2008)
18. Wagle, S.N., Kovalerchuk, B.: Self-service data classification using interactive visualization and interpretable machine learning. In: Integrating Artificial Intelligence and Visualization for Visual Knowledge Discovery, pp. 101–139. Springer, Cham (2022)
19. Wolberg, W.H., Street, W.N., Mangasarian, O.L.: Machine learning techniques to diagnose breast cancer from image-processed nuclear features of fine needle aspirates. Cancer Lett. **77**(2–3), 163–171 (1994)

20. Coxeter, H.S.M.: Regular Polytopes, 3rd edn, pp. 122–123. Dover, New York (1973)
21. Inselberg, A.: Parallel Coordinates: Visual Multidimensional Geometry and its Applications. Springer, New York (2009)
22. Johnson, W.B., Lindenstrauss, J.: Extensions of Lipschitz mappings into a Hilbert space. Contemp. Math. **26**, 189–206 (1984)
23. Di Cicco, V., Firmani, D., Koudas, N., Merialdo, P., Srivastava, D.: Interpreting deep learning models for entity resolution: an experience report using LIME. In: Proceeding of the 2nd Inter. Workshop on Exploiting Artificial Intelligence Techniques for Data Management Jul 5. pp. 1–4, (2019)
24. Kovalerchuk, B., Andonie, R., Datia, N., Nazemi, K., Banissi, E.: Visual knowledge discovery with artificial intelligence: challenges and future directions. In: Integrating Artificial Intelligence and Visualization for Visual Knowledge Discovery, pp. 1–27. Springer, Cham (2022)
25. DSCVis, https://github.com/CWU-VKD-LAB

Algorithm of Trading on the Stock Market, Providing Satisfactory Results

Alexander Rubchinsky and Kristina Baikova

1 Introduction

The presented work is in line with a broad scientific direction associated with the analysis of market graphs that describe the behavior of the stock market (see, for example, articles [4, 5]). Within this direction, much attention was paid to decompositions of market graphs. Methods have been developed for selecting clusters corresponding to certain groups of stocks. Considerable attention has been given to the study of the dynamics of stock markets, including predictions of major crises in them.

Along with such general problems, of great interest are the problems of short-term analysis of the stock market, including the development of algorithms for daily trading in the stock market that provide positive financial results. As an example, we point to publications [6–9] devoted to such algorithms. In these works, as in most studies of the stock market, the value of shares is described by random processes.

The choice of shares for trading in the next days is defined by various deterministic mechanisms that process known prices in the previous days. Naturally, the results obtained depend both on the probabilistic models used and on the rules of preliminary data processing. It is not the purpose of this paper to review in any detail the numerous publications in this direction. We can only note that the great variety of methods, problem statements and algorithms most likely indicate a certain dissatisfaction with both the existing models and the financial results obtained on their basis.

A. Rubchinsky (✉)
National Research University (HSE), Moscow, Russia

K. Baikova
National University of Science and Technology (MISIS), Moscow, Russia

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_20

The presented paper uses the reverse scheme in a sense. No probabilistic considerations about share prices are assumed simply because stock market processes can be considered as probabilistic with a great reserve. At the same time there is no doubt about their uncertainty. The use of probabilistic models and methods to analyze systems under uncertainty may be successful in some cases, but there are no guarantees in the general case. However, not in the analysis of the raw data, but in the proposed trading algorithm itself, simple probabilistic methods are used, which – although theoretically – allow us to apply the central limit theorem and obtain results with some appropriate reliability. The presentation of the proposed approach to the development of a daily trading algorithm is the content of this paper.

In more detail, it is possible to imagine the general scheme of the daily trading algorithm consisting of four consecutive blocks:

1. Analysis of market data for a current day and for a certain number of previous days.
2. Building a group of shares on the basis of this analysis, which is advisable to trade the next day, or recommendation to skip trading the next day, which is quite rare.
3. Calculation of profits (or losses) as a result of trading the next day.
4. Making a decision to stop trading for a certain period, depending on the results achieved during the previous period.

Let us pay attention to the essential difference between one-day trading skipping and cessation of trading for a more or less noticeable period. The next day's trading is skipped when negative results are expected, while the termination of trading for some appreciable period is performed according to results already achieved (without forecasting). This approach, previously unseen, is proposed – among other new algorithms related to the first three blocks – in this article.

Let us point out the main stages of the proposed general scheme. Two decompositions relating to today and yesterday are constructed to analyze previous data, i.e. the clusters, into which the set of all shares is divided today and yesterday are determined. For this purpose, we use the previously developed method of frequency decomposition of graphs [1–3], applicable to arbitrary undirected graphs. Here it is used to decompose the market graphs introduced in the article [9] – graphs constructed on the basis of correlation matrices for all the shares presenting in the market.

Then we consider the intersection of clusters from the two constructed decompositions, consisting of the maximum number of shares and at the same time dense enough (in the sense of the average closeness of the shares included in them, i.e. the closeness to 1 of the correlation coefficients for most pairs of shares presenting in the market on both days and earlier). One would hope that for most of these shares with similar behavior, the pattern of price changes (i.e., their decreases or increases) would remain the same tomorrow as today. To what extent this hope is justified and what to do if it does not materialize, is shown in this article.

The calculation of profits and losses does not require any special methods, because after the close of trading, all of tomorrow's prices are known.

Table 1 Annual and accumulative incomes for 1990–2010

Year	Annual income, $	Added with previous years, $
1990	110	110
1991	−50	50
1992	84	134
1993	150	284
1994	53	337
1995	189	526
1996	376	902
1997	254	1156
1998	292	1448
1999	−4	1444
2000	−380	1068
2001	585	1653
2002	596	2249
2003	−265	1984
2004	160	2144
2005	−10	2134
2006	107	2251
2007	127	2378
2008	−12	2366
2009	−174	2204
2010	−18	2186

The last stage shows what should be done in the general case when the monotonicity assumption is not realized. Great losses can be avoided by means of a fairly simple stopping algorithm, which is well known in other situations, but has proven useful in the analysis of the stock market.

The proposed trading scheme was used for daily operations (considering the skipping of some days and the cessation of trade for some periods not exceeding one quarter) from 01.01.1990 to 31.12.2010, i.e. 21 years.

These numbers in Table 1 show the gains (or losses) for each year as well as accumulated incomes till every year. The amounts may seem insignificant, but each share is used no more than 5 times in daily trading. Increasing the number of the same shares will correspondingly increase the income.

Obtained experimental results together with regression line are also given in the graphic below (see Fig. 1). It clear demonstrates the time dependence of gains increase, close enough to the linear one.

The material of the article is structured as follows.

- The Introduction gives the idea of the proposed approach to the development of everyday trading scheme in the stock market.
- Section 2 demonstrates the preliminary description of the suggested scheme and present the informal algorithm of this scheme.

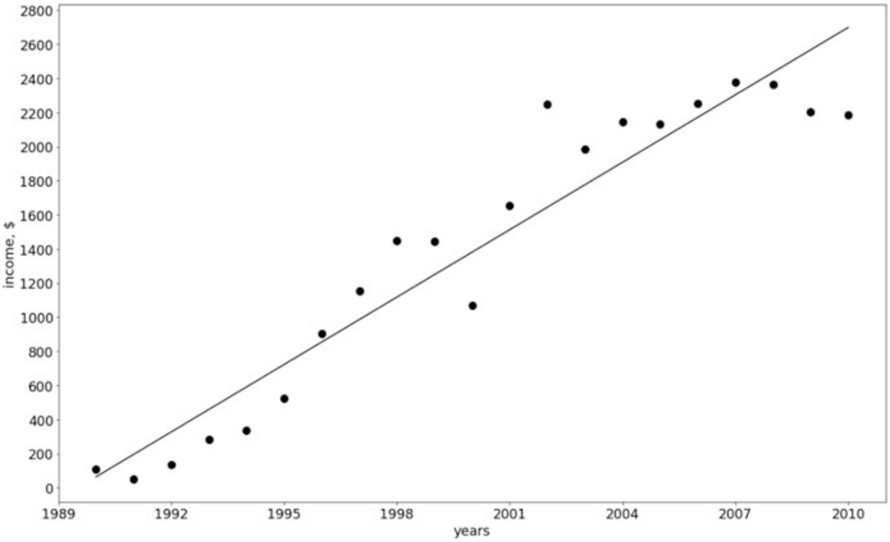


Fig. 1 Time dependence of accumulated income

- Section 3 is devoted to the detailed description of the essential algorithms included in the general scheme.
- Material in Conclusion indicates possible modifications of the considered approach aimed at increasing its efficiency, and summarized the main results of the article.

2 Preliminary Description of the Suggested General Scheme

Let us give a meaningful (not completely formal) description of the trading scheme under consideration. A period of one quarter is chosen for detailed independent analysis. The first day of the quarter has index 0. All operations start from day 1 and continue no later than the penultimate day of the quarter.

We denote the current day by the index *t*. After the market closes, we analyze data on the prices of all stocks at the closure time today (on day *t*), yesterday (on day *t*–1), as well as on 15 preceding days, which can be partially earlier than the beginning of the given quarter.

Within the framework of the trading scheme under consideration, operations related to day *t* are performed after the market closure on day *t* (steps 1–9), when at the time of market functioning on day $t + 1$ (step 10), and after the market closure on day $t + 1$ (steps 11–13).

Below 4 algorithms, named as A, B, C, D, are mentioned. Their detail descriptions and all the encountered below notations, are presented in Sect. 3. Here we describe only the structure of the considered trading scheme.

It is assumed that we know the prices of all the shares at the closure time of trading on day *t* and in the previous 15 working days.

The flowchart of the scheme is presented in Fig. 2. Its short description is given below.

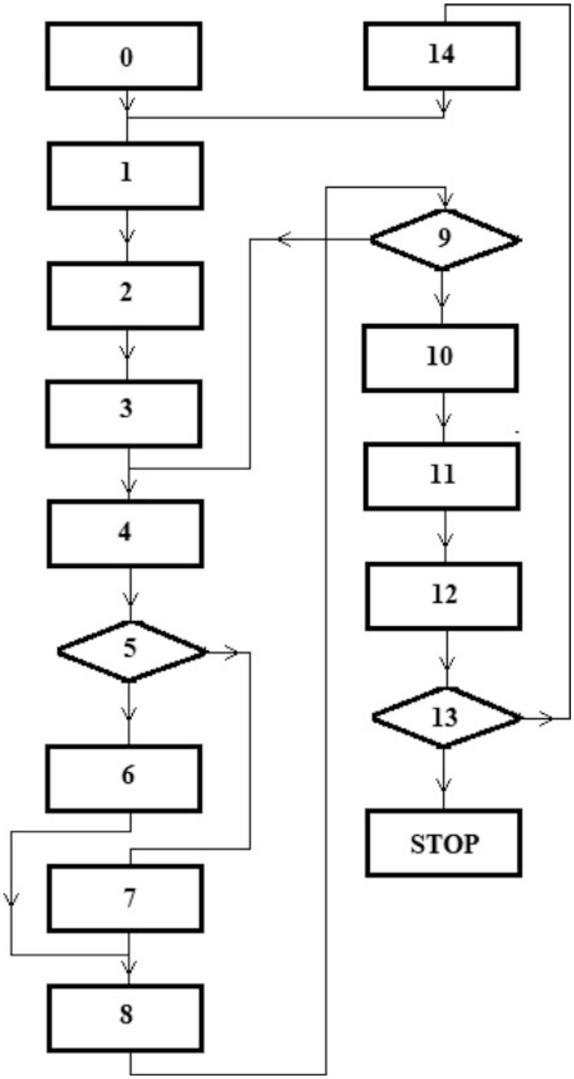


Fig. 2 Flowchart of the general scheme

The Concise Description of the General Scheme

Step 0. Initialization. Specify operations, which are to be executed before the repeated steps start at $t = 1$.

Step 1. Specify the complete of all the shares which are traded in the market on day t and in the previous 14 working days.

Step 2. Immediately after the closure of trading on day t, algorithm A constructs in parallel 5 decompositions $L_0(t) - L_4(t)$ of the set of shares traded on day t.

Step 3. $k = 0$.

Step 4. Algorithm B constructs the set of shares $M_k(t)$.

Step 5. Checks the condition $|M_k(t)| < 200$. If it is satisfied, go to step 7. Otherwise, go to step 6

Step 6. Decomposition $L_k(t)$ is not considered further, the income $H_k(t + 1)$ for day $t + 1$ is defined as 0, and the accumulated income $G_k(t + 1)$ is defined equal to $G_i(t)$. Go to step 8.

Step 7. Add $M_k(t)$ to the remaining group of sets.

Step 8. $k = k + 1$

Step 9. If $k < 5$ go to step 4. Otherwise, go to step 6

Step 10. Trade shares of the remaining groups $M_k(t)$. These shares are traded on the next day $t + 1$ at the opening and closing of the stock market according to Algorithm C.

Step 11. After closure time at the same day $t + 1$ actual income $H_k(t + 1)$ for day $t + 1$ and actual accumulated income $G_i(t + 1)$ are calculated for the shares of set $M_k(t)$, belonging to the remaining group.

Step 12. The accumulated incomes are added up. The result is denoted and saved as $S(t + 1)$. This number is the impotent indicator of the proposed trading method.

Step 13. Algorithm D produces one of two decisions depending on the value of $S(t + 1)$:

- Go to step 14
- Stop trading until the beginning of the next quarter.

Step 14. Continue trading for the next value $t = t + 1$, according to the steps 1–13 of the above-described general scheme.

It is quite clear that the considered scheme demonstrates a very small part of the opportunities provided by the stock market. We accentuate that the main goal of this work is not so much to develop a specific gaining trading algorithm, as to experimentally justify the existence of such algorithms.

3 Detailed Description of Algorithms

In this section we consider separately the steps 0–14 from the general scheme (see the previous section). All the required notations are introduced as it appears. Evident steps like $t = t + 1$ are not considered here. They were presented in the general scheme above in Sect. 2.

Detailed Scheme of the Approach

2.0. Initialization

2.0.1. Prices matrix determination. Every day from the considered period 1990–2010 shares of 500 greatest companies in USA were traded in stock market S&P-500 (in 2014 this number was increased to 505). The prices (at closure time) of all the shares in the S&P-500 stock market are extracted from Blumberg database.

Rather rare changes in the list of the 500 shares in 14 days preceding day 0 can still occur. Since we will need correlation coefficients for 15 days, including day 0 as the last day, it is necessary to "align" the list, that is, the final list should include the maximal possible number of the same shares that were traded for all 15 days ending in day 0. Recall that day 0 is the first working day of the quarter in question. There are usually from 490 to 500 such shares. This number is further denoted as n. The prices of these shares at closure time form the required matrix with n columns and 15 rows. It is designated as $P(0)$. Pay attention that only data known before day 0 and day 0 itself is used in construction matrix P, which is the output of step 2.0.1. Pay attention that the number 15 is one of parameters used in the considered algorithms.

2.0.2. Calculation a matrix of distances between shares for day 0. The matrix R of pairwise correlation coefficients is calculated basing on the above constructed matrix of prices. Its elements r_{ij} are equal to standard correlation coefficient between i-th and j-th columns of matrix P ($i, j = 1, \ldots, n$). Distance d_{ij} between two shares (say, i and j) is defined by the formula $d_{ij} = 1 – r_{ij}$, where r_{ij} is the correspondent element of matrix R. The determined distance d is close to 0 for «very similar» shares and is close to 2 for «very dissimilar» shares. Therefore, matrix $D = (d_{ij})$ is considered as the dissimilarity matrix. For convenience we put $d_{ii} = \infty$ ($i = 1, \ldots, n$). This will avoid the appearance of loops at the vertices in all further algorithms. By the construction $d_{ij} = d_{ji}$ (symmetry). This matrix D is the output of step 2.0.2, while matrix P is its input.

2.0.3. Building a market graph for day 0. From a detailed description of the original system by the distance matrix, it is easy to obtain its more concise (but in many cases no less useful) description by a graph. Here is – just for the sake of clarity – a well-known algorithm of building this graph, commonly called the neighborhood graph. Because our initial system is a stock market, its neighborhood graph is often named as market graph.

Algorithm for constructing a neighborhood graph.

The input to this algorithm is an $n \times n$ distance matrix D and a counting number $k > 0$. Assume $k = 4$ (another parameter of the general algorithm). Let us recall that n denotes the number of objects in the system under consideration (number of shares in day 0).

Step 1. Define an $n \times n$ integer matrix A and set all elements a_{ij} equal to 0 in it.

Step 2. For each $i = 1, \ldots, n$, perform the following operations.

2.1. Define d_k equal to the distance from i to its k-th nearest neighbor.
2.2. Let $a_{ij} = 1$ and $a_{ji} = 1$ for all j such that $d_{ij} \leq d_k$.

The graph G, whose adjacent matrix is the constructed matrix A, is the output of the step 3.1.3 of the considered algorithm, while distance matrix D is its input.

2.0.4. Construction of 5 decompositions of the market graph for day 0 into 12 subgraphs. This algorithm (designated as A) is the central in the suggested general algorithm. It required a special attention as well as a significant volume, exceeding the volume requiring for exposition of all the other steps together. At the same time this method of graph decomposition is applicable for arbitrary graphs and can be used in many situations far from stock markets.

Therefore, this material is presented as special section in the Appendix.

2.0.5. Constant for stopping rule. Assume $Q^{-} = -25$ and $Q^{+} = 50$.

2.1. Specify the complete of all the shares which are traded in the market on day *t* and in the previous 14 working days.

This step completely coincides with step 2.0.1.

2.2. Construction of 5 decompositions $L_0(t) - L_4(t)$ of the set of shares traded on day *t*.

This step completely coincides with step 2.0.4 relatively to data for day t.

Step 3 (see page 5) is clear.

2.4. Construction of the set of shares $M_k(t)$. Input of this algorithm B incudes already known decompositions $L_k(t - 1)$ and $L_k(t)$. Its output consists of the set of shares $M_k(t)$. Algorithm B is described in the Appending B.

Steps 5–9 (see page 6) are clear.

2.10. Algorithm C. Its input includes a set M of shares, presented on the market in both days t–1 and t (index k is omitted for simplicity). This set was found at item 2.4. It also includes price matrices $P(t\text{–}1)$ and $P(t)$ for days t–1 and t, used at item 2.4. Let us go directly to the trading algorithm. Consider this set M and shares from set M. For each of them, one of three situations is possible, known at the end of day t:

1): c(t–1) > c(t),
2): c(t–1) < c(t),
3): c(t–1) = c(t),

where $c(t)$ and $c(t\text{–}1)$ denote the share price at closure time on the current day t and on the previous day t–1.

For shares from the first group, on the next day $t + 1$, it is proposed to sell this share at a price at the market opening, which is assumed to be close to the price $c(t)$

at the market closure time in the day before, and buy it at the market closure time at price $c(t+1)$, which will be in day $t+1$.

For shares from the second group, on the next day $t+1$, it is proposed to buy this share at a price at the market opening, which is assumed to be close to the price $c(t)$ at the market closure time the day before, and sell it at the market closure time at price $c(t+1)$, which will be on day $t+1$.

For shares from the third group (of which there are usually very few or none at all), do nothing.

The essence of the proposed approach lies in the fact that one can hope that the trend of price changes in 1 day will remain the same, and then both operations will bring income. If you do the same with all shares on the market, then the result will coincide with the movement of the entire market, which can hardly be of interest. However, in this case, the shares under consideration belong to the intersection of two clusters of shares with similar behavior, which allows us to hope for a positive result for most of them. It is clear that in many cases the situation is the opposite one, but we partially correct it omitting the trade at some days (see simple steps 5 and 6 in page 6, and algorithm D below).

Note that in both cases the number of shares does not change. In the first case, you need to have one share in order to sell it at the morning, and at the end of the day to buy it, after which there will again be one share. In the second case, you buy one share at the morning and sell it at the evening and have the same number of shares. Since it is not known in advance which case occurs, one must have one copy of each share at the beginning of trading. Further, it is no longer necessary to buy shares, and the presence of one share of each type will be permanent. Sometimes you will have to buy one share when a new company enters the S&P-500 list, which is quite rare. Finally, you can buy shares only as needed, when a new share is included in the considered set M under case 1). We emphasize once again that there is no need to buy such a share further due to the algorithm described above.

Steps 11 and 12 (see page 6) are clear.

2.13. Algorithm D. Stopping rule. At the initialization stage (see item 2.0) two numerical parameters were given, a negative number Q^- and a positive number Q^+. If the accumulated income on day $t+1$ is between Q^- and Q^+, then we let $t = t+1$ and go to the next day (see item 2.1). In the second case no further action is taken till the beginning of the next quarter. The last accumulated income $S(t+1)$ since the beginning of the current quarter till the day $t+1$ is considered as the result of trading in this quarter. The annual sums of these quarter incomes are shown in the left column of Table 1.

The situation is remotely reminiscent of the well-known problem of hiring a secretary. In that problem we had to stop at the already arrived candidate, not knowing whether the next candidate would be better or worse than this one. But in the secretary problem, some probabilistic assumptions about all the girls are used. In our case, there are no assumptions about the next day's incomes.

The two-threshold strategy under consideration is cautious. It seeks not to maximize gains, but to minimize losses. In this case it leads to the fact that the negative number Q^- is noticeably less modulo than the positive number Q^+. Some examples of this strategy are given in Table 2. The point here is that positive results can be obtained even when the result for the whole quarter is sharply negative.

Example 1 The columns in Table 2 show the accumulated incomes during one quarter (in dollars). The corresponding periods are indicated in the column headings. The results of the algorithm (i. e. revenues for the quarter) are in bold italics. Let us take a look at the quarters one by one.

In the third quarter of 1990, acting according to our cautious algorithm, we would have lost 37.81. By continuing to play, we could win 10 to 20 times more. But after all, we do not know the future and do not try to predict it – therefore we console ourselves that our losses are relatively small.

In the second quarter of 2003, we would have lost (according to our algorithm) 48.53. But by continuing to play, we would have lost many times more (see the second column of Table 2).

In the second quarter of 2009, according to the algorithm, we win 110.04, despite sharply negative earnings in this quarter in the future. At the same time, at the beginning of the quarter, you can win much more. But again, we must remember that we do not know and cannot predict the future with any reliability. And the positive gain is determined just by the proposed cautious algorithm.

In the second quarter of 2010, the situation is almost the same as in the previous case. It is worth paying attention to the fact that in the first 7 days, in accordance with the algorithm, we continue to play, since the accumulated winnings lie within the specified limits. And only on the seventh day we get a positive income of 64.97.

Approximately the same situation occurs in the third quarter of 2008. On any day after the third one, all accumulated incomes are large negative numbers in absolute value. Note also that this period immediately precedes the great hypothec crisis. However, in this case, the same algorithm gives a win.

4 Conclusion

Of course, individual examples do not provide any, even experimental, evidence of the effectiveness of the proposed approach. The results of its application for 21 years (84 quarters), presented in Table 1 and Fig. 1, give more confidence. Of course, our model is very crude and it is possible to achieve better results, but the main thing is that even in such a very limited model it is possible to get reliable winnings. That is, a stock market is still not the casino in Monte-Carlo, as some economists and mathematicians believe.

This section briefly discusses some possible modifications of the proposed approach to building a stock market trading algorithm. Their detailed development and experimental analysis are expected to be carried out in the future.

Table 2 Results of the stopping rule

day	1990.03	2003.02	2009.02	2010.02	2008.03
2	1.87	−20.45	***111.04***	28.30	3.86
3	***−37.81***	***−48.53***	−22.36	−16.59	***54.53***
4	−29.28	−48.64	148.25	−9.44	−222.08
5	−30.60	−69.85	−17.76	2.01	−916.43
6	−63.45	−37.14	190.73	17.94	−1039.49
7	−23.80	−71.35	175.18	***64.97***	−1105.21
8	−38.08	−74.88	193.53	106.42	−932.70
9	−16.50	−69.82	49.62	100.08	−868.61
10	−20.38	−3.00	81.91	169.59	−1068.26
11	−12.21	−29.44	220.45	154.55	−872.12
12	2.20	−99.40	147.45	123.23	−906.28
13	−9.97	−119.01	−158.20	200.43	−899.01
14	23.86	−131.40	−163.12	282.49	−1023.60
15	43.19	−104.03	−224.21	311.81	−862.44
16	40.74	−117.17	−132.39	313.97	−968.81
17	45.27	−78.16	−277.57	334.28	−1044.34
18	54.97	−191.34	−228.20	276.57	−1080.09
19	65.04	−131.22	−364.99	354.90	−1078.26
20	99.41	−128.53	−359.86	221.31	−1264.07
21	124.13	−151.42	−377.81	142.00	−1688.44
22	162.42	−195.96	−378.17	6.07	−1583.05
23	254.55	−228.00	−411.95	27.201	−1647.96
24	387.14	−239.65	−452.49	82.94	−1697.20
25	395.83	−265.69	−551.34	320.68	−1642.82
26	391.82	−272.08	−798.60	−292.23	−1766.43
27	435.10	−362.69	−1049.88	−371.70	−1835.13
28	387.94	−278.48	−997.94	−403.57	−1773.20
29	372.00	−288.64	−871.18	−653.62	−1847.60
30	341.46	−280.09	−951.90	−373.94	−1735.40
31	345.53	−323.43	−947.46	−395.17	−1901.70
32	309.71	−315.62	−1081.20	−437.35	−1705.62
33	461.33	−378.93	−1119.23	−367.52	−1736.97
34	462.93	−368.71	−1097.77	90.99	−1685.75
35	514.56	−370.35	−1083.21	−200.04	−1488.36
36	644.86	−348.14	−1073.82	−392.17	−1387.45
37	768.67	−298.61	−1126.20	−466.49	−1393.37
38	694.44	−268.00	−1258.27	−498.34	−1516.16
39	808.70	−280.17	−1300.08	−549.75	−1629.21
40	794.72	−243.74	−1251.13	−615.60	−1574.56
41	740.76	−286.73	−968.94	−598.67	−1614.50
42	694.17	−265.99	−973.64	−622.40	−1527.08
43	621.82	−306.70	−976.48	−495.78	−1194.11
44	638.01	−296.01	−1094.78	−732.76	−1103.36
45	657.14	−240.30	−1118.52	−584.18	−990.84
46	698.90	−240.90	−1094.18	−619.31	−1022.11
47	670.21	−199.15	−1094.18	−636.68	−863.94
48	657.02	−365.78	−1093.69	−679.71	−668.36
49	654.46	−159.07	−1010.27	−639.85	−735.37
50	643.02	−140.38	−1072.76	−657.56	−653.44
51	643.11	−161.73	−1001.57	−694.39	−563.36
52	657.43	−440.17	−917.57	−691.72	−1148.45
53	639.95	−462.85	−900.69	−687.11	−1477.14
54	643.64	−434.17	−913.50	−702.59	−1829.09
55	635.74	−403.30	−892.12	−693.88	−2310.33
56	702.99	−436.16	−914.96	−635.22	−1788.85
57	727.64	−444.14	−916.67	−610.49	−2250.99
58	776.62	−453.86	−926.93	−585.11	−2144.64
59	751.98	−481.22	−773.61	−603.75	−2132.37
60	794.38	−512.26	−787.02	−605.57	−2129.00
61	868.18	−547.67	−842.32	−397.75	−2073.36
62	751.35	−548.38	−911.70	−322.10	−2170.33
63					−2625.77

1. Instead of summing at step 12, one can consider individual accumulated payoffs $S_k(t + 1)$ ($k = 0, 1, \ldots, 4$) and apply the stopping algorithm to them separately. The decision to continue or stop trading is also made separately for 5 processes.
2. With both modifications, it is possible to increase the number of decompositions and, accordingly, the considered groups of shares, etc. (for example, up to 10). Recall that stock prices are not assumed to be random: they are exact inputs. At the same time, the construction of decompositions is a random process that uses a standard random generator. Therefore, the average gains for one quarter obtained by the proposed algorithm (averaging is taken over the number of decompositions −5, 10, etc.) are limited random variables to which the central limit theorem can be applied. Their limiting value can be considered as one of the characteristics of the real process of trade in a given quarter. Strictly speaking, the obtained experimental results show that the sum of the mentioned quarterly averages grows with time. Of course, so far this is not an exact (albeit experimental) statement, but a substantive hypothesis that needs careful verification.
3. When looking at some periods (for example, 4 out of 5 given in Table 2), large accumulated losses in the second half of the periods are striking. This means that for the majority of stocks from the selected groups, the direction of price change is unstable. The algorithm has a parameter $sg = 1$, which indicates our main assumption about the preservation of the sign of the cost change. However, when it is replaced by -1, the winnings will be given by precisely those shares for which the change in value turned out to be nonmonotonic. It is possible that an adaptive strategy of replacing sg with the opposite value can give a noticeable gain, given the rather long periods of time in which the price change has a fixed direction or, conversely, it is unstable. Some analogy is the classic example of controlling the stop of a rocket, the engine of which has only two possibilities − maximum thrust in one of two directions. This example is given in the first chapters of almost all textbooks on the theory of automatic control. However, in our case, special numerical experiments are required.
4. The possibilities of the proposed approach can significantly increase if, instead of a fixed time period of one quarter, we consider variable periods, depending on the already accumulated gains. The same applies to trading within 1 day. The essence of the matter is that it can be assumed that the basic regularities of the functioning of the stock market do not depend much on changes in the scale of analysis, i.e., it has a fractal character. Of course, this also requires detailed research.
5. The material in this article is based on an analysis of data from a single S&P-500 market. It would be important and interesting to apply the proposed approach to other stock markets.

One of the authors (Rubchinsky) is grateful to professor F.T. Aleskerov for his support and professor B.G. Mirkin for many years of attention to his work in the field of data analysis.

Appendix. Algorithm A

Here the considered step 2.0.4 of the algorithm is presented as the algorithm of constructing decomposition of an arbitrary graph into K subgraphs, where K is an arbitrary counting number larger than 1. The initial graph G (see step 2.0.3), distance matrix D, the number K and the repetitions parameter T (a counting number larger than 100) form the input of the algorithm.

The method consists in constructing a binary tree whose vertices correspond to sequentially defined subsets of the set of vertices. The root of the tree corresponds to the set of all the vertices of the graph in question. For each constructed subset of vertices (denoted by X) there are two natural indicators. They are as follows:

The average value $d(X)$ of distances between all pairs of vertices (for one-element subsets X by definition $d(X) = 0$).

The number of elements $N(X)$ in a given subset X. Of course, this number is known. After these preliminary remarks we pass to the direct description of the algorithm.

1. The set X_0 of all vertices of the original graph is taken as the root of the binary tree under construction
2. Consider all the leaves of the already constructed binary tree. Select the leave X contained the maximal number of elements. Let us perform the following operations:

   3.1 Construct a neighborhood graph based on the set of vertices X and the complete distance matrix D (see step 2.0.3 above in Sect. 2).

   3.2. Apply to the constructed graph the frequency dichotomy algorithm (FDA), summarized in subsection FDA below, and construct T dichotomies.

   3.3. The result of applying FDA is a family of pairs of sets (X_1, X_2). If there are pairs, in which both sets X_1 and X_2 consist of more than one element) go to step 3.4. Otherwise (it means that each pair has at least one single element set) go to step 3.5.

   3.4. For each pair (X1, X2), for which both are not single element sets, calculate the indicators d(X1) and d(X2) (once again recall that the matrix D of all pairwise distances is known, so all the calculations are done very quickly). Let us proceed to step 3.6.

   3.5. For each pair (X1, X2), for which one of the sets is single element set, calculate the indicators d(X1) and d(X2) (recall that for any single element set X d(X) = 0).

   3.6. Let us choose the pair (X1, X2) for which the maximum of {d(X1), d(X2)} is minimal.

   3.7. The found subsets X1 and X2 are added to the binary tree as sons of vertices X. Their first estimates are already computed, and the second estimates (number of elements) are known, since the sets themselves are known.

   3.8. If the number of leaves in the constructed tree is less than the input parameter K, return to step 2. Otherwise, go to the last step 4.

   4. Required decomposition is constructed.

FDA The above algorithm uses (at step 3.2) the frequency dichotomy algorithm (FDA). It is the most complicated but inalienable part in the suggested approach. This algorithm was expounded in publications [5, 6] in other environments and purposes, and therefore it seems of expedient to present it here. As the other algorithms, included in step 2.0.4, it is applicable to arbitrary undirect graphs. There are two of above-mentioned input parameters – graph G and repetitions parameter T.

The algorithm constructs the family of dichotomies. It consists of initialization stage and consecutive repetitions of the main stage (described below) whose number T is one of prespecified parameters. Let us give its formal description.

1. **Initialization.** At this stage, each edge of the original graph is associated with a random integer from 1 to 5 inclusive (5 is a parameter of FDA, too). The maximum of these random values is denoted by $F_{\max}$. Further, in the process of performing FDA, the counting numbers assigned to edges e_j of the graph will be denoted by f_j and called the ***frequency*** in the edge e_j.
2. **The main stage**. The input of the main stage is the aforementioned current frequency values in all the edges of the graph and the current value $F_{\max}$. The output of the main stage will be described below (after describing the operations performed on it). The **flowchart** of the main stage is shown in Fig. 3. Below is a detailed description of the steps to be performed.

Step 1. Using a standard generator of uniformly distributed random numbers, two different vertices of the graph are selected.

Step 2. For two selected vertices, Dijkstra algorithm finds the shortest path connecting them. The length of an edge is its current frequency. Path length is equal to the length of its ***longest*** edge, not the sum of the lengths of all its edges. It is well known that Dijkstra algorithm is applicable in such cases, with the only change: when determining the continued path, instead of the sum of the lengths of the initial segment and the added edge, the maximum of the same two numbers is considered.

Step 3. The maximum edge frequency F_p in the path found at step 2 is determined.

Step 4. If Fp < Fmax, then go to step 5. Otherwise, go to step 6.

Step 5. The frequencies are modified: the number 1 is added to the frequencies of all the edges of the path found in step 2. Go to step 1.

Step 6. As in step 5, the frequencies are modified: the number 1 is added to the frequencies of all edges of the last path found in step 2. Only one difference takes place as compared to step 5: the next step is step 7.

Step 7. The maximum frequency in the edges is increased: Fmax $=$ Fmax $+1$.

The output of every execution main stage will be determined by the end of this Section FDA.

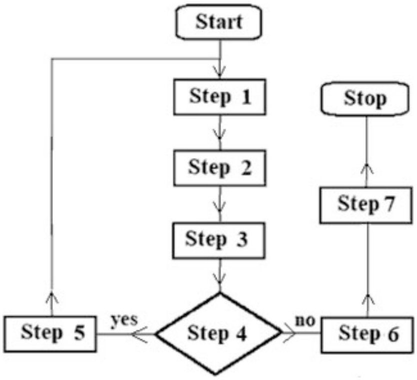


Fig. 3 Flowchart of the main stage of FDA

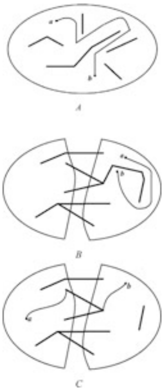


Fig. 4 Cuts and paths in the graph

Let us give some explanations and comments to the main stage of FDA. There are three different cases just before performing the frequency comparison in step 4, denoted as cases *A*, *B*, *C* in Fig. 4. Bold lines represent the edges with the maximal frequency, while the thin lines represent the paths connecting a pair of vertices *a* and *b*.

In case *A* the set of all edges with the maximal frequency does not contain any cut of the original graph. Therefore, the shortest path found at step 2 does not contain edges with the maximal frequency due to the minimax definition of the path length. Consequently, the maximal frequency F_p, found at step 3, is less than $F_{\max}$, and we go to step 5, at which the frequencies in all edges of the found path increase by 1, after which we return to step 1 of the main stage. This consideration is central to this algorithm. Indeed, if the maximal frequency F_p in the edges on the found path is equal to $F_{\max}$, then this means that the set of edges, the frequency of which is equal to $F_{\max}$, contains a graph cut, so the constructed path intersects this cut. If these edges did not contain a cut, then Dijkstra algorithm for minimax would find a path, in which in all the edges the frequency would be less than $F_{\max}$.

In case *B*, the set of all the edges with the maximal frequency does contain a cut of the original graph, but the found path does not contain the edges with the maximal frequency, since both of its ends are located on the same side of the cut. The process continues as in case A.

In case C, the set of all the edges with the maximal frequency contains a cut of the original graph, and the ends of the found path are located in opposite sides of the cut. Therefore, this path has at least one edge that is included in the specified cut. And the frequency in this edge is the maximal, that is, it coincides with $F_{\max}$. Therefore, after the comparison at step 4, the process will proceed in a different way (with steps 6 and 7).

The Output of the Main Stage By the end of the main stage a cut of the initial graph is found. Therefore, if all the edges with maximal frequency were removed from the graph, the number of connectivity components in the remaining graph would be more than 1. We declare the component with the maximal number of vertices as the 1-st part of the next constructed dichotomy of the graph, and declare another component (if there is only one) or the union of all the other components (if there is more than one) as the 2-nd part of this dichotomy. This dichotomy of the set of vertices of the initial graph is the output of the main stage. The frequencies in all the edges and the maximum frequency $F_{\max}$ are also memorized. These values are used in the subsequent execution of the main stage.

We emphasize that the aforementioned deletion is purely virtual and in fact not a single edge is removed from the graph.

The main stage is repeated T times and T dichotomies (some of them can coincide) form the output of step 2.0.4, considered in Sect. 2.

References

1. Rubchinsky, A.: Family of graph decompositions and its applications to data analysis: Working paper WP7/2016/09 – Moscow: Higher School of Economics Publ. House, 2016. – (Series WP7 “Mathematical methods for decision making in economics, business and politics”). p. 60, (2016).
2. Rubchinsky, A.: A new approach to network decomposition problems. In: Kalyagin, V., Nikolaev, A., Pardalos, P., Prokopyev, O. (eds.) Models, algorithms, and technologies for network analysis NET 2016 Springer proceedings in mathematics & statistics, vol. 197. Springer, Cham (2017)
3. Rubchinsky, A.: Graph dichotomy algorithm and its applications to analysis of stocks market. In: Kalyagin, V., Pardalos, P., Prokopyev, O., Utkina, I. (eds.) Computational aspects and applications in large-scale networks NET 2017 Springer proceedings in mathematics & statistics, vol. 247. Springer, Cham (2018)
4. Jallo, D., Budai, D., Boginski, V., Goldengorin, B., Pardalos, P.M.: Network-based representation of stock market dynamics: an application to American and Swedish stock markets. Models, algorithms, and technologies for network analysis. In: Goldengorin, B., et al. (eds.) Springer proceedings in mathematics & statistics 32, pp. 93–106. Springer Science+Business Media, New York (2013). https://doi.org/10.1007/978-1-4614-5574-55
5. Goldengorin, B., Kocheturov, A., Pardalos, P.M.: A pseudo-Boolean approach to the market graph analysis by means of the p-median model. In: Aleskerov, F., et al. (eds.) Clusters, orders, and trees: methods and applications Springer optimization and its applications, vol. 92, pp. 77–89. Springer, Cham (2014)

6. Lohrmann, C., Luukka, P.: Classification of intraday S&P500 returns with a random Forest. Int. J. Forecast. **35**(1), 390–407 (2019)
7. Guo, Y.: Stock trading based on principal component analysis and clustering analysis. IOP Conf. Ser. Mater. Sci. Eng. **740**(1), 012129 (2020)
8. Bruni, R.: Stock market index data and indicators for day trading as a binary classification problem. Data Br. **10**, 569–575 (2017)
9. Fung, P.Y.: Online two-way trading: randomization and advice. Theor. Comput. Sci. **856**(8), 41–50 (2021)

Classification Using Marginalized Maximum Likelihood Estimation and Black-Box Variational Inference

Soroosh Shalileh

1 Introduction: Background, Previous Works, and Motivation

Classification is a popular field of machine learning, with various applications. To this date, various methods have been proposed, one may group them into five categories as follows.

The first category of methods regularly is referred to as the discriminative methods: and the goal is to learn a mapping function so that the difference between the predicted values and the corresponding target values is minimized [13, 29]. The second category of methods is frequently referred to as the generative methods: and the objective is to learn the process which generates data so that the error between the predictions and the corresponding target values is minimized [2, 5, 9, 24].

The third category of methods is designed to learn a set of simple decision rules inferred from data to predict the target values, see [6, 25, 28] for more details. The fourth category of methods aims to combine the predictions of several base estimators (usually from the previous category) to enhance the generalizability power. This category of the method is called ensemble learning methods; Arcing classifier [7] and Random Forest [8] are some the well-known examples of this category.

This fifth category of methods is based on the concept of variational inference. The complex mathematical and implementation aspects of this category and the advances in automatic differentiation have led to the emergence of black-box variational inference (BBVI) [27]. The goal is to reduce the mathematical and

S. Shalileh (✉)
Center for Language and Brain, HSE University, Moscow, Russian Federation

Vision Modelling Lab, HSE University, Moscow, Russia

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_21

implementation complexities while advancing the prediction power. The core idea in BBVI is that instead of obtaining a closed-form solution of an objective function, an integrated sampling technique with automatic differentiation will be applied to compute the stochastic gradient of an objective function for estimating the underlying parameters of a data set.

Our proposed method belongs to this class of methods. More precisely, we assume that there exists a set of latent variables during the data generation process and accordingly we marginalize the conventional maximum likelihood estimation to obtain a new objective function; then we apply BBVI to optimize the newly obtained objective function for enhancing the classification power and taking into account the uncertainty.

It ought to mention that the proposed method of this work differs from the previous works in various aspects as it is described below.

In [9, 14, 24] a Mean-Field variational inference is used to estimate the marginalized likelihood. Furthermore, to reduce the variance, the “control variates” method is applied. The authors obtained a closed-form solution for a specific model in their objective function. On the contrary, we avoid obtaining any closed-form solution for optimizing our objective function. Instead, we apply the so-called reparametrization technique associated with stochastic gradient descent to optimize our objective function.

The proposed methods in [5] and this work both utilize the reparametrization technique and a stochastic optimizer to estimate the marginalized likelihood. However, the objective function in [5] is a function of two arguments: (1) the underlying parameters of data points and (2) the underlying parameters of latent variables. On the contrary, in our objective function, we integrate these two parameters.

The proposed method of [2] is the most similar work from the literature to the current work. To be more precise, in both of these works, the objective function is marginalized over latent variables to obtain a marginalized likelihood estimation. Nevertheless, in [2] after applying the optimality condition, a discrete indicator function is used to reduce the variance of the gradient estimator. And based on this indicator function, they modified the objective function. On the contrary, we use the definition of the derivative of logarithm during the optimization of our objective function. More importantly, the algorithm used in [2] is a random sampling of the data points for computing the gradient of the objective function. Nevertheless, we applied the so-called BBVI.

Noteworthy to mention that we published our preliminary results over four real-world data sets and the moon-shape synthetic data at the IDEAL 2021 conference [30]; however, the current work contains significantly more experimental results. More precisely, in the current work, we study the performance of our proposed method at seven real-world data sets and 80 synthetic data sets to scrutinize the impact of (1) different feature spaces, (2) imbalanced data representation, and (3) dataset sizes.

The rest of this paper is organized as follows. Section 2 describes our proposed method. Section 3 is devoted to experimental settings and experimental results are explained in Sect. 4. Finally, Sect. 5 concludes the paper and explains the future directions.

2 A Marginalized Likelihood Estimation Using Black Box Variational Inference

Let $X = \{x_i\}_{i=1}^N$ be a set of N data points such that $x_i \in \mathbb{R}^V$ is a V-dimensional data point. And let $Y = \{y_i\}_{i=1}^N$ be the set of corresponding labels. Where y_i is in the form K dimensional one-hot vector, that is, $y_i \in \mathbb{R}^K$ where K is the number of classes. And let θ represents the model parameters to be estimated. We assume that during the generation of data points X, there exist a set of latent variables $Z = \{z_j\}_{j=1}^M$. Therefore we can define an objective function marginalized over the latent variables as follows:

$$\begin{aligned}\mathcal{L}_m(\theta) &= -\log \int p(Z|X;\theta)p(Y|X,Z;\theta)\,dZ \\ &= -\sum_{i=1}^{N} \log \sum_{j=1}^{M} p(z_j|x_i;\theta)p(y_i|x_i,z_j;\theta)\end{aligned} \tag{1}$$

In this case, each data point is marginalized with respect to all of the latent variables (of all classes for instance). In order to optimize the proposed objective function (1), one can find an analytical solution for a specific distribution(s): yet there is no guarantee for the model to fit the data properly. More so, for some distributions $p(Z|X;\theta)$ might become intractable.

Therefore, in this work instead of finding an analytical solution for a specific distribution, we adopt BBVI [27] to find an approximation for the proposed objective function. Noteworthy to add that applying optimality conditions directly onto Eq. (1) does not lead to a straightforward solution; thus, we optimize its evidence lower bound (ELBO).

To this end, let us recall the definition of ELBO first. For any two given distributions $r(x)$ and $q(x)$ over the same support, the ELBO of $r(x)$ using $q(x)$ is: $ELBO := E_{q(x)}[\log r(x)] - H(x)$, where $H(x)$ is Shannon Entropy of $q(x)$.

Considering the ELBO's definition and specifying $P(Y, Z|X;\theta)$ and $P(Z|X;\theta)$ as $r(x)$ and $q(x)$ respectively, and applying Jensen's inequality implies:

$$-\log \int P(Z|X;\theta)P(Y|X,Z;\theta)\,dZ \leq -\int P(Z|X;\theta)\log p(Y,Z|X;\theta)\,dZ + H(Z) \tag{2a}$$

$$= -\int p(Z|X;\theta)\log p(Y|X,Z;\theta)\,dZ \tag{2b}$$

Where the RHS of Eq. (2a) is derived from the definition of ELBO. And clearly, $H(Z) = \int P(Z|X;\theta)\log P(Z|X;\theta)dZ$ is, indeed, the Shannon entropy.

By substituting the definition of $H(Z)$ and opening the log of joint probability, i.e., $\log p(Y, Z|X, \theta)$, Eq. (2b) will be obtained.

Let us denote Eq. (2b) with $\tilde{\mathcal{L}_m}(\theta)$ that is:

$$\tilde{\mathcal{L}_m}(\theta) = -\int p(Z|X; \theta) \log P(Y|X, Z; \theta)\, dZ \tag{3}$$

Applying optimality condition on Eq. (3) yields:

$$\begin{aligned}
\nabla_\theta \tilde{\mathcal{L}_m}(\theta) &= \nabla_\theta[-\int p(Z|X; \theta) \log P(Y|X, Z; \theta)\, dZ] &\text{(4a)}\\
&= -\int \nabla_\theta[p(Z|X; \theta) \log P(Y|X, Z; \theta)]\ \ dZ &\text{(4b)}\\
&= -\int \nabla_\theta[p(Z|X; \theta)] \log P(Y|X, Z; \theta)\\
&+ p(Z|X; \theta)\nabla_\theta[\log p(Y|X, Z; \theta)]\ \ dZ &\text{(4c)}\\
&= -\int p(Z|X; \theta)\nabla_\theta[\log p(Z|X; \theta)] \log P(Y|X, Z; \theta)\\
&+ p(Z|X; \theta)\nabla_\theta[\log p(Y|X, Z; \theta)]\ \ dZ &\text{(4d)}\\
&= -\mathbb{E}_{p(Z|X;\theta)}[\nabla_\theta[\log p(Z|X; \theta)] \log P(Y|X, Z; \theta)\\
&+ \nabla_\theta[\log p(Y|X, Z; \theta)]] &\text{(4e)}
\end{aligned}$$

Due to dominated convergence theorem [10], we can push the derivative inside the integral which justifies Eq. (4b). Obviously, Eq. (4c) derived by taking derivatives of multiplication of two functions. Equation (4d) is obtained by replacing $\nabla_\theta[p(Z|X; \theta)]$ with the definition of derivative of logarithm, that is, $\nabla \log f(.) = \frac{\nabla f(.)}{f(.)}$. And clearly, Eq. (4e) is derived from the definition of expectation.

Equation (4e) can be optimized using any Monte Carlo gradient estimators like as Path-Wise Monte Carlo Gradient Estimator [19], which also known as reparametrization, or Score-Function gradient estimator [22, 27], local reparametrization [20], FlipOut [23]. In this work, we adopt the Path-Wise Monte Carlo gradient estimator.

To adopt this estimator, we need to assume that there exists a transformation rule: such that we can draw samples ϵ_i $(i = 1, \ldots, N)$ from a simpler distribution $S(\epsilon_i)$ which is independent of θ and then we can transform this variate through a deterministic path $t(\epsilon_i; \theta)$. Concretely, $z_i = t(\epsilon_i; \theta)$ for $\epsilon_i \sim S$ where S is a parameter-free distribution. This is equivalent to saying that for a given ϵ that comes from a distribution S with no free parameters, it is possible to transform that noise source with a function that depends on the parameters to get a random variable that has the same distribution as the original one. As an example, assume $\epsilon \sim \mathcal{N}(0, I)$, and then do the location and scale transformation of ϵ, that is, $Z = \epsilon\sigma + \mu$ and this implies $Z \sim \mathcal{N}(\mu, \sigma^2)$. This technique is also called the reparametrization technique [19].

Applying the reparametrization technique, i.e rewriting the Eq. (4e) using $Z = t(\epsilon, \theta)$ yields:

$$\begin{aligned}\nabla_\theta \tilde{\mathcal{L}}_m(\theta) =& -\mathbb{E}_{p(Z|X;\theta)}[\nabla_\theta[\log p(Z|X;\theta)] \log P(Y|X,Z;\theta) \\ &+ \nabla_\theta[\log p(Y|X,Z;\theta)]] \qquad (5a)\\ =& -\mathbb{E}_{p(t(\epsilon,\theta)|X;\theta)}[\nabla_\theta[\log p(t(\epsilon,\theta)|X;\theta)] \log p(Y|X,t(\epsilon,\theta),\theta) \\ &+ \nabla_\theta[\log p(Y|X,t(\epsilon,\theta);\theta)]] \qquad (5b)\end{aligned}$$

For the sake of convenient we rewrite Eq. (4e) as Eq. (5a). And Eq. (5b) is the result of applying reparametrization trick on it. In the next step, we can apply Monte Carlo gradient estimation on Eq. (5b). Concretely, (1) we draw samples $\{\epsilon^l\}_{l=1}^L$; (2) we evaluate the argument of expectation using the above set; (3) and finally, we compute the empirical mean of evaluated quantities. Equation (6) explains the Monte Carlo estimate of gradient of objective function:

$$\begin{aligned}\nabla_\theta \tilde{\mathcal{L}}_m(\theta) \approx& \frac{1}{L}\sum_{l=1}^{L} \nabla_\theta[\log p(t(\epsilon^l,\theta)|X;\theta)] \log p(Y|X,t(\epsilon^l,\theta),\theta) \\ &+ \nabla_\theta[\log p(Y|X,t(\epsilon^l,\theta);\theta)] \qquad (6)\\ &where\ \ \epsilon^l \sim S(\epsilon);\ \ Z = t(\epsilon,\theta)\end{aligned}$$

Since, indeed, we maximize a marginalized likelihood estimation we name the base of our proposed algorithm as MMLE. And for ease of notation let $g(.)$ represent the summation argument. And $t(\epsilon^l, \theta)$ is the Gaussian location and scale transform. With this new notation, we summarized our proposed algorithm for optimizing Eq. (6) in Algorithm 3.

Algorithm 3: Marginalized Maximum Likelihood Estimation (MMLE)

```
Input: $X = \{(x_i\}_{i=1}^N$ and $Y = \{y_i)\}_{i=1}^N$: training set
Hyper-parameters: $\alpha$: learning rate
Result: $\theta$: learned parameters
$\theta, t$ initialize the model parameters and step counter respectively
while not converged do
    $\mathcal{M} = \{x_i, y_i\}_{i=1}^M \sim X, Y$; % draw mini-batch of samples $\mathcal{M}$
    $\varepsilon = \{\epsilon^l\}_{l=1}^M$ $(\epsilon^l \sim S(\epsilon)$; % draw samples, $\epsilon^l$, from $S(.)$
    $\theta = \theta + \alpha\frac{1}{M}\sum_{l=1}^M g(.)$; % update rule from Eq. (6)
    $t = t + 1$; % step counter
end
```

It is noteworthy to add that, during the training process, i.e, optimizing $\tilde{\mathcal{L}}_m(\theta)$ the latent variables are considered, therefore the obtained parameters, θ, implicitly contain the impacts of latent variables: consequently, for predicting the labels of a given test data point we can integrate out the term $p(t(\varepsilon, \theta)|X, \theta)$. Concretely, once the model parameters θ are obtained we can use this model to predict the probability of *j*-th test data points using $p(y_j|x_j, t(\epsilon, \theta), \theta)$.

We have implemented our proposed algorithm using TensorFlow Probability [1]. If the linear activation function is applied, we add the suffix “Li” to denote this version (MMLE-Li), and if we use the “ReLu” activation function, we denote it with MMLE-Re.

The source code of our proposed algorithm, as well as other algorithms under consideration and other supplementary materials, can be found in our GitHub repository: https://github.com/Sorooshi/MMLE-by-BBVI.

3 Experimental Setting

To set a computational experiment, one should specify its constituents:

1. The purposes of experiments;
2. The set of data sets for scrutinizing the algorithms under consideration;
3. The set of algorithms under comparison;
4. The set of criteria for evaluation and comparison of the experimental results.

We describe them, in sequence, in separate subsections.

3.1 Purposes of Experiments

We investigate: (1) the impact of the existence of non-informative features at different features dimensionality with the fixed number of informative features; (2) the impact of non-linear data; (3) the impact of the imbalanced data representation with the fixed number informative feature; (4) the impact of data sets size; and, finally, (5) comparison of the performance of the algorithms under consideration over the real-world and synthetic data sets.

Each case mentioned above consists of various settings—as we explain shortly—we repeat each of those settings five times and report the average and standard deviation.

3.2 Data Sets

In order to scrutinize items one to four, we use different synthetic data sets as they are described in the corresponding subsections. We also use seven real-world data sets to compare our proposed method's performance with other works from the literature.

3.2.1 Synthetic Data Sets

1. *Investigating the impact of the existence of non-informative features at different features dimensionality with a fixed number of informative features:* We use the designed scheme in [15] which is publicly available at the python scikit-learn library. We generate normally distributed data points with standard deviation equal to unity about vertices of a two-dimensional cube with sides of length equal to two. After generating the hypercube, next we considered three cases: (A) without non-informative features, (B) with 18, and (C) 198 non-informative features. To make the data sets more challenging, once we generate a data set, we swap 50% of the labels. We set the number of classes and informative features equal to two.
2. *Investigating the impact of non-linear data:* In order to study the ability of algorithms in classifying non-linear data sets, we consider a moon-shape data generator. The source code and more details can be found [26].
3. *Investigating the impact of imbalanced representation:* We investigate the impact of non-informative versus informative features in ba lanced and imbalanced data sets. In imbalanced data sets, 0.95% of data points are devoted to one class while the second class consists of the remaining 0.05%. Noteworthy to add that investigating the impact of features overlap over the balanced and imbalanced data set is considered future work.
4. *Investigating the impact of data set size:* For all the experiments above, two different sizes of data sets are considered, namely, (1) medium-size data sets having 1000 samples; (2) big-size data sets having 100,000 samples. Moreover, five repeats are considered for each of these cases, and the average and standard deviations are reported.

3.2.2 Real World Data Sets

Table 1 describes the characteristics of the seven real-world data sets we use to validate and compare the performance of our proposed method.

Noteworthy is that if a data set contains the categorical feature, those features are converted to one-hot-encoded vectors. Moreover, if the authors of a data set do not provide train and test splits, we use 60% for training the algorithms and the remaining 40% for evaluating the performance of the algorithms under consideration. And This procedure is repeated five times, and the average and standard deviation of the metric in use are reported.

Table 1 The real-world data set's characteristics

Data set	Points	Features	Classes
IRIS [11]	150	4	3
MNIST [21]	70,000	28 × 28 gray-scale image	10
KDD-99 [31]	4,898,431	41	2
German Credit Risk (GCR) [17]	1000	20	2
Breast cancer Wisconsin (BCW) [3]	569	30	
Wine data set [12]	178	13	3
Forest Cover Type (CovType) [4]	581,012	54	7

3.3 Competitors

We compare the performance of our proposed algorithm with three classification algorithms. Below we list those algorithms and the corresponding hyperparameters.

1. Conventional Maximum Likelihood Estimation (C-MLE): a three-layers neural network with the number of units equal to the number of features;
2. AdaBoost [16]: base_estimator=Decision Tree, n_estimators=50, learning_rate=1.0, algorithm='SAMME.R'
3. Classification with Path-Wise Gradient Estimator (CLS PW-Re) [5]:

The learning rate and the number of epochs, in respect, are fixed to 0.01 and 1500. Adam optimizer [18] is used.

3.4 Evaluation Metrics

We use Area Under the Receiver Operating Characteristic Curve (ROC AUC) from prediction scores as the metric to compare the performance of our proposed algorithm with other works from the literature.

4 Experimental Results

4.1 Study the Impact of the Existence of Non-informative Features vs. Fixed Number of Informative Features: Balanced and Imbalanced

The results of this study at medium-size and big-size data sets are recorded in Tables 2 and 3 respectively.

Table 2 Comparison of methods at medium-size Gaussian hyper-cube synthetic data with two informative features. The best results are highlighted in the bold-face font

	Data					
	Balanced			Imbalanced		
	2+0-F	2+18-F	2+198-F	2+0-F	2+18-F	2+198-F
Algorithms	ave(std)	ave(std)	ave(std)	ave(std)	ave(std)	ave(std)
AdaBoost	0.769(0.034)	0.766(0.011)	0.783(0.020)	0.562(0.021)	0.549(0.030)	0.771(0.022)
C-MLE	**0.791(0.028)**	0.783(0.018)	**0.801(0.024)**	**0.588(0.034)**	0.589(0.043)	**0.793(0.014)**
CLS PW-Re	0.387(0.103)	0.266(0.040)	0.318(0.219)	0.495(0.010)	0.485(0.027)	0.277(0.018)
MMLE-Li	0.781(0.034)	0.779(0.011)	0.797(0.027)	0.573(0.037)	0.565(0.020)	0.729(0.064)
MMLE-Re	0.790(0.025)	**0.785(0.019)**	0.797(0.022)	0.576(0.042)	**0.599(0.053)**	0.783(0.016)

Table 3 Comparison of methods at big-size Gaussian hyper-cube synthetic data with two informative features. The best results are highlighted in the bold-face font

	Data					
	Balanced			Imbalanced		
	2+0-F	2+18-F	2+198-F	2+0-F	2+18-F	2+198-F
Algorithms	ave(std)	ave(std)	ave(std)	ave(std)	ave(std)	ave(std)
AdaBoost	0.796(0.004)	0.786(0.008)	0.790(0.007)	0.576(0.004)	0.577(0.007)	0.577(0.003)
C-MLE	**0.798(0.002)**	0.787(0.008)	**0.791(0.007)**	0.575(0.004)	0.576(0.004)	**0.578(0.003)**
CLS PW-Re	0.500(0.000)	0.298(0.055)	0.343(0.129)	**0.579(0.003)**	0.577(0.005)	0.570(0.010)
MMLE-Li	0.796(0.005)	0.772(0.011)	0.790(0.008)	0.575(0.004)	**0.578(0.004)**	0.523(0.059)
MMLE-Re	0.795(0.006)	0.780(0.009)	0.788(0.009)	0.573(0.006)	0.577(0.004)	0.547(0.039)

C-MLE wins the competition of the balance and imbalanced data sets with two and 200 features. Moreover, our proposed method wins the competition when we have 20 features. By comparing the corresponding results of balanced and imbalanced data sets, the performance of all the algorithms under consideration decreases significantly once the representation of data sets becomes imbalanced. More interestingly, one can observe that, on average, the increase in the number of features from 20 to 200 leads to an improvement in the performance of algorithms.

Notwithstanding, the obtained results show the limit of algorithms under consideration. To be more specific, it shows that increasing the number of training samples improves the performance of algorithms (on average). However, it may not necessarily lead to perfect or nearly perfect classification results in the case of having quite a few non-informative features.

4.2 Study the Performance of Algorithms over Moon-Shape Data Set: Balanced and Imbalanced

The results of applying different algorithms on moon-shape data sets are recorded in Table 4.

Table 4 Comparison of methods over moon shape data set. The best results are highlighted in the bold-face font

	Data			
	Balanced		Imbalanced	
	Med.	Big	Med.	Big
Algorithms	ave(std)	ave(std)	ave(std)	ave(std)
AdaBoost	0.881(0.011)	0.899(0.001)	0.873(0.018)	**0.899(0.001)**
C-MLE	**0.898(0.004)**	**0.903(0.001)**	**0.904(0.012)**	0.888(0.001)
CLS PW-Re	0.301(0.096)	0.256(0.122)	0.220(0.075)	0.892(0.001)
MMLE-Li	0.886(0.008)	0.796(0.005)	0.891(0.013)	0.888(0.001)
MMLE-Re	0.880(0.008)	0.795(0.006)	0.887(0.015)	0.882(0.002)

Table 5 Comparison of methods over Real-World Multi-Class data sets. The best results are highlighted in the bold-face font

	GCR	BCW	IRIS	MNIST	CovType	Wine
Algorithms	ave(std)	ave(std)	ave(std)	ave(std)	ave(std)	ave(std)
AdaBoost	0.989(0.010)	**0.988(0.006)**	0.979(0.010)	0.647(0.000)	0.802(0.002)	0.862(0.153)
C-MLE	**1.000(0.000)**	0.969(0.003)	0.995(0.003)	**0.761(0.000)**	0.922(0.001)	0.999(0.001)
CLS PW-Re	0.860(0.095)	0.960(0.011)	0.995(0.003)	0.708(0.010)	**0.972(0.001)**	0.997(0.003)
MMLE-Li	**1.000(0.000)**	0.970(0.007)	**0.998(0.001)**	0.671(0.009)	0.928(0.002)	**1.000(0.000)**
MMLE-Re	**1.000(0.000)**	0.987(0.004)	0.996(0.004)	0.496(0.002)	0.952(0.006)	**1.000(0.000)**

One can easily observe that C-MLE dominates this table, though the performance of the proposed method is also acceptable.

4.3 Experimental Results at Real-World Data Sets

The comparison of the performance of the algorithms under consideration over the real-world data sets is reported in Table 5.

AdaBoost and CLS PW-Re, in respect, win the competition of BCW and CovType data sets. GCR competition has three winners, namely, two versions of the proposed methods of this work and C-MLE. Moreover, our proposed method wins the IRIS and Wine competitions.

5 Conclusion and Future Work

In this paper, we propose a marginalized likelihood objective function by assuming the existence of latent variables during the data generation process. We optimally estimate the data distribution's parameters by applying black-box variational inference. The determination of the parameters of marginalized likelihood estimation allows for the classification of the test data points. We evaluate and compare the proposed method's performance using both fundamental and start-of-the-art algorithms over both real-world and synthetic data sets. This led us to conclude that the proposed method is effective and competitive.

Moreover, we scrutinize the impact of: (1) existence of non-informative features at different features dimensionalities with a fixed number of informative features, (2) different data set size, (3) non-linear data, and (4) imbalanced data representation.

Our experiments show that when the number of non-informative features increases from 2 to 20, on average, the performance of algorithms degenerates. However, when this number increases to 200, the performance of algorithms improves rather significantly.

We also observed that in the case of the balanced data set, although the increase in the number of training samples leads to some relatively subtle improvements in the algorithm's performance, this improvement is still far from the point of having outstanding results, especially in the non-linear data set, like moon-shape data.

Drawing conclusions from our experiments on the imbalanced data sets are not as straightforward as in other cases. Nevertheless, we observe that the increase in the number of training samples, on average, does not improve the performance of the algorithms under consideration.

The list of our future works is as follows:

1. Investigating the impact of applying different distributions instead of standard Gaussian distribution in path-wise gradient estimator;
2. Investigating the impact of applying different transformation rules;
3. investigating the impact of applying different Monte Carlo gradient estimators on our proposed objective function;
4. increasing the computational speed;
5. modifying our proposed method, i.e., the objective function and corresponding algorithm for the task of regression;
6. extending the current implementation of the proposed objective function with a more complicated network structure.

Acknowledgments Support from the Basic Research Program of the National Research University Higher School of Economics (HSE University) is gratefully acknowledged.

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015). Software available from tensorflow.org

2. Ba, J., Mnih, V., Kavukcuoglu, K.: Multiple object recognition with visual attention. Preprint. page arXiv 1412.7755 (2014)
3. Bennett, K.P., Mangasarian, O.L.: Robust linear programming discrimination of two linearly inseparable sets. Optim. Methods Softw. **1**(1), 23–34 (1992). Available online: https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)
4. Blackard, J.A., Dean, D.J.: Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables. Comput. Electron. Agric. **24**(3), 131–151 (1999). Available online: https://archive.ics.uci.edu/ml/datasets/covertype
5. Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D.: Weight uncertainty in neural networks. Preprint. page arXiv:1505.05424 (2015)
6. Breiman, L., Friedman, J., Stone, C.J., Olshen, R.A.: Classification and Regression Trees. CRC Press (1984)
7. Breiman, L.: Arcing classifier (with discussion and a rejoinder by the author). Ann. Stat. **26**(3), 801–849 (1998)
8. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
9. Brodersen, K.H., Daunizeau, J., Mathys, C., Chumbley, J.R., Buhmann, J.M., Stephan, K.E.: Variational bayesian mixed-effects inference for classification studies. Neuroimage **76**, 345–361 (2013)
10. Cinlar, E.: Probability and Stochastics, vol. 261. Springer Science & Business Media (2011)
11. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Ann. Eugenics **7**(2), 179–188 (1936)
12. Forina, M.: An Extendible Package for Data Exploration, Classification and Correlation. Institute of Pharmaceutical and Food Analysis and Technologies (1998)
13. Friedman, J., Hastie, T., Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent. J. Stat. Softw. **33**(1), 1 (2010)
14. Girolami, M., Rogers, S.: Variational bayesian multinomial probit regression with gaussian process priors. Neural Comput. **18**(8), 1790–1817 (2006)
15. Guyon, I.: Design of experiments for the nips 2003 variable selection benchmark. In: NIPS 2003 Workshop on Feature Extraction and Feature Selection, vol. 253 (2003). Available Online: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.make_classification.html
16. Hastie, T., Rosset, S., Zhu, J., Zou, H.: Multi-class adaboost. Stat. Its Interface **2**(3), 349–360 (2009)
17. Hofmann, H.: German credit risk data set (2000). Available online: https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
18. Kingma, D.P., Ba, J.: A method for stochastic optimization. Preprint. cs.LG,:arXiv 1412.6980 (2014)
19. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. Preprint. page arXiv 1312.6114 (2014)
20. Kingma, D.P., Salimans, T., Welling, M.: Variational dropout and the local reparameterization trick. In: Neural Information Processing Systems, pp. 2575–2583 (2015)
21. LeCun, Y., Cortes, C., Burges, C.J.: Mnist handwritten digit database. ATT Labs [Online] (2010). Available: http://yann.lecun.com/exdb/mnist, 2, 2010
22. Mnih, V., Heess, N., Graves, A.: Recurrent models of visual attention. In: Advances in Neural Information Processing Systems, pp. 2204–2212 (2014)
23. Molchanov, D., Ashukha, A., Vetrov, D.: Variational dropout sparsifies deep neural networks. In: International Conference on Machine Learning. Preprint: arXiv:1701.05369 (2017)
24. Paisley, J., Blei, D., Jordan, M.: Variational bayesian inference with stochastic search. Preprint. page arXiv:1206.6430 (2012)
25. Pandya, R., Pandya, J.: C5.0 algorithm to improved decision tree with feature selection and reduced error pruning. Int. J. Comput. Appl. **117**(16), 18–21 (2015)

26. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python (2011). Moon data set: https://scikit-learn.org/stable/modules/generated/sklearn.datasets.make_moons.html
27. Ranganath, R., Gerrish, S., Blei, D.M.: Black box variational inference. In: Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics (2014)
28. Rutkowski, L., Jaworski, M., Pietruczuk, L., Duda, P.: The cart decision tree for mining data streams. Inf. Sci. **266**, 1–15 (2014)
29. Schölkopf, B., Smola, A.J., Williamson, R.C., Bartlett, P.L.: New support vector algorithms. Neural Comput. **12**(5), 1207–1245 (2000)
30. Shalileh, S.: Improving maximum likelihood estimation using marginalization and black-box variational inference. In: International Conference on Intelligent Data Engineering and Automated Learning, pp. 204–212. Springer (2021)
31. Stolfo, S.J., Fan, W., Wenke, A., Prodromidis, L., Chan, P.K.: Cost-based modeling and evaluation for data mining with application to fraud and intrusion detection: Results from the jam project. Results from the JAM Project by Salvatore, pp. 1–15 (2000). http://archive.ics.uci.edu/ml/datasets/KDD+Cup+1999+Data

Generating Genomic Maps of Z-DNA with the Transformer Algorithm

Dmitry Umerenkov, Vladimir Kokh, Alan Herbert, and Maria Poptsova

1 Introduction

Deep learning models such as convolutional neural networks (CNN) and recurrent neural networks (RNN) have been successfully applied to the tasks of genomics to predict various genomic features such as promoters [1], enhancers [2], transcription factor binding sites [3], protein-RNA binding sites [4], splice sites [5], histone codes [6], nucleosome positions [7], and non-coding functional variants [6, 8]. The advantage of CNN is that it recognizes a genomic element of within a region of interest using filters that can be interpreted as DNA motifs. RNN has strength in capturing long-range dependencies in the sequence, but its interpretation is not straight forward.

Since the first key publication on the transformer architecture that uses an attention mechanism to draw global dependencies between input and output in an unsupervised manner [9], this novel approach to learning and prediction has been gradually replacing CNN and RNN in all applications, initially in machine translation area and later in computer vision. Over last 2 years applications with transformers appeared in proteomics and genomics gradually replacing and outperforming CNNs and RNNs.

D. Umerenkov · V. Kokh
Sber Artificial Intelligence Lab, Moscow, Russia

A. Herbert
Laboratory of Bioinformatics, Faculty of Computer Science, HSE University, Moscow, Russia

InsideOutBio, Charlestown, MA, USA

M. Poptsova (✉)
Laboratory of Bioinformatics, Faculty of Computer Science, HSE University, Moscow, Russia
e-mail: mpoptsova@hse.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_22

The first successful applications of transformers in bioinformatics were for protein sequences. A transformer model was used to learn representations of 250 million protein sequences. The high-capacity 34-layer transformer had approximately 670 million parameters, and was trained using data from the three datasets with different evolutionary diversity. Even though learning was unsupervised, the final model was able to identify biochemical properties of amino acids and structural properties of proteins from sequence alone [10].

A similar approach was implemented in ProtTrans [11] that is based on auto-regressive models (Transformer-XL, XLNet) and auto-encoder models (BERT (Bidirectional Encoder Representations from Transformers), Albert, Electra, T5). The model was refined using a protein sequence dataset containing almost 400 billion amino acids. With the learned representations, the model generated highly accurate per-residue prediction of protein secondary structure, protein sub-cellular localization, and membrane and water-soluble proteins.

The next step forward was made in MSA Transformer [12] with an approach based on multiple sequence alignments (MSA) of proteins with relationships learnt from both rows and columns. MSA transformer was used in supervised structure prediction, and attention maps were used to predict protein contacts. The success of transformer architecture culminated in AlphaFold2 that predicts *de novo* 3D protein conformation from an input of the primary amino acid sequences of a previously uncharacterized protein. The algorithm was trained using MSAs and the available collection of PDB structures [13]. The final model, Evoformer has a hybrid architecture consisting of a number of attention-based and non-attention-based components.

Learning representations of DNA sequence also progressed with the use of transformer-based models. One approach called Enformer predict gene expression and promoter-enhancer interactions using DNA sequence information extracted from contiguous genomic regions of up to 200 kb in length [14]. The model based on genomic features extracted by BERT improves performance in predicting enhancers [15] and N6-methyladenine sites when trained in one species and tested in another, consistent with the evolutionary conservation of the processes involved [16].

Further improvement in prediction quality came with pretrained self-supervised BERT-like models. The implementations include DNABERT [17], GeneBERT [18], and LOGO [19]. DNABERT was pretrained on human genome using k-mers representations and then finetuned to improve its power to predict promoters, splice sites and transcription factor binding motifs. DNABERT can also be applied to genomes of species other than the one used in the initial training. GeneBERT is another approach that combines genomic one-dimensional sequences with matrices of transcription factors and corresponding binding regions [18]. The updated model improved classification of promoters, transcription factor binding sites (TFBS), splicing sites, and disease-related regions. LOGO utilizes the same idea of pre-training on k-mers representations but has much lighter architecture than DNABERT. It was tested on promoters, promoter-enhancer interactions, histone modifications and TFBS. It also showed good result in determining non-coding functional variants for both inherited diseases and complex traits or diseases.

All the applications of the above-mentioned transformer models predict regulatory DNA elements that are linear. 3D DNA secondary structures represent another layer of genomic encoding [20]. They act as flipons [21] that launch or suppress genetic programs depending on their DNA conformation that can vary as the context changes. Flipons can form a range of non-B-DNA structures, such as the left-handed Z-DNA and the four-stranded quadruplex structures that depend on their sequence composition. The change in 3D structure enables the switch from one genetic programs to another by engaging proteins that bind specifically to each conformer [22].

Prediction of flipons is a challenge. Machine learning, especially deep learning approaches, helped identify factors that contribute to flipon state. Here we focus on Z-DNA in which the double helix twists to the left rather than to the right as it does in Watson-Crick B-DNA. The transition is driven by the energy produced during processes such as RNA transcription or when nucleosomes or other protein complexes are evicted from DNA. The energy stored by Z-DNA is then available to drive the assembly of protein complexes that perform specific functions. In the best studied example, Z-DNA and its Z-RNA counterpart play a key role in regulating interferon responses and in initiating cellular death pathways [23–28].

ZHUNT is the first model for Z-DNA prediction [29, 30] and is based on the experimentally determined energetic cost of dinucleotide transitions from the B- to Z-conformation as measured *in vitro* [29]. The best Z-DNA forming sequences have an alternating pyrimidine-purine motif, with the d(CG)n sequence flipping most easily. The flip to Z-DNA involves every second base adopting a *syn* conformation with the base pointing back to the deoxyribose ring rather than away from it as in the *anti*-conformation. Of all the bases, guanine adopts the *syn* conformation most easily. The *syn-anti* alternation accounts for the zig-zag nature of the Z-DNA backbone. However, sequences such as GGGG where the pyrimidine base is replaced by guanine will also form Z-DNA. While the energy cost for GGGG is higher, it is still lower than the cost for form Z-DNA from alternating d(AT)n, a sequence that is more prone to form a hairpin as the repeat lengthens. Overall, the major energetic cost to forming Z-DNA is that incurred from forming two B-Z junctions, as this process requires disruption of the double helical structure at both ends of the flipon. Once established, the transition is cooperative and can extend from the site where Z-DNA was nucleated to adjacent regions. Currently, it is uncertain how many Z-DNA forming sequences identified by ZHUNT affect biological function, although enrichment of Z-flipons in promoter regions is reported [30].

Recently we developed DeepZ [31] – a deep learning approach based on CNN and RNN architectures and on the experimental mapping of Z-flipons *in vivo*. The algorithm built the model using information from sequence composition, ZHUNT B- to Z- transition energies and approximately 1000 omic features extracted from the extensive genomewide ENCODE datasets that describe many different cell and tissue states. Performance of the DeepZ model was assessed using human and mouse ChIP-seq data. While the model improves the prediction quality for flipon state *in vivo*, precision is not high, likely reflecting the complex nature of the model and the high correlation between feature sets.

Here we applied state-of-the-art transformer models to the task of Z-DNA prediction using data obtained from *in vivo* experiments that employ Z-DNA specific binding proteins to enrich for Z-DNA (or Z-RNA) forming flipons. We chose DNABERT model pretrained on 6-mers for human genomes that can also be fine-tuned for mouse genome. We explored attention maps of the fine-tuned Z-DNABERT model to learn sequence specificity of Z-DNA regions as well as the surrounding regions and compared model performance with DeepZ and ZHUNT. We also discussed complementarity between the two different approaches, DeepZ and Z-DNABERT.

2 Material and Methods

2.1 Z-DNA Data Sets

ChIP-seq data from Shin et al. for human genome [32] comprised 359 Z-DNA regions covering 134,807 bps. ChiP-seq data from mouse genome [28] comprised 1765 regions with total length of 709,708 bps. Permanganate/S1 Nuclease Footprinting Z-DNA data contained 41,324 regions with total length of 773,788 bp in human and 24,885 regions with total length of 609,476 bases in mouse genomes [33]. All original datasets were filtered for ENCODE blacklisted regions.

For DNABERT the data was preprocessed by converting a sequence into 6-mer representation. Each nucleotide position is represented by a k-mer consisting of a current nucleotide and the next 5 nucleotides. The data was split into 5 stratified folds so we could train 5 individual models with 80% of the data and assess precision and recall using the remaining 20%. Due to the large imbalance between positive (Z-DNA) and negative (not Z-DNA) classes we randomly sampled the negative class with the following ratio: 20 for human ChIP-seq data, 4 for mouse ChIP-seq data, and 2 for Kouzine et al. human and mouse data.

2.2 Benchmark Models

The DeepZ model was run with the set of 1054 omics features as described in [31] for human Shin et al. data set [32] and with the set of 874 omics features as described in [28] for mouse genome from which only 544 are in common. Predictions for the test set and whole genome were done the same way as for DNABERT models.

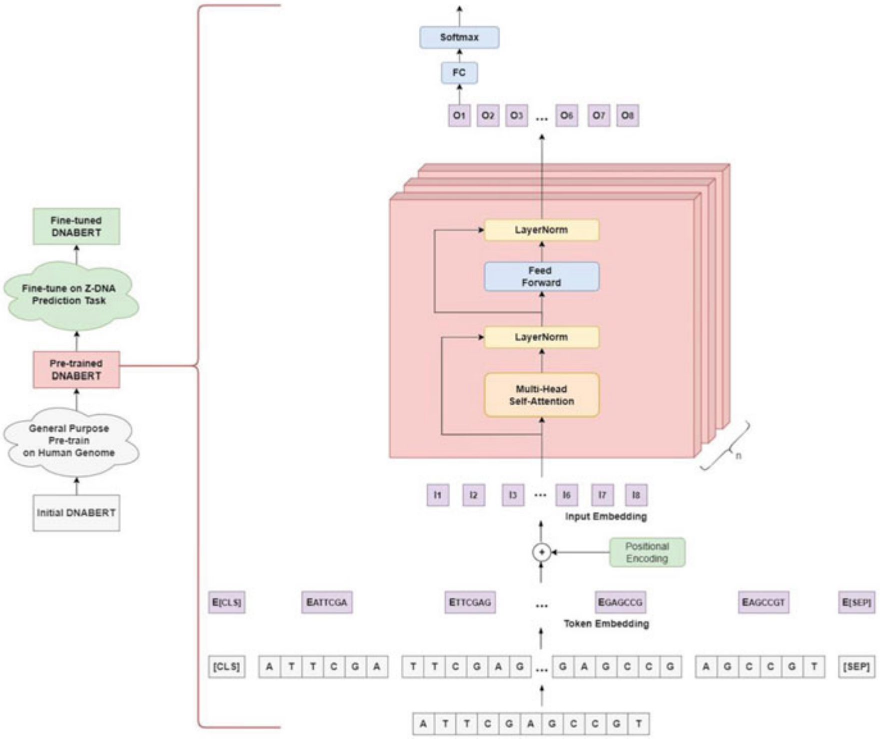

Fine-tuned DNABERT
Fine-tune on Z-DNA Prediction Task
Pre-trained DNABERT
General Purpose Pre-train on Human Genome
Initial DNABERT
Softmax
FC
O1 O2 O3 ... O6 O7 O8
LayerNorm
Feed Forward
LayerNorm
Multi-Head Self-Attention
n
I1 I2 I3 ... I6 I7 I8
Input Embedding
Positional Encoding
E[CLS] EATTCGA ETTCGAG ... EGAGCCG EAGCCGT E[SEP]
Token Embedding
[CLS] A T T C G A T T C G A G ... G A G C C G A G C C G T [SEP]
A T T C G A G C C G T

Fig. 1 Schema of DNABERT used for fine-tuning the model with Z-DNA datasets

2.3 *Fine-Tuning DNABERT Model*

Schema of Z-DNABERT model is presented in Fig. 1. DNABERT was fine-tuned for the Z-DNA segmentation task with the following hyperparameters: epochs =3, max_learnirng_rate = 1e-5, learning_rate_scheduler = one_cycle (warmup 30%) batch size = 24. We trained 5 models, each on 80% of the positive class examples, and randomly sampled negative class examples. For each 512 bp region from the whole genome the final prediction was made by averaging the predictions of the models that used data not seen during training.

3 Z-DNABERT: Fine-Tuning DNABERT for Z-DNA Prediction

Currently there are 4 experimental Z-DNA data sets available: ChIP-seq (Chromatin Immunoprecipitation followed by DNA sequencing of fragments) data from Shin

Table 1 Comparison of Z-DNABERT with other Z-DNA prediction ML models. Predictions are made both on validation set and at the genome-wide level

Validations on the test set:		**Precision**	**Recall**	**F1**	**ROC AUC**
Human Shin et al. *Zaa ChIP-seq on HeLa cells*	DeepZ	0.59	0.56	0.57	0.937
	Z-DNABERT	0.68	0.43	0.53	0.95
Human Kouzine et al	DeepZ[a]	0.011	0.298	0.023	0.893
	Z-DNABERT	0.78	0.89	**0.83**	**0.99**
Mouse Zhang et al. *ZBP1 and Z22 ChIP-seq on MEF*	DeepZ	0.62	0.54	0.58	0.90
	Z-DNABERT	0.57	0.42	0.48	0.97
Mouse Kouzine et al.	DeepZ[b]	0.0006	0.17	0.001	0.678
	Z-DNABERT	0.71	0.79	**0.75**	**0.99**
Whole-genome predictions		**Prec@0.5**	**Recall@0.5**	**F1@0.5**	**ROC AUC**
Human Shin et al. *Zaa ChIP-seq on HeLa cells*	DeepZ	0.11	0.27	0.16	0.915
	Z-DNABERT	0.01	0.48	0.02	0.95
Human Kouzine et al	DeepZ[a]	0.04	0.01	0.02	0.88
	Z-DNABERT	0.12	0.73	**0.20**	**1.00**
Mouse Zhang et al. *ZBP1 and Z22 ChIP-seq on MEF*	DeepZ	0.40	0.16	0.23	0.873
	Z-DNABERT	0.04	0.42	**0.07**	**0.96**
Mouse Kouzine et al.	Z-DeepZ[b]	0.003	0.013	0.005	0.633
	DNABERT	0.05	0.76	**0.09**	**1.00**

4[a]*Deep-Z that was trained on Shin* et al. *data*
[b]*Deep-Z that was trained on Zhang* et al. *data*

et al. for human genome [32], ChiP-seq data from mouse genome resulted from curaxin treatment of mouse embryonic fibroblasts from Zhang et al. [28], and Permanganate/S1 Nuclease (KS1) Footprinting in human and mouse genomes is based on the single-stranded nature of the junctions between B- and Z-DNA [33]. These different methods vary in their resolution. While KS1 maps directly at the level of nucleotides, ChIP-seq is based on pull down of fragments of 100–150 base pairs long and depends on the specificity of the antibody used in the pull-down. Each method therefore supplies different data, with ChIP-seq also identifying other sequences that closely associate with Z-flipons.

We compared performance of Z-DNABERT with our previous machine learning method DeepZ (Table 1). Z-DNABERT showed high performance when taking two metrics into account – F1 and ROC AUC. The highest performance was achieved on the more extensive Kouzine et al. data sets, especially on the whole-genome predictions.

When comparing the test set and whole genome prediction results, the recall metric does not change much, so the models correctly find all the regions labeled as Z-DNA. Meanwhile, the precision drops sharply, indicating many false positives in the model's predictions. These false positives could be predictions of novel potential

Z-forming regions that were not detected under the experimental conditions used for mapping as only a subset of all-possible Z-flipons is active in the cell line used. Supporting this idea is the higher precision of the Kouzine data compared to the smaller ChIP-seq data set. The former has more nucleotides labelled as Z-DNA 0,02% (815 thousand out of 3 billion) compared to the ChIP-seq data 0,004% in human ChIP-seq data (136 thousand out of 3 billion nucleotides). Also, very high ROC-AUC metrics on whole-genome data show that the model false-positives have probability scores consistently lower than true positives, which could indicate that the regions detected as false-positive are actually regions which have a lower probability of forming Z-DNA in the cells tested.

Learning Z-DNA Sequences from Attention Maps

It was noticed experimentally that CG/TG/CA repeats are more prone to flip from B- to Z-conformation. However, the detailed analysis of experimentally determined Z-DNA regions showed that other sequences also form Z-DNA, including sequences such as GGGG where the pyrimidine base is replaced by guanine. Transformer architecture allows interpretation of important features by analysis of attention maps. Results can be interpreted according to the difference in the expected frequency of k-mers in the input sequence versus their rank in the output and compared to the frequency in the genome or in the genomic region of interest. This approach is helpful for assessing ChIP-seq data, as *a priori*, the distribution of ZHUNT predicted Z-flipons in the genome is highly enriched in promoter regions. Many sequences associated with promoters, such as TATA boxes or GC rich segments will have high frequencies in the pull-downs independently of their ability to flip to Z-DNA.

The distributions of 6-mers according to their rank in the attention map are given in Table 2. When the model is learning it pays attention not only to the k-mers inside Z-DNA regions but also to the k-mers in the flanking regions. For example, according to attention ranking k-mer GGGGAA is the seventh most frequent that the model uses to define Z-DNA, however this k-mer is the 40th according to the frequency of occurrence inside Z-DNA regions. Also, k-mers GGGGAA CAGGGA TGGGGA GGGGGA AGGGAG GGGAGC are rarely at the site of Z-DNA nucleation, they likely can propagate the flip to Z-DNA once it is initiated. As predicted by this model, these non-canonical Z-forming sequences are frequently associated with alternating pyrimidine/purine sequences that nucleate the flip in conformation. Use of such contestual information improves Z-DNA prediction by Z-DNABERT.

Visualization of attention maps of a short Z-DNA region from Kouzine et al. data is given in Fig. 2 and for a longer region from Shin et al. data – in Fig. 3. While Z-DNA formation in the longer region in Fig. 3 is slightly more costly energetically due to the replacement of pyrimidines by purines, once flipped, it will accumulate energy from that shorter Z-DNA sequences in the region and revert them back to B-DNA [34]. The propensity of both short and long regions to form Z-DNA is supported by the ZHUNT score that is shown overlapped with attention maps.

Table 2 6-mers with top 21 attentions vs frequencies

	hg Shin et al		hg Kouzin et al		mm Zhang et al		mm Kouzin et al	
Att	6-mer	Freq	6-mer	Freq	6-mer	Freq	6-mer	Freq
1	TGTGTG	1	GCGCGC	1	AAGAAG	1	CACACA	2
2	GTGTGT	2	GTGTGT	5	AGAAGA	2	TGTGTG	1
3	CGCGCG	4	CGCGCG	2	GAAGAA	3	GTGTGT	3
4	GCGCGC	3	ACACAC	6	CTTCTT	4	ACACAC	4
5	CACACA	5	TGTGTG	3	CTCGAG	13	CGCGCG	6
6	ACACAC	6	GCGCGG	7	TCTTCT	5	GCGCGC	5
7	***GGGGAA***	40	CACACA	4	CACACA	7	GTGTGC	7
8	***AAAAAA***	17	CCGCGC	10	CAGCAG	6	ACACAT	9
9	CA***GGG***A	43	***GGG***CGC	11	TCCTCC	9	AT**GTGT**	11
10	GTGCGC	11	GCG***CCC***	12	TTCTTC	8	GCACAC	8
11	T***GGGG***A	331	GTGCGC	17	GGAGGA	12	ATACAC	65
12	***GGGGG***A	39	GGCGCG	9	***CCCGGG***	11	**GTGT**AT	43
13	GCTGGG	9	GTGTGC	14	***AAAAAA***	14	TACACA	61
14	GTGTGC	7	GCGCAC	19	CTGCTG	10	GGCGCG	12
15	TGCGCG	8	GCACAC	15	CTGCCT	19	TGCGTG	14
16	TGCATG	21	GCCCGC	20	CAGAGG	16	GTATGT	63
17	GGGAAG	33	GCGGGC	16	GT***CCC***G	18	CACACG	27
18	AGGGAG	429	CGCGCC	8	CTTAGG	21	ACATAC	55
19	GGGAGC	458	GCGTGC	25	GCCGGC	25	CATACA	52
20	AGAAAG	38	GCACGC	26	AGGAGG	29	CACATA	38
21	***GGGAAA***	80	***CCC***GCG	18	CTCTGG	23	CACGCA	19

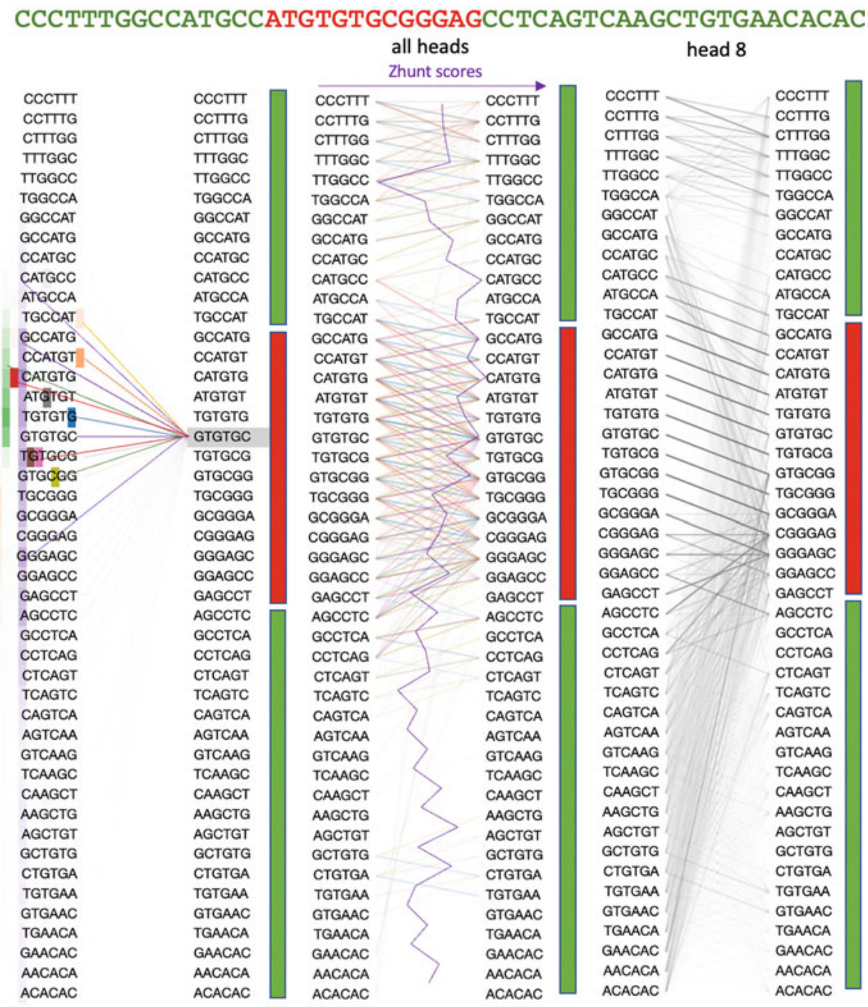


Fig. 2 Attention-head plots of a Z-DNA region from Kouzine et al. human dataset. Z-DNA – red, B-DNA – green. Left column – attention from Z-DNA prone GTGTGC to the surrounding regions. Middle column – attention from all 12 heads. Right column – attention from the head 8, that paid attention to the flanking regions to define Z-DNA region. Violet line in the middle plot depicts relative fluctuations of Z-score predicted with ZHUNT

4 Z-DNABERT Cross-Species Predictions

We tested how well Z-DNABERT model trained on one genome can predict Z-DNA regions in another genome. Table 3 show the results of the model performance that was trained on mouse and then applied on human genome using Kouzine et al. data sets. Performance metrics remain high. The provided Z-DNA prediction tool allows

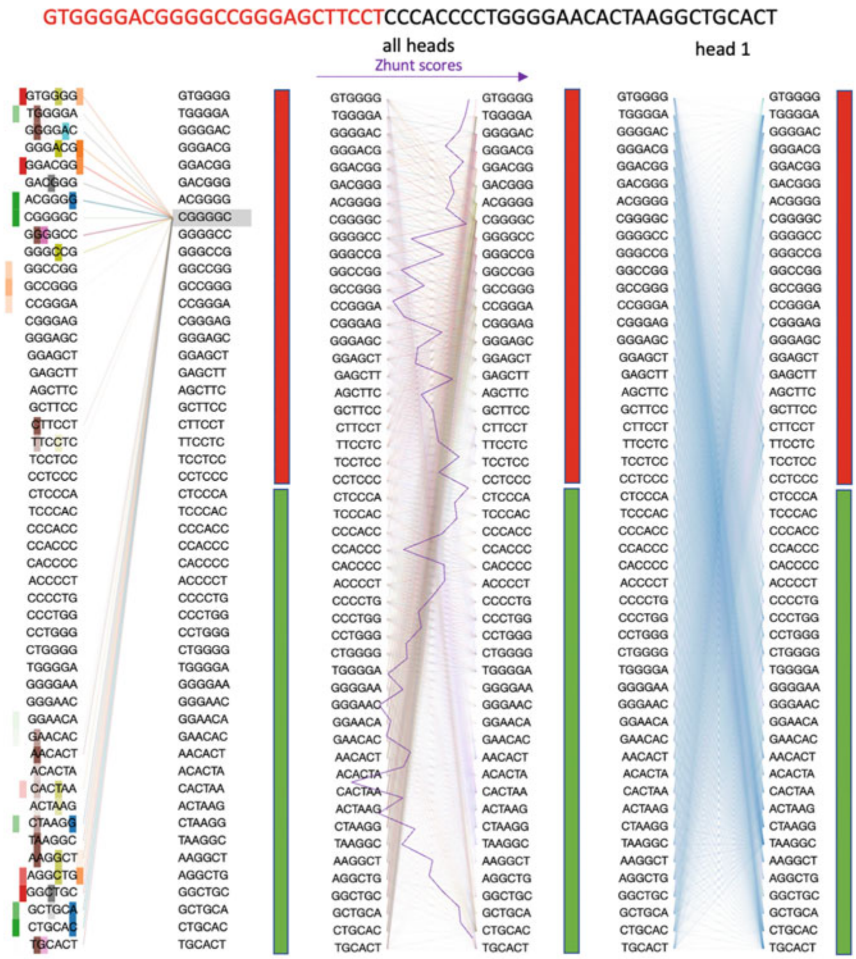


Fig. 3 Attention-head plots of a Z-DNA region from Shin et al. human dataset. Z-DNA – red, B-DNA – green. Left column – attention from CGGGGG to the entire region. Middle column – attention from all 12 heads. Right column – attention from the head 1, that paid attention to the flanking regions to define Z-DNA region. Violet line in the middle plot depicts relative fluctuations of Z-score predicted with ZHUNT

Table 3 DNABERT cross-species predictions

Trained	Predict	Prec	Recall	F1	ROC AUC
Human Kouzine et al.	hg Kousine et al.	0.78	0.89	0.83	0.998
Mouse Kouzine et al.	hg Kousine et al.	0.70	0.87	0.77	0.993

a user to input sequence into four pretrained model to identify Z-flipons with a high level of confidence.

5 Discussion

Determining functional Z-DNA regions is not a trivial task and has been resolved by various approaches starting with the ZHUNT algorithm that is based on thermodynamic and biophysical model of energies measured *in vitro*. Newer models like DeepZ employed deep neural network models trained on DNA sequence, its biophysical properties, and *in vivo* effects measured with omics data. Here we propose another approach that differ both from ZHUNT and DeepZ. The approach takes use of a next generation machine learning approach based on the transformer algorithm implemented in DNABERT. We tuned the model, which we call Z-DNABERT, for detecting Z-flipons with experimental Z-DNA data derived from human and mouse genomes. Based only on DNA sequence, with contextual learning both from KS1 nucleotide resolution data and ChIP-seq fragments, DNABERT-Z outperformed DeepZ and showed high performance on cross-species predictions. A further advantage of Z-DNABERT approach is we can make use of omics datasets to identify features associated with Z-flipons as the data was not used for training, whereas previously they were incorporated into the DeepZ model. Further, use of datasets like KS1 allows localization of the epigenetic features at nucleotide resolution, something not possible with DeepZ. The Z-DNABERT approach also allows for generalization, allowing us to develop a user-friendly interface where Z-flipons can be predicted for the genome of interest with a single Z-DNABERT pretrained model.

Transformer architectures allows for interpretation by highlighting tokens (DNA k-mers in our case) to which the model paid high attention. Analysis of the top high-attention k-mers in all of the four models trained on different Z-DNA data sets revealed a regularity not only in Z-DNA but also in surrounding regions. In the agreement with earlier studies, TG/GT repeats and their complimentary counterparts CA/AC followed by GC/CG are the top-six most frequent in all datasets except for Zhang et al. where Z-flipons were identified in promoter of L1 LINE elements by ChIP-seq using the Z-DNA specific Z22 antibody. These repeats are both most frequent and have high attention. Of interest are repeats that have high attention but they are not the most frequent, such as GGGGAA k-mer, which is the seventh by attention and 40th by frequency in the Shin et all data set or CTCGAG, which is fifth by attention and 13th by frequency in Zhang et al. dataset. Based on the measurements of Shing et al. and the ZHUNT energetics, these sequences are will require higher energy to form Z-DNA under physiological conditions. It is also likely that Z-DNABERT incorporates regions that are prone to form B-Z junctions as they greatly affect the energetics of the flip. Also, Z-prone regions may be associated with other sequences that are bound by architectural proteins like those that contain HMG domains. Such proteins bind to A-rich regions and induce them to bend, providing a barrier preventing the transmission of supercoiling to adjacent regions. These bends may then promote Z-DNA formation by trapping negative supercoils that arise in active promoter regions. In the presence of polyG repeats, Z-DNA formation may compete with folding of other non-B-structures like G-quadruplexes, with the outcome depending on whether the energy is sufficient to initiate formation of one flipon or the other. Indeed, the possible co-occurrence of Z-flipons and G-flipons is supported by the Kouzine et al. data [35]. The finetuned Z-DNABERT showed good performance in this situation as it was able to distinguish the higher frequency of k-mers enriched in Z-flipons compared to that of other k-mers present in the DNA from ChIP-seq experiments.

DeepZ and Z-DNABERT are complementary approaches. DeepZ learns from omics signals (mostly from ChIP-seq signals for TFs and histone marks) that often produce broad peaks due to the size of the fragments analyzed while Z-DNABERT can exploit the nucleotide resolution of the Kouzine et al. data. Omics signals detected by DeepZ can then be further explored with DNABERT-Z to identify more precise boundaries of a potential Z-flipon.

The full extent of gene regulation by Z-DNA is still largely unknown. We lack experimental data for the majority of cell types, tissues and species, with little information on the effect of perturbations. The advances in deep learning models can take what is available and provide insights into the underlying biology by analysis of large orthogonal datasets. The pre-trained transformer model DNABERT, fine-tuned on the experimental data provides a framework for whole-genome Z-DNA annotations and makes predictions that are testable at the bench.

Acknowledgements The publication was supported by the grant for research centers in the field of AI provided by the Analytical Center for the Government of the Russian Federation (ACRF) in accordance with the agreement on the provision of subsidies (identifier of the agreement 000000D730321P5Q0002) and the agreement with HSE University No. 70-2021-00139.

References

1. Umarov, R.K., Solovyev, V.V.: Recognition of prokaryotic and eukaryotic promoters using convolutional deep learning neural networks. PLoS One. **12**(2), e0171410 (2017)
2. Lim, A., Lim, S., Kim, S.: Enhancer prediction with histone modification marks using a hybrid neural network model. Methods. **166**, 48–56 (2019)
3. Zhang, Y., Wang, Z., Zeng, Y., Zhou, J., Zou, Q.: High-resolution transcription factor binding sites prediction improved performance and interpretability by deep learning method. Brief. Bioinform. **22**(6), bbab273 (2021)
4. Ben-Bassat, I., Chor, B., Orenstein, Y.: A deep neural network approach for learning intrinsic protein-RNA binding preferences. Bioinformatics. **34**(17), i638–i646 (2018)
5. Zuallaert, J., Godin, F., Kim, M., Soete, A., Saeys, Y., De Neve, W.: SpliceRover: interpretable convolutional neural networks for improved splice site prediction. Bioinformatics. **34**(24), 4180–4188 (2018)
6. Yin, Q., Wu, M., Liu, Q., Lv, H., Jiang, R.: DeepHistone: a deep learning approach to predicting histone modifications. BMC Genomics. **20**(2), 11–23 (2019)

7. Zhang, J., Peng, W., Wang, L.: LeNup: learning nucleosome positioning from DNA sequences with improved convolutional neural networks. Bioinformatics. **34**(10), 1705–1712 (2018)
8. Quang, D., Xie, X.: DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. Nucleic Acids Res. **44**(11), e107–e107 (2016)
9. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. Adv. Neural. Inf. Process. Syst. **30**, (2017)
10. Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C.L., Ma, J.: Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proc. Natl. Acad. Sci. **118**(15), e2016239118 (2021)
11. Elnaggar, A., Heinzinger, M., Dallago, C., Rihawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M.: ProtTrans: towards cracking the language of Life's code through self-supervised deep learning and high performance computing. arXiv preprint arXiv:200706225, (2020)
12. Rao, R.M., Liu, J., Verkuil, R., Meier, J., Canny, J., Abbeel, P., Sercu, T., Rives, A.: Msa transformer. In: International Conference on Machine Learning, pp. 8844–8856. PMLR, Cambridge (2021)
13. Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A.: Highly accurate protein structure prediction with AlphaFold. Nature. **596**(7873), 583–589 (2021)
14. Avsec, Ž., Agarwal, V., Visentin, D., Ledsam, J.R., Grabska-Barwinska, A., Taylor, K.R., Assael, Y., Jumper, J., Kohli, P., Kelley, D.R.: Effective gene expression prediction from sequence by integrating long-range interactions. Nat. Methods. **18**(10), 1196–1203 (2021). https://doi.org/10.1038/s41592-021-01252-x
15. Le, N.Q.K., Ho, Q.-T., Nguyen, T.-T.-D., Ou, Y.-Y.: A transformer architecture based on BERT and 2D convolutional neural network to identify DNA enhancers from sequence information. Brief. Bioinform. **22**(5), bbab005 (2021)
16. Le, N.Q.K., Ho, Q.-T.: Deep transformers and convolutional neural network in identifying DNA N6-methyladenine sites in cross-species genomes. Methods. **204**, 199–206 (2022)
17. Ji, Y., Zhou, Z., Liu, H., Davuluri, R.V.: DNABERT: pre-trained bidirectional encoder representations from transformers model for DNA-language in genome. Bioinformatics. **37**(15), 2112–2120 (2021)
18. Mo, S., Fu, X., Hong, C., Chen, Y., Zheng, Y., Tang, X., Lan, Y., Shen, Z., Xing, E.: Multimodal Self-supervised Pre-training for Large-scale Genome Data. In: NeurIPS 2021 AI for Science Workshop, (2021)
19. Yang, M., Huang, H., Huang, L., Zhang, N., Wu, J., Yang, H., Mu, F.: Integrating convolution and self-attention improves language model of human genome for interpreting non-coding regions at base-resolution. bioRxiv, (2021)
20. Herbert, A.: A genetic instruction code based on DNA conformation. Trends Genet. **35**(12), 887–890 (2019)
21. Herbert, A.: ALU non-B-DNA conformations, flipons, binary codes and evolution. R. Soc. Open Sci. **7**(6), 200222 (2020). https://doi.org/10.1098/rsos.200222
22. Herbert, A.: The simple biology of Flipons and condensates enhances the evolution of complexity. Molecules. **26**(16), 4881 (2021). https://doi.org/10.3390/molecules26164881
23. Herbert, A.: Z-DNA and Z-RNA in human disease. Communications biology. **2**(1), 1–10 (2019)
24. Herbert, A.: Contextual cell death in adaptive immunity: selecting a winning response. Front. Immunol. **10**, 2898 (2019). https://doi.org/10.3389/fimmu.2019.02898
25. Herbert, A.: ADAR and immune silencing in cancer. Trends Cancer. **5**(5), 272–282 (2019). https://doi.org/10.1016/j.trecan.2019.03.004
26. Herbert, A.: Mendelian disease caused by variants affecting recognition of Z-DNA and Z-RNA by the Zalpha domain of the double-stranded RNA editing enzyme ADAR. Eur. J. Hum. Genet. **28**(1), 114–117 (2020). https://doi.org/10.1038/s41431-019-0458-6

27. Zhang, T., Yin, C., Boyd, D.F., Quarato, G., Ingram, J.P., Shubina, M., Ragan, K.B., Ishizuka, T., Crawford, J.C., Tummers, B., Rodriguez, D.A., Xue, J., Peri, S., Kaiser, W.J., Lopez, C.B., Xu, Y., Upton, J.W., Thomas, P.G., Green, D.R., Balachandran, S.: Influenza virus Z-RNAs induce ZBP1-mediated necroptosis. Cell. **180**(6), 1115–1129 (2020). https://doi.org/10.1016/j.cell.2020.02.050
28. Zhang, T., Yin, C., Fedorov, A., Qiao, L., Bao, H., Beknazarov, N., Wang, S., Gautam, A., Williams, R.M., Crawford, J.C.: ADAR1 masks the cancer immunotherapeutic promise of ZBP1-driven necroptosis. Nature. **606**, 1–9 (2022)
29. Ho, P.S., Ellison, M.J., Quigley, G.J., Rich, A.: A computer aided thermodynamic approach for predicting the formation of Z-DNA in naturally occurring sequences. EMBO J. **5**(10), 2737–2744 (1986)
30. Schroth, G.P., Chou, P.-J., Ho, P.S.: Mapping Z-DNA in the human genome. Computer-aided mapping reveals a nonrandom distribution of potential Z-DNA-forming sequences in human genes. J. Biol. Chem. **267**(17), 11846–11855 (1992)
31. Beknazarov, N., Jin, S., Poptsova, M.: Deep learning approach for predicting functional Z-DNA regions using omics data. Sci. Rep. **10**(1), 19134 (2020). https://doi.org/10.1038/s41598-020-76203-1
32. Shin, S.-I., Ham, S., Park, J., Seo, S.H., Lim, C.H., Jeon, H., Huh, J., Roh, T.-Y.: Z-DNA-forming sites identified by ChIP-Seq are associated with actively transcribed regions in the human genome. DNA Res. **23**(5), 477–486 (2016)
33. Kouzine, F., Wojtowicz, D., Baranello, L., Yamane, A., Nelson, S., Resch, W., Kieffer-Kwon, K.R., Benham, C.J., Casellas, R., Przytycka, T.M., Levens, D.: Permanganate/S1 nuclease Footprinting reveals non-B DNA structures with regulatory potential across a mammalian genome. Cell Syst. **4**(3), 344–356e347 (2017). https://doi.org/10.1016/j.cels.2017.01.013
34. Ellison, M.J., Fenton, M.J., Ho, P.S., Rich, A.: Long-range interactions of multiple DNA structural transitions within a common topological domain. EMBO J. **6**(5), 1513–1522 (1987). https://doi.org/10.1002/j.1460-2075.1987.tb02394.x
35. Fang, S., Zhang, L., Guo, J., Niu, Y., Wu, Y., Li, H., Zhao, L., Li, X., Teng, X., Sun, X., Sun, L., Zhang, M.Q., Chen, R., Zhao, Y.: NONCODEV5: a comprehensive annotation database for long non-coding RNAs. Nucleic Acids Res. **46**(D1), D308–D314 (2018). https://doi.org/10.1093/nar/gkx1107

Manipulation by Coalitions in Voting with Incomplete Information

Yuliya A. Veselova

1 Introduction

We consider the problem of manipulation in collective decision making. It is well-known that voters can misrepresent their preferences in order to achieve a more preferable result. Of course, it is better when all voters want to declare their sincere preferences, otherwise, a collective decision would be biased and, consequently, would not reflect the preference of a society. Unfortunately, all social choice rules which have at least three possible outcomes are either manipulable or dictatorial. This result is called Gibbard-Satterthwaite theorem [5, 7, 14].

One approach to comparing manipulability of social choice rules is calculating the probability of manipulation. This method has been successfully applied to studying both individual and coalitional manipulation in different probabilistic models many times in the literature, see [1–4, 8, 9, 11, 12, 15]. However, the common assumption in publications of this line of research is that voters know each others' sincere preference, i.e. public information is reliable and complete. This is a rather strong assumption, but helps to simplify the comparative analysis of manipulability

The paper was prepared within the framework of the Basic Research Program at the National Research University Higher School of Economics (HSE) and supported within the framework of a subsidy by the Russian Academic Excellence Project '5–100'. The work was conducted by the International Laboratory of Decision Choice and Analysis (DeCAn Lab) of the National Research University Higher School of Economics.

Y. A. Veselova (✉)
National Research University Higher School of Economics, Moscow, Russian Federation

V. A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russian Federation
e-mail: yul-r@mail.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_23

of social choice rules. Intuitively, incomplete information would make manipulation more difficult and rare.

A more realistic assumption is that voters have some information from opinion polls held before voting. This information could be represented, for example, by preferences of a subset of voters, or a list of candidate scores, or the winner of the election. A mathematical model for manipulation under poll information is presented by Reijngoud and Endriss [13].

In the current research we apply this model to studying coalitional manipulability of social choice rules under different types of poll information. We consider the probability that in a randomly chosen preference profile there exists a coalition which has an incentive to manipulate under a given type of poll information. The formalization of coalitional manipulation in voting includes some assumptions: (1) voters form a coalition if they have the same preference; (2) all members of a coalition manipulate in the same way; (3) a coalition has an incentive to manipulate, if there exists an insincere strategy such that the coalition cannot become worse off and there is a chance of becoming better off with this strategy.

In our study the analysis of manipulation probability has three directions:

1. We analyze the power of a coalition: how coalitional manipulability differs from individual. Could coalitional manipulability be less than individual?
2. We compare manipulability of different social choice rules (we consider six popular rules which have polynomial complexity of calculating a winner: plurality rule, Borda rule, veto rule, runoff procedure, STV rule, and Copeland rule).
3. We study the role of information available to voters. How do different types of poll information affect coalitional manipulability?

We answer these questions via both theoretical investigation and computational experiments. We prove that for scoring rules (plurality, Borda, and veto rule in our analysis) the probability of coalitional manipulation is equal to the probability of individual manipulation if voters have information about the election winner after tie-breaking. Computational experiments are conducted in MatLab for all six rules and five poll information types for 3 alternatives and the number of voters from 3 to 15. It is shown that the probability of coalitional manipulation is almost always higher than individual manipulation and in many cases is very close to 1. The exceptions are the Borda rule and veto rule: the probability of coalitional manipulation is less than the probability of individual manipulation in some cases. This observation shows that manipulating with the same strategy is not optimal for coalition members in these cases. The veto rule even becomes almost strategy-proof in incomplete information if there are more than 10 voters.

2 The Model

2.1 Definitions and Notations

There is a finite set of *voters* $N = \{1, \ldots, n\}$ and a finite set of m *alternatives* X. Each voter i has a strict *preference* on X, a linear order P_i. If voter i prefers an alternative a to an alternative b, we write aP_ib. The set of all linear orders is denoted by $L(X)$. An *upper contour set* of an alternative a in a preference order P_i is $P_ia = \{b \in X : bP_ia\}$. Similarly, a *lower contour set* of a in P_i is $aP_i = \{b \in X : aP_ib\}$.

An ordered set of individual preferences, $\mathbf{P} = (P_1, \ldots, P_n) \in L(X)^N$, is called a *preference profile*. A contraction of a preference profile onto the set $A \subseteq X$ is $\mathbf{P}/A = (P_1/A, \ldots, P_n/A)$, where $P_i/A = P_i \cap (A \times A)$. A *coalition* is a subset of voters, $K \subseteq N$, $\mathbf{P}_K$ is a preference profile of coalition members, $\mathbf{P}_{-K}$—preference profile of all other voters, $N \setminus K$. $\mathbf{P} = (\mathbf{P}_K, \mathbf{P}_{-K})$.

A vector of positions for alternative a is $v(a, \mathbf{P}) = (v_1(a, \mathbf{P}), \ldots, v_m(a, \mathbf{P}))$, where $v_j(a, \mathbf{P})$ denotes the number of voters having a on the j-th position in a preference order.

An $m \times m$ matrix of a *weighted majority graph* for a profile $\mathbf{P}$ is denoted by *WMG*($\mathbf{P}$) and consists of elements

$$WMG(\mathbf{P})_{kl} = |\{i \in N : a_k P_i a_l\}|. \tag{1}$$

By μ we denote *majority relation*: $a_k \mu a_l$ if $WMG(\mathbf{P})_{kl} > WMG(\mathbf{P})_{lk}$.

A matrix of a majority graph is *MG*($\mathbf{P}$), where

$$MG(\mathbf{P})_{kl} = \begin{cases} 1, & \text{if } a_k \mu a_l, \\ -1, & \text{if } a_l \mu a_k, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

A *social choice correspondence* (SCC) is a mapping $C\colon L(X)^N \to 2^X \setminus \{\emptyset\}$. A *social choice rule* or simply *rule* is a mapping $F\colon L(X) \to X$. A rule can be obtained from SCC by using a tie-breaking rule $T : 2^X \setminus \{\emptyset\} \to X$. We consider an alphabetic tie-breaking rule: assume some linear order on X to be predefined, $aP_TbP_Tc \ldots$, and when alternatives are tied, we choose the one which dominates all others by P_T (has a higher priority). So a rule F is derived from SCC C, if $T(C(\mathbf{P})) = F(\mathbf{P})$.

2.2 Poll Information Functions

It is assumed that an opinion poll is held before voting and it reveals voters' sincere preferences, $\mathbf{P}$. However, for some reasons not all information becomes available to voters. Instead of $\mathbf{P}$, voters get to know just $\pi(\mathbf{P})$, function π is called a poll information function (PIF). We consider the following types of PIF.

1. Profile: $\pi(\mathbf{P}) = \mathbf{P}$.
2. Score: $\pi(\mathbf{P}) = \mathbf{S}(\mathbf{P}) = (S(a_1, \mathbf{P}), \ldots, S(a_m, \mathbf{P}))$ assigns to each alternative its score (to be explained further) according to a given SWF F. It may be defined specifically for some rules, e.g. sequential procedures.
3. Rank: $\pi(\mathbf{P}) = R$ returns a social ordering.
4. Winner: $\pi(\mathbf{P}) = C(\mathbf{P})$.
5. Unique winner (1Winner): $\pi(\mathbf{P}) = F(\mathbf{P})$

2.3 Individual Manipulation

Thus, a voter i has information $\pi(\mathbf{P})$ about a preference profile $\mathbf{P}$ and knows her own preference order. A set of preference profiles of $N \setminus \{i\}$ consistent with her knowledge is called *information set* and defined as follows

$$W_i^{\pi(\mathbf{P})} = \{\mathbf{P}'_{-i} \in L(X)^{N\setminus\{i\}} : \pi(P_i, \mathbf{P}'_{-i}) = \pi(\mathbf{P})\}. \tag{3}$$

Given two PIFs π and π', if $\forall \mathbf{P} \in L(X)^N$ $\forall i \in N$ $W_i^{\pi(\mathbf{P})} \subseteq W_i^{\pi'(\mathbf{P})}$, then π is *at least as informative as* π'. Of course, the most informative is Profile-PIF.

Then, when is a voter willing to manipulate, i.e. misrepresent her preference in order to achieve a more preferable result? It is assumed that if there is at least one possible situation in which manipulation makes her better off and nothing changes in all other possible situations, then a voter has an incentive to manipulate under PIF π [13].

Definition 1 Given a rule F and a preference profile $\mathbf{P}$, we say, that voter i *has an incentive to* π*-manipulate under* F, if there exists $\tilde{P}_i \in L(X)$ s.t.

(i) there is no $\mathbf{P}'_{-i} \in W_i^{\pi(\mathbf{P})}$, s.t. $F(\mathbf{P})\ P_i\ F(\tilde{P}_i, \mathbf{P}'_{-i})$;
(ii) there exists $\mathbf{P}'_{-i} \in W_i^{\pi(\mathbf{P})}$, s.t. $F(\tilde{P}_i, \mathbf{P}'_{-i})\ P_i\ F(\mathbf{P})$.

Definition 2 A rule F is called *susceptible to individual* π*-manipulation* if there exists a profile $\mathbf{P} \in L(X)^N$ and a voter $i \in N$ who has an incentive to π-manipulate in $\mathbf{P}$ under F.

Let $I_{ind}(m, n, \pi, F)$ be the probability that in a preference profile, randomly chosen from $L(X)^N$ there exists at least one voter who has an incentive to π-manipulate under F.

2.4 Coalitional Manipulation

We assume that voters form a manipulating coalition if they have identical preferences. A coalition of voter i is denoted by K and it consists of all voters having the same preference as voter i. However, π is the only information available to voters, each voter does not know exactly who is in her coalition. In each preference profile $\mathbf{P}'$ of voter i's information set there is a set K' of her coalition members (allies).

Then, a voter is willing to manipulate within a coalition when there is a strategy $\tilde{P}$ (insincere preference), such that the voting result is not less preferable in all profiles and is more preferable in at least one profile of her information set assuming that all members of her coalition vote $\tilde{P}$ (denoted by $\tilde{\mathbf{P}}'_{K'}$) in each possible preference profile $\mathbf{P}'$. More formally,

Definition 3 Given a rule F and a preference profile $\mathbf{P}$, we say, that voter i *has an incentive to π-manipulate within a coalition*,[1] if there exists $\tilde{P} \in L(X)$ s.t.

(i) there is no $\mathbf{P}' \in W_i^{\pi(\mathbf{P})}$ s.t. $F(\mathbf{P}')\ P_i\ F(\tilde{\mathbf{P}}'_{K'}, \mathbf{P}'_{-K'})$;
(ii) there exists $\mathbf{P}' \in W_i^{\pi(\mathbf{P})}$ s.t. $F(\tilde{\mathbf{P}}'_{K'}, \mathbf{P}'_{-K'})\ P_i\ F(\mathbf{P}')$, where $\tilde{\mathbf{P}}'_{K'}$ is a preference profile of a coalition K' in $\mathbf{P}'$, s.t. all voters of K' vote $\tilde{P}$.

If voter i has an incentive to manipulate within a coalition, then we similarly say that the whole coalition has an incentive to manipulate.

Definition 4 A rule F is called *coalitionally manipulable under π* if there exists a profile $\mathbf{P} \in L(X)^N$ and a voter $i \in N$ who has an incentive to π-manipulate within a coalition in $\mathbf{P}$.

Denote by $I_{coal}(m, n, \pi, F)$ the probability that in a preference profile, randomly chosen from $L(X)^N$ there exists at least one voter who has an incentive to π-manipulate within a coalition under F.

2.5 Social Choice Correspondences

Here we give the definition of social choice correspondences that we focus on in this chapter. For each of them we need to specify how scores and rankings are computed to use them in *Score*-PIF and *Rank*-PIF.

- *Scoring rules*. A scoring rule is defined by a scoring vector $s = (s_1, \ldots, s_m)$, where s_j denotes the score assigned to an alternative for the j-th position in individual preferences. The total score of each alternative $a_j \in X$ is calculated as $S(a_j, \mathbf{P}) = \sum_{h=1}^{m} s_h \cdot v_h(a_j, \mathbf{P})$. Then, $R = P \cup I$ is defined as follows: $\forall a_k, a_l \in X$ i) $a_k P a_l \Leftrightarrow S(a_k, \mathbf{P}) > S(a_l, \mathbf{P})$; ii) $a_k I a_l \Leftrightarrow S(a_k, \mathbf{P}) = S(a_l, \mathbf{P})$.
  - *Plurality*: $s_{Pl} = (1, 0, \ldots, 0)$.
  - *Veto* (Antiplurality): $s_V = (1, \ldots, 1, 0)$.
  - *Borda*: $s_B = (m-1, m-2, \ldots, 1, 0)$.

[1] The definition of coalitional manipulation differs from a standard one due to simplification we made: voters have identical preferences and manipulate in the same way. In a general framework, voters may have different preferences and manipulate differently.

- *Run-off procedure*. It has two stages:
  (1) The plurality score is calculated for each alternative. A first-stage vector of scores

  $$S^1(\mathbf{P}) = (S^1(a_1, \mathbf{P}), \ldots, S^1(a_m, \mathbf{P})),$$

  where $S^1(a_j, \mathbf{P}) = \langle s_{Pl}, v(a_j, \mathbf{P})\rangle$. If $\exists\ a_k \in X$ s.t. $S^1(a_k) > n/2$, then social ordering is $a_k P a_j, a_j I a_h\ \forall a_j, a_h \in X \backslash \{a_k\}$ and procedure terminates. Otherwise, procedure moves on to the stage two.
  (2) Two alternatives with maximal number of scores are chosen:
  $a_k = argmax_{a_j \in X}(S^1(a_j, \mathbf{P})), a_l = argmax_{a_j \in X \backslash \{a_k\}}(S^1(a_j, \mathbf{P}))$.

  If there are ties, they are broken according to the alphabetical tie-breaking rule T. Then a second-stage vector of scores is calculated: $S^2(\mathbf{P}) = (S^2(a_k, \mathbf{P}), S^2(a_l, \mathbf{P}))$, where

  $$S^2(a_k, \mathbf{P}) = \langle s_{Pl}, v(a_k, \mathbf{P}/\{a_k, a_l\})\rangle,$$

  $$S^2(a_l, \mathbf{P}) = \langle s_{Pl}, v(a_l, \mathbf{P}/\{a_k, a_l\})\rangle.$$

  The alternative with the higher score is considered better, $a_k P a_l$ if $S^2(a_k, \mathbf{P}) > S^2(a_l, \mathbf{P})$ and $a_l P a_k$ if $S^2(a_l, \mathbf{P}) > S^2(a_k, \mathbf{P})$. Both of them are better than all other alternatives, $\forall a_j \in X \setminus \{a_k, a_l\}\ a_l P a_j, a_k P a_j$. All other alternatives are considered as indifferent $\forall a_j, a_h \in X \setminus \{a_k, a_l\}\ a_j I a_h$. The output of Score-PIF is $S(\mathbf{P}) = (S^1(\mathbf{P}), S^2(\mathbf{P}))$.
- *Single Transferable vote (STV)*. This is a multi-stage procedure, which we define in an iterative form.
  (0) $t := 1, X^t := X, \mathbf{P}^t := \mathbf{P}$.
  (1) $\forall a_j \in X^t\ S^t(a_j, \mathbf{P}) := \langle s_{Pl}, v(a_j, \mathbf{P}^t)\rangle$.
  (2) If $\exists a_j \in X^t$ s.t. $S^t(a_j, \mathbf{P}) > n/2$, then $\forall a_h, a_l \in X^t \setminus \{a_j\}\ a_j P a_h,\ a_h I a_l$, the procedure terminates. Else $a_k := argmin_{a_j \in X^t}(S^t(a_j, \mathbf{P})), \forall a_j \in X^t \setminus \{a_k\}$ $a_j P a_k$.
  (3) $t := t + 1, X^t := X^t \setminus \{a_k\}, \mathbf{P}^t := \mathbf{P}/X^t$. Go to step 1.
  The output of Score-PIF is $S(\mathbf{P}) = (S^1(\mathbf{P}), \ldots, S^{t*}(\mathbf{P}))$, where $t*$ is the number of cycles done by procedure.
- *Copeland*. A majority graph is computed. Then scores of alternatives are computed as follows

  $$S(a_k, \mathbf{P}) = \sum_{l=1}^{m} MG(\mathbf{P})_{kl}.$$

  Ranking $R = P \cup I$ is defined as usual: $\forall a_k, a_l \in X$

  (i) $a_k P a_l \Leftrightarrow S(a_k, \mathbf{P}) > S(a_l, \mathbf{P})$;
  (ii) $a_k I a_l \Leftrightarrow S(a_k, \mathbf{P}) = S(a_l, \mathbf{P})$.

3 Theoretical Results

In this section we prove some statements about the probability of individual and coalitional manipulation under incomplete information for any number of voters and alternatives. Before proving theorems, let us introduce some notations and consider an auxiliary statement, Lemma 1. Let d denote the number of preference profiles for n voters and m alternatives, i.e. $d = (m!)^n$, and $d_F(a)$ is the number of profiles in $L(X)^N$ where alternative a wins according to a rule F. Further, let $z(a)$ be the number of preference profiles in $L(X)^N$ where no voter has alternative a on the last position in a preference order and let $z_F(a)$ denote the number of preference profiles where alternative a wins according to a rule F and does not take the last position in any preference order.

Lemma 1 *For any alternative a and any rule F, s.t. for every $x \in X$ $\lim_{n\to\infty} d_F(x)/d = 1/m$, $\lim_{n\to\infty} z_F(a)/d_F(a) = 0$.*

Proof The total number of preference profiles for n voters and m alternatives is $d = (m!)^n$. The number of preference profiles where no voter has alternative a on the last place in a preference order does not depend on a rule or alternative and equals $z(a) = (m! - (m-1)!)^n$. Thus, the share of preference profiles where no voter has alternative a on the last place in a preference order:

$$\frac{z(a)}{z} = \frac{(m! - (m-1)!)^n}{(m!)^n} = \left(\frac{m! - (m-1)!}{m!}\right)^n = \left(1 - \frac{1}{m}\right)^n. \tag{4}$$

$$\frac{z(a)}{d_F(a)} = \frac{z(a)}{d} \cdot \frac{d}{d_F(a)}. \tag{5}$$

Since the rule is such that for any $x \in X$ $\lim_{n\to\infty} d_F(x)/d = 1/m$, $\lim_{n\to\infty} d/d_F(x) = m$. Using this and Eq. (5), we have

$$\lim_{n\to\infty} \frac{z(a)}{d_F(a)} = 0. \tag{6}$$

As $z_F(a)/d_F(a) < z(a)/d_F(a)$, $z_F(a)/d_F(a)$ also tends to 0 as n goes to infinity.

Now let us introduce some simplifying notations. Let $S(a)$ be initial scores of a, i.e $S(a, \mathbf{P})$, and $\tilde{S}(a)$ be scores of a after manipulation of an individual or a group (depending on the context), i.e. $S(a, (\tilde{P}_i, \mathbf{P}_{-i}))$ or $S(a, (\tilde{\mathbf{P}}_K, \mathbf{P}_{-K}))$. The first result concerns individual manipulation under *Winner*-PIF for plurality rule.

Theorem 1 $\lim_{n\to\infty} I_{ind}(m, n, \mathit{Winner}, \mathit{Plurality}) = 1 - 1/m$ *with alphabetic tie-breaking.*

Proof Let $X = \{a_1, a_2, \ldots, a_m\}$ and $a_1 P_T a_2 P_T \ldots P_T a_m$. The PIF is $\pi(\mathbf{P}) = F(\mathbf{P})$. The result $F(\mathbf{P})$ could consist of one alternative, i.e. $F(\mathbf{P}) = \{a_k\}$, $k \in \{1, 2, \ldots, m\}$, or there can be a draw.

First, consider the case $F(\mathbf{P}) = \{a_k\}$, $k \in \{2, 3, \ldots, m\}$. It means that $S(a_k) \geq S(a_j) + 1\ \forall j \neq k$. Suppose there is a voter i who thinks that a_k is the worst alternative. Since it is known that $S(a_k) \geq S(a_j) + 1\ \forall j \neq k$ there is an equal chance (from voters' point of view) for any other alternative to win if it gets plus one score. Thus, the best strategy for voter i is to vote for alternative a_h, which is the best for voter i among alternatives that tie-break against a_k. So, if $F(\mathbf{P}) = \{a_k\}$, $k \in \{2, 3, \ldots, m\}$ and there is at least one voter that has a_k one the lowest position in a preference order, then a preference profile $\mathbf{P}$ is individually manipulable under $\pi(\mathbf{P}) = F(\mathbf{P})$.

If $F(\mathbf{P}) = \{a_1\}$, then $S(a_1) \geq S(a_j) + 1\ \forall j \neq 1$. However, even if any voter manipulates in favor of some other alternative a_h, it could not win, since $\tilde{S}(a_1) \geq \tilde{S}(a_h)$ and $a_1 P_T a_h$. Thus, in case of a tie, $\tilde{S}(a_1) = \tilde{S}(a_h)$, a_1 wins. So, all profiles with the unique winner a_1 are not manipulable by individuals under $\pi(\mathbf{P}) = F(\mathbf{P})$.

The proportion of profiles with a single-valued outcome for plurality rule tends to 1 as n goes to infinity.[2] Since the rule is neutral (it means, it treats all the alternatives equally), the chance of winning for each of them tends to $1/m$. As we derived earlier, when the winner is $F(\mathbf{P}) = \{a_k\}$, $k \in \{2, 3, \ldots, m\}$, then manipulation is possible in profiles with at least one voter having a_k on the last place, if $F(\mathbf{P}) = \{a_1\}$, then individual manipulation is impossible. By Lemma 1, for all $a \in X$, $\lim_{n\to\infty}(d_F(a) - z_F(a))/d_F(a) = 1$. The number of manipulable profiles is not less then $d_F(a_2) - z_F(a_2) + \ldots + d_F(a_m) - z_F(a_m)$ and the number of non-manipulable is not less then $d_F(a_1)$. Thus, $\lim_{n\to\infty}(d_F(a_2) - z_F(a_2) + \ldots + d_F(a_m) - z_F(a_m))/d = 1 - 1/m$.

Thus, with infinite n only $1/m$ of profiles will be non-manipulable. In this connection, let us recall the result from [16] which says that the same probability, but with *1Winner*-PIF equals one. If a voter manipulates within a coalition under *Winner*-PIF, then again asymptotic probability equals 1.

Theorem 2 $\lim_{n\to\infty} I_{coal}(m, n, Winner, Plurality) = 1$ *with alphabetic tie-breaking.*

Proof Let $X = \{a_1, a_2, \ldots, a_m\}$ and $a_1 P_T a_2 P_T \ldots P_T a_m$. The PIF is $\pi(\mathbf{P}) = F(\mathbf{P})$.

Let us prove that all preference profiles with a single winner and a voter having the winning alternative on the last position in a preference order are coalitionally manipulable. If $F(\mathbf{P}) = \{a_k\}$, $k \in \{1, 2, \ldots, m\}$, then $S(a_k) \geq S(a_j) + 1\ \forall j \neq k$. Consider a voter i who thinks that a_k is the worst alternative, a_h is her second-best, and a_l is her best alternative.

Suppose, in voter's information set there is at least one preference profile $\mathbf{P}'$, such that the number of voters in her coalition is not less than 2, $|K'| \geq 2$, and $S(a_k) - S(a_h) < |K'|$. In this case, manipulation of K' by voting for a_h leads to

[2] We refer to [6]. It is shown that the probability of a tie between any pair of alternatives with plurality rule tends to 0 as the number of voters goes to infinity (by Central Limit Theorem).

$\tilde{S}(a_h) > \tilde{S}(a_k)$ and the winning of a_h. If $n \geq 7$, then such profile exists in voter i's information set $W_i^{\pi(\mathbf{P})}$ (since $S(a_l) \geq |K'| \geq 2$, $S(a_k) \geq S(a_l) + 1 \geq 3$, and $S(a_k) = S(a_h) + 1$). In other profiles of $W_i^{\pi(\mathbf{P})}$, s.t. $S(a_k) - S(a_h) \leq |K'|$, this manipulation will not change anything. Thus, in profile $\mathbf{P}$ voter i has an incentive to manipulate within a coalition.

As a consequence of Lemma 1, the share of profiles with at least one voter having the winning alternative on the last place in a preference order tends to 1. Furthermore, the share of profiles that result in a tie tends to zero. Thus, the probability of coalitional manipulation under *Winner*-PIF tends to 1 as n goes to infinity.

Theorem 3 $\lim_{n\to\infty} I_{coal}(m, n, Winner, Borda) = 1$ *with alphabetic tie-breaking.*

Proof Again, we prove that if the winner is unique, $F(\mathbf{P}) = \{a_k\}$, then a voter having a_k on the last place in a preference order has an incentive to *Winner*-manipulate within a coalition in Borda rule. Let us fix the tie-breaking order $a_1 P_T a_2 P_T \ldots P_T a_m$ and assume that voter i's preference is $a_h P_i a_l P_i \ldots P_i a_k$. If $F(\mathbf{P}) = \{a_k\}$, then for all $j \in \{1, 2, \ldots, m\}$, $j \neq k$, $S(a_k) \geq S(a_j) + 1$.

Consider those profiles $\mathbf{P}'$ of $W_i^{\pi(\mathbf{P})}$, s.t. $|K'| > S(a_k) - S(a_l)$. Suppose voter i and her coalition members change their preferences to $a_l \tilde{P} a_h \tilde{P} \ldots \tilde{P} a_k$ (switch the best and the second-best alternative). Then the new scores will be $\tilde{S}(a_h) = S(a_h) - |K'|$, $\tilde{S}(a_l) = S(a_l) + |K'|$, $\tilde{S}(a_j) = S(a_j)$ for all $a_j \in X \setminus \{a_h, a_l\}$, and $\tilde{S}(a_l) > \tilde{S}(a_k)$, so, a_l will be the winner. In profiles $\mathbf{P}'$ of $W_i^{\pi(\mathbf{P})}$, s.t. $|K'| < S(a_k) - S(a_l)$ noting will change, and if $|K'| = S(a_k) - S(a_l)$, it depends on tie-breaking between a_k and a_l. Thus, voters with a_k on the last place have an incentive to manipulate within a coalition in $\mathbf{P}$.

Borda rule also satisfies the requirement of Lemma 1, i.e. for any $x \in X$ we have $\lim_{n\to\infty} d_F(x)/d = 1/m$ by neutrality and zero ties probability [10]. Thus, the share of profiles with at least one voter having the winning alternative on the last place tends to 1 as n goes to infinity. Since all these profiles are coalitionally manipulable under *Winner*-PIF, $\lim_{n\to\infty} I_{coal}(m, n, Winner, Borda) = 1$.

The next theorem shows that the probability of manipulation for scoring rules is the same when we consider individual or coalitional manipulation under *1Winner*-PIF. Let us introduce some more notations for the proof.

For a scoring vector $s = (s_1, s_2, \ldots, s_m)$, a *jump* is a non-zero difference between two adjacent scoring values. If s has r jumps, then this means that there are distinct $k_1, \ldots, k_r \in \{1, \ldots, m-1\}$ such that $s_{k_1} > s_{k_1+1}, \ldots, s_{k_r} > s_{k_r+1}$, while all other differences are zero. We use the notation $\Delta_j = s_{k_j} - s_{k_j+1}$ for $j = 1, \ldots, r$ to denote the non-zero differences between scoring values.

Theorem 4 *For any number of voters n and any number of alternatives m $I_{ind}(m, n, 1Winner, F) = I_{coal}(m, n, 1Winner, F)$ for scoring rules.*

Proof Let $X = \{a_1, \ldots, a_m\}$. Consider a scoring rule with a scoring vector $s = (s_1, s_2, \ldots, s_m)$, the first jump in s goes after k_1. Let us prove that voter i with a preference $a_1 P_i a_2 P_i \dots P_i a_m$ has no incentive to manipulate under *1Winner*-PIF, if $F(\mathbf{P}) \in \{a_1, a_2, \dots, a_{k_1+1}\}$. If $F(\mathbf{P}) = a_1$, then obviously there is no need for i to misrepresent her preference. Suppose that $F(\mathbf{P}) = b$, $b \in \{a_2, a_3, \dots, a_{k_1+1}\}$ and i manipulates in favor of some a, s.t. aPb. If i puts alternative a higher (if a is not a_1), then nothing changes for a since $s_1 = \dots = s_{k_1}$. Thus, i could only put b lower in $\tilde{P}_i$, but then some alternative $c \in \{a_{k_1+2}, \dots, a_m\}$ goes higher in a preference order. If there are no jumps in s after $k_1 + 1$, then nothing changes for b and c. If there are other jumps after $k_1 + 1$, then c gets plus A scores. Since the only information is $F(\mathbf{P}) = b$, i.e. $S(b, \mathbf{P}) \geq S(x, \mathbf{P})$ $\forall x \in X$, there exists $\mathbf{P}' \in W_i^{\pi(\mathbf{P})}$, s.t. $S(c, (\tilde{P}_i, \mathbf{P}'_{-i})) = S(c, \mathbf{P}') + A > S(x, (\tilde{P}_i, \mathbf{P}'_{-i}))$ $\forall x \in X \setminus \{c\}$. The same is true for the concept of coalitional manipulation. If for some $\mathbf{P}' \in W_i^{\pi(\mathbf{P})}$ holds $S(c, (\tilde{P}_i, \mathbf{P}'_{-i})) = S(c, \mathbf{P}') + A > S(x, (\tilde{P}_i, \mathbf{P}'_{-i}))$ $\forall x \in X \setminus \{c\}$, then $S(c, (\tilde{\mathbf{P}}_K, \mathbf{P}'_{-K})) = S(c, \mathbf{P}') + |K|A > S(x, (\tilde{\mathbf{P}}_K, \mathbf{P}'_{-K}))$ $\forall x \in X \setminus \{c\}$. Therefore, i does not have an incentive to manipulate under *1Winner*-PIF when $F(\mathbf{P}) \in \{a_1, a_2, \dots, a_{k_1+1}\}$ either individually or within a coalition.

Now suppose that $F(\mathbf{P}) = c$ and $c \in \{a_{k_1+2}, \dots, a_m\}$. Voter i cannot give alternatives from $\{a_1, a_2, \dots, a_{k_1}\}$ more scores, but can increase the score of a_{k_1+1} by Δ_1. So, let $\tilde{P}_i$ equal P_i but a_{k_1+1} is switched with $a \in \{a_1, a_2, \dots, a_{k_1}\}$. Thus, $S(a_{k_1+1}, (\tilde{P}_i, \mathbf{P}'_{-i})) = S(a_{k_1+1}, \mathbf{P}') + \Delta_1$ and $S(a, (\tilde{P}_i, \mathbf{P}'_{-i})) = S(a, \mathbf{P}') - \Delta_1$ and $S(x, (\tilde{P}_i, \mathbf{P}'_{-i})) = S(x, \mathbf{P}')$ for all $x \in X \setminus \{a, a_{k_1+1}\}$. So, either $S(a_{k_1+1}, (\tilde{P}_i, \mathbf{P}'_{-i})) > S(x, (\tilde{P}_i, \mathbf{P}'_{-i}))$ for all $x \in X \setminus \{a_{k_1+1}\}$ and a_{k_1+1} wins or $S(c, (\tilde{P}_i, \mathbf{P}'_{-i})) > S(a_{k_1+1}, (\tilde{P}_i, \mathbf{P}'_{-i}))$ and c wins. In case of a tie the result is again either a_{k_1+1} or c depending on a tie-breaking order. The same holds for coalitional manipulation, but $S(a_{k_1+1}, (\tilde{P}_K, \mathbf{P}'_{-K})) = S(a_{k_1+1}, \mathbf{P}') + |K|\Delta_1$ and $S(a, (\tilde{P}_K, \mathbf{P}'_{-K})) = S(a, \mathbf{P}') - |K|\Delta_1$. Thus, for all $\mathbf{P}' \in W_i^{\pi(\mathbf{P})}$ the result is not worse then $F(\mathbf{P})$ after manipulation of i or K and better for some $\mathbf{P}' \in W_i^{\pi(\mathbf{P})}$. Therefore, if $F(\mathbf{P}) \in \{a_{k_1+2}, \dots, a_m\}$, voters with a preference $a_1 P_i a_2 P_i \dots P_i a_m$ have an incentive to manipulate both individually and within a coalition under.

Thus, for any voter having an incentive to manipulate individually there is also an incentive to manipulate within a coalition under *1Winner*-PIF. At the same time, if a voter does not have an incentive to manipulate individually under *1Winner*-PIF, then there is no incentive to manipulate coalitionally. It means that the set of individually manipulable profiles is the same as the set of profiles manipulable within a coalition. So, for scoring rules $I_{ind}(m, n, \mathit{1Winner}, F) = I_{coal}(m, n, \mathit{1Winner}, F)$.

As shown by Veselova [16], the probability of manipulation for plurality rule under *1Winner*-PIF tends to 1. By Theorem 4, $I_{coal}(m, n, \mathit{1Winner}, \mathit{Plurality})$ also tends to 1. On the other hand, it was proved that $I_{ind}(m, n, \mathit{1Winner}, \mathit{Veto}) = 0$ by Reijngoud and Endriss [13], and by Theorem 4 $I_{coal}(m, n, \mathit{1Winner}, \mathit{Veto}) = 0$.

4 Computational Experiments

This section shows computed values of $I_{coal}(m, n, \pi, F)$ for all PIFs from Sect. 2.2 and all rules listed in Sect. 2.5. Moreover, we compare these values with $I_{ind}(m, n, \pi, F)$ computed in the work of Veselova [16]. We consider $m = 3$ and n from 3 to 15. All computations were done in MatLab. Results are represented in Figs. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, and 12.

We make the following observations.

- Except for veto rule and only one case with Borda rule, coalitional manipulability is not less than individual. Particularly, we observe a clear going-to-1 tendency not only for *1Winner*-PIF, but also for *Winner*-PIF (all rules except for veto) and *Rank*-PIF in some cases (plurality, Borda, runoff, STV).
- In all cases with non-zero individual manipulability of veto rule the values of coalitional manipulability are strictly lower than individual.
- Individual and coalitional manipulability under *1Winner*-PIF coincide not only for scoring rules, but for all rules under consideration. Moreover, for runoff and Copeland rule these values coincide under *Winner*-PIF.
- The observation 'less information—equal or higher manipulability' is still true in the coalition case for plurality rule, runoff, and STV. With little exceptions it holds for Copeland and with only one exception case for Borda rule. For veto rule the opposite is true: 'less information—equal or less manipulability'.

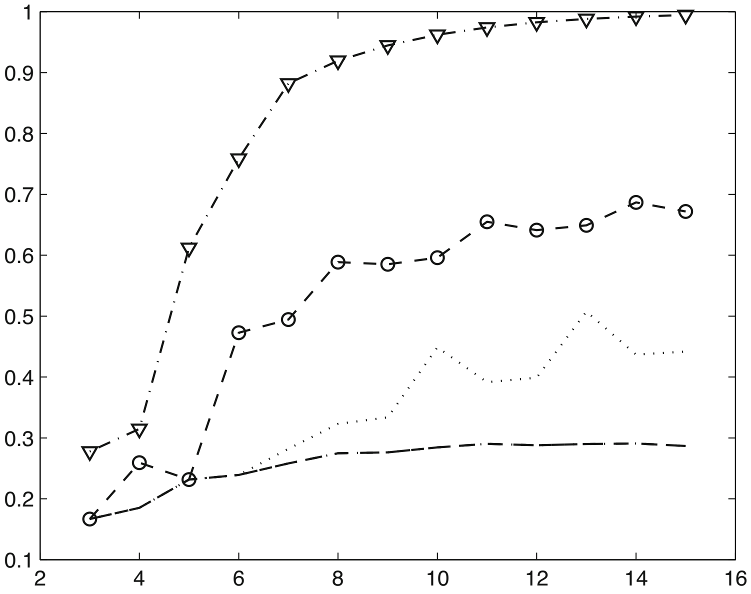


Fig. 1 Plurality rule, individual

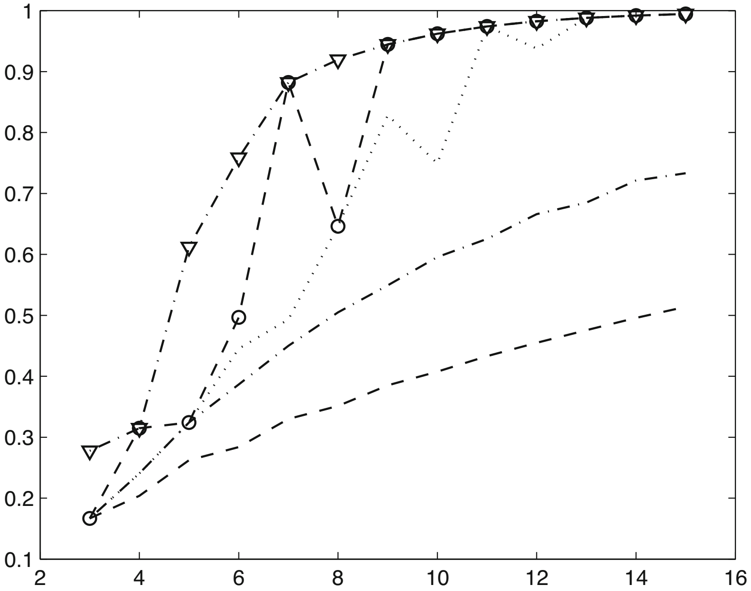


Fig. 2 Plurality rule, group

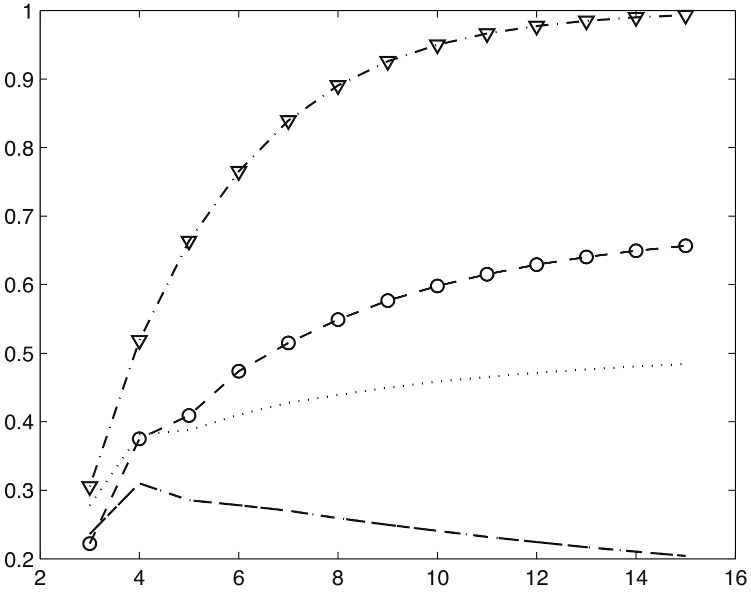


Fig. 3 Borda rule, individual

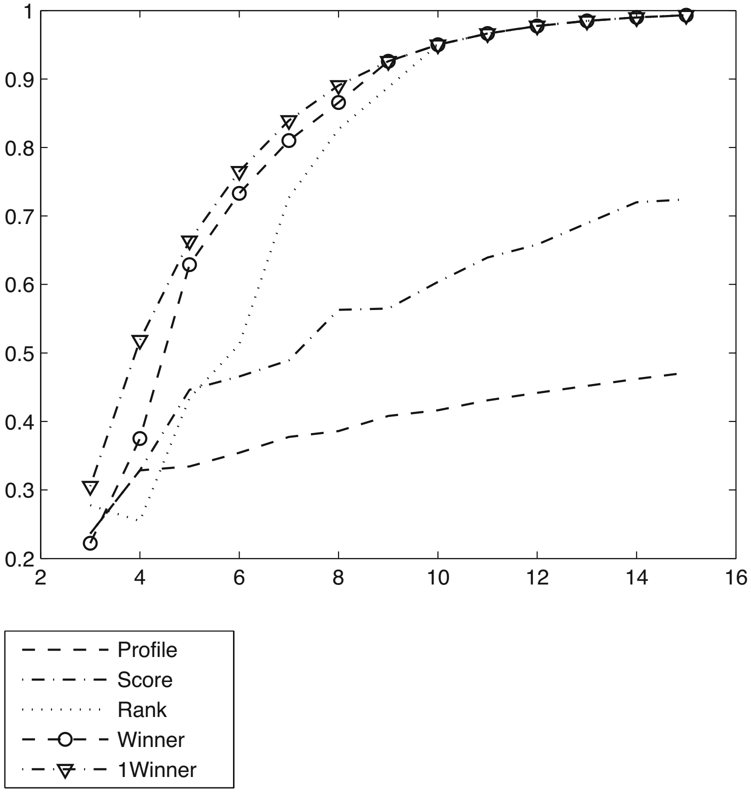


Fig. 4 Borda rule, group

- A rule is called strongly computable from π-images if a voter knowing $\pi(\mathbf{P})$ can compute the result of the rule for any way of her misrepresenting preference. One of results of the work by Veselova [16] is that individual manipulability under π does not change compared to a complete information case if the rule is strongly computable from π-images. The same does not hold for coalitional manipulation.
- With n growing coalitional manipulation of veto rule quickly becomes zero for any kind of incomplete information. It could be explained by the following argument. Manipulation in veto rule means switching the least preferred alternative and some other. The larger is the number of voters, the larger is the cardinality of the maximal possible coalition of voter i. The larger is the coalition, the bigger is the chance of making the least preferred alternative the winner by adding scores to it.
- Periodicity of manipulability index for Copeland rule is rather strong for *Winner*-PIF and *1Winner*-PIF, its amplitude is around 0.4–0.6. So, slight changes in the number of voters may lead to a considerable reduction in manipulation possibilities.

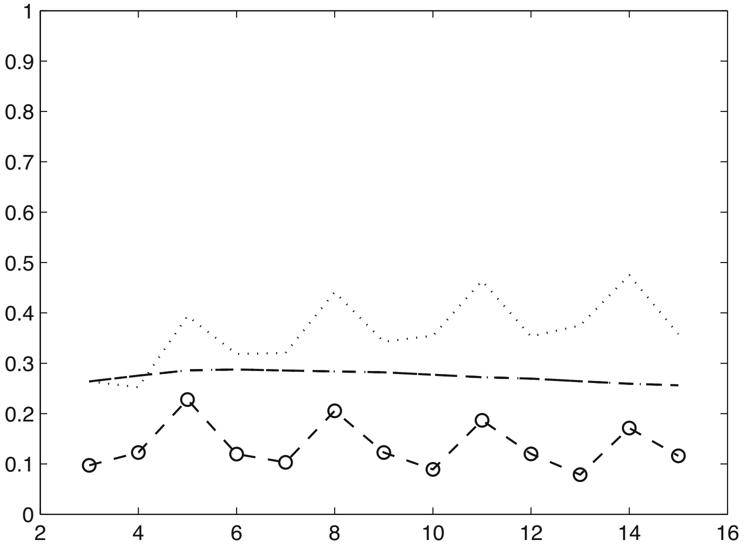


Fig. 5 Veto rule, individual

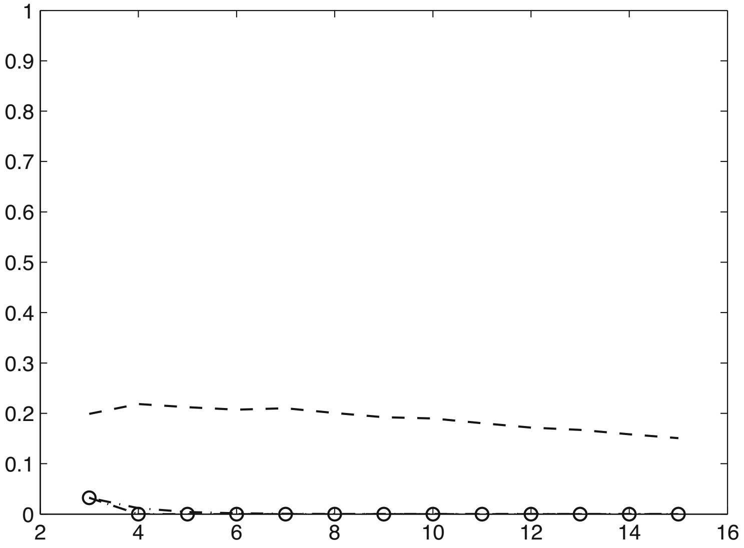


Fig. 6 Veto rule, group

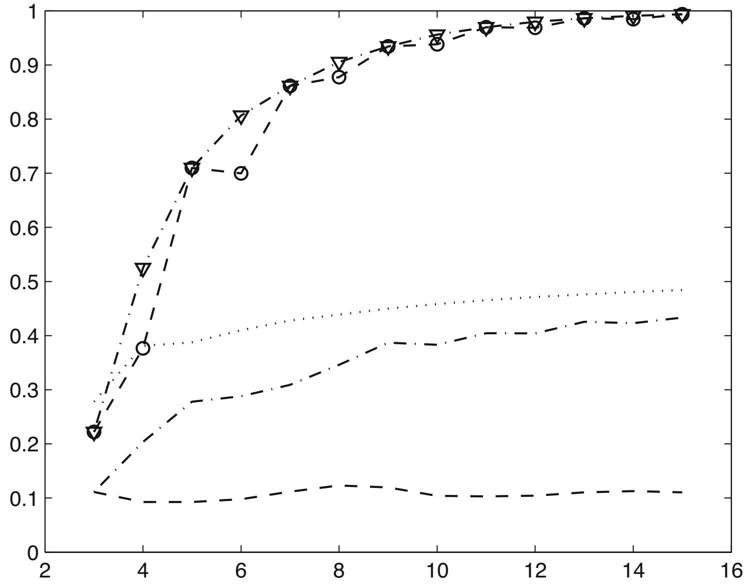


Fig. 7 Runoff, individual

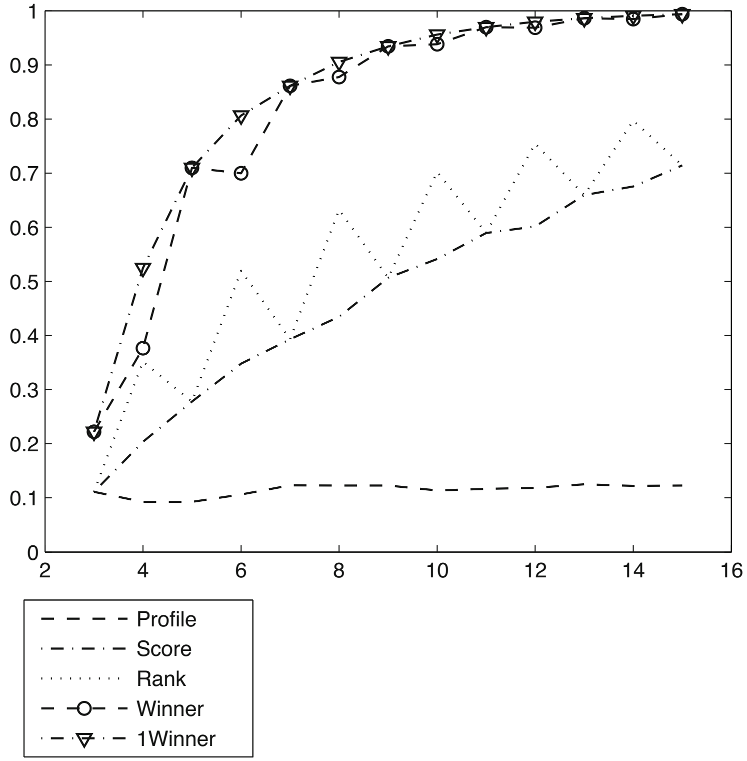


Fig. 8 Runoff, group

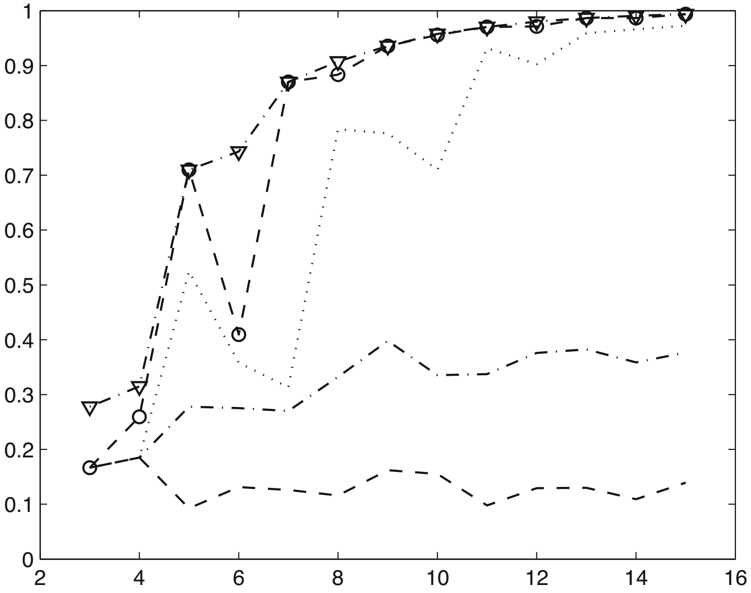


Fig. 9 STV, individual

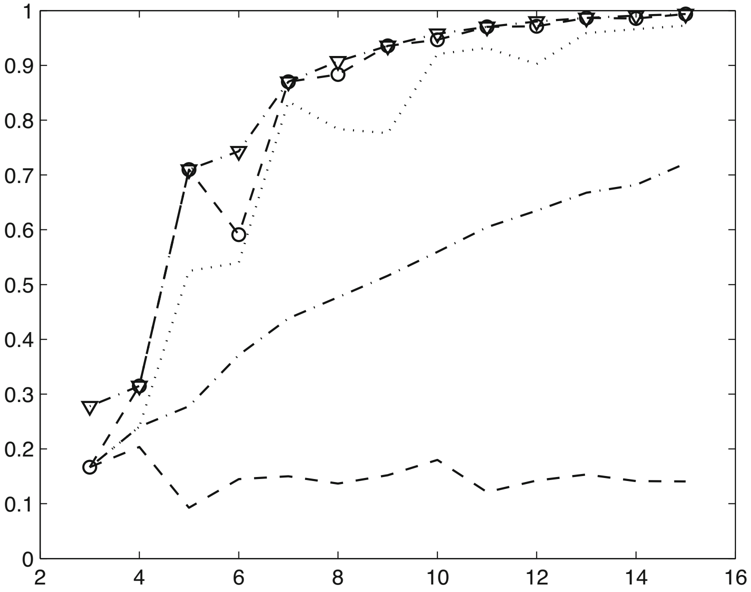


Fig. 10 STV, group

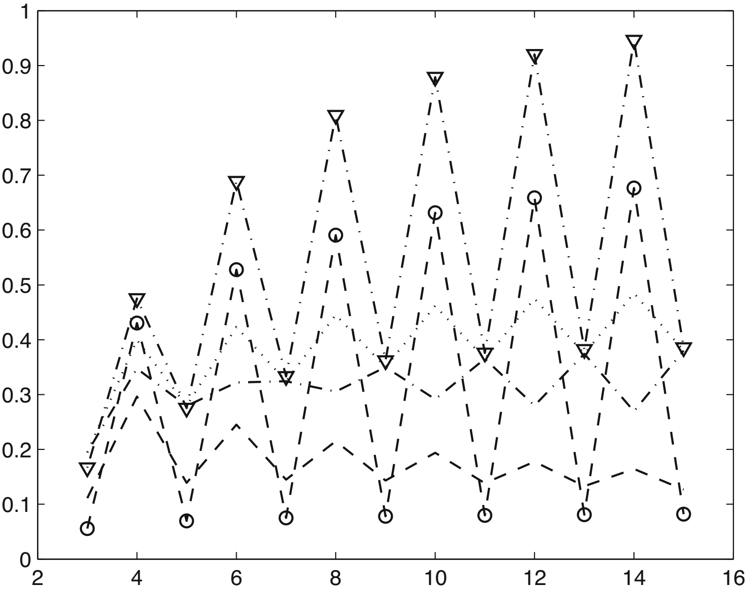


Fig. 11 Copeland rule, individual

5 Conclusion

Studying individual manipulation is convenient for modeling and revealing incentives of separate voters. However, other voters might also take part in manipulation and this assumption can change voters' incentives. Every voter has a group of co-minded people and she can take them into account even if she does not know their exact number. For a single voter it is easier to predict actions of voters of her type. Having the same preference, they also have the same incentives. So, the aim of this work was to consider group actions of co-minded people in manipulation model under incomplete information and compare results with individual manipulation.

In the theoretical part, we considered asymptotic behavior of individual and coalitional manipulation probability for plurality rule and coalitional manipulation probability for Borda rule under *Winner*-PIF. Finally, we proved that individual and coalitional manipulation are equal for scoring rules under *1Winner*-PIF. The computational part of the research illustrates theoretical findings for the case of 3 alternatives and, additionally, allows to observe the behavior of manipulation probabilities for other rules and PIFs.

This work is just the first attempt to combine informational aspect and manipulation by groups in one model. It sheds some light on the problem of their joint influence. Thus, we showed that incomplete information of the types that allow to compute the winner increases manipulability for plurality, Borda, runoff, STV, and Copeland rules. The effect of coalitional manipulation is the same. On the

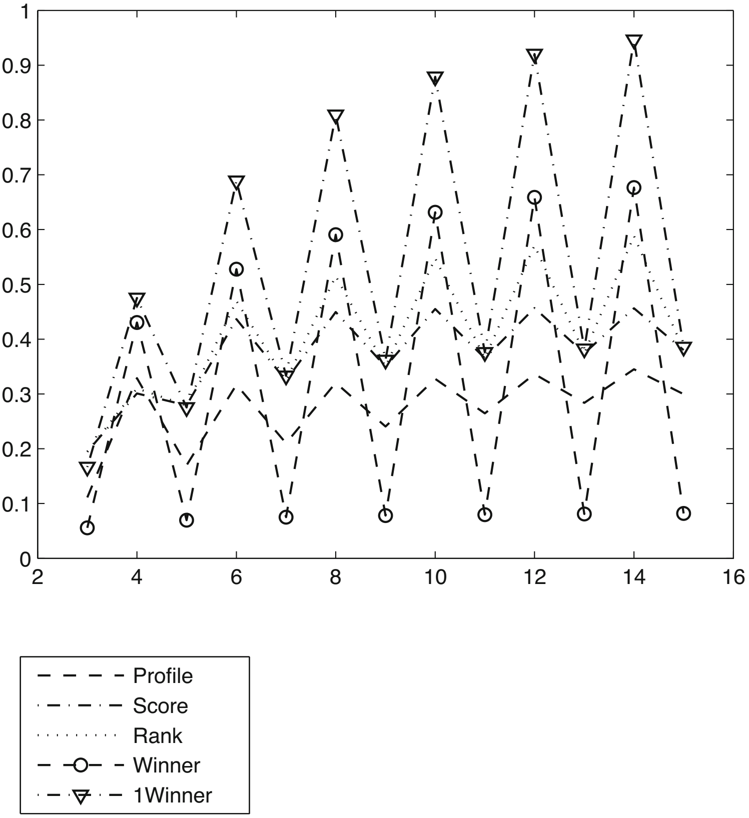


Fig. 12 Copeland rule, group

contrary, for veto rule manipulability decreases under incomplete information and considering also coalitional manipulation makes this effect stronger.

The question that we did not touch here is that some coalition members may decide not to manipulate and that is related to the question of the safety of coalitional manipulation. Additionally, a more significant influence on incentives to manipulation may be expected from voters with a different preference, because they can also manipulate, counter-manipulate, etc. If we add such uncertainty about the actions of other manipulators and consider it together with incomplete information, results may be difficult to predict.

Acknowledgments The author is grateful to Hans Peters and Ton Storcken for lots of discussions and their comments concerning this work. The author acknowledges the support of the Center of Decision Choice and Analysis in HSE and 25th Laboratory of ICS RAS.

References

1. Aleskerov, F., Karabekyan, D., Sanver, R., Yakuba, V.: Evaluation of the degree of manipulability of known aggregation procedures under multiple choice (in Russian). J. New Econ. Assoc. **1**(1), 37–61 (2009)
2. Aleskerov, F., Karabekyan, D., Sanver, M.R., Yakuba, V.: On the degree of manipulability of multi-valued social choice rules. Homo Oeconomicus **28**, 205–216 (2011)
3. Aleskerov, F., Karabekyan, D., Sanver, M. R., Yakuba, V.: On the manipulability of voting rules: the case of 4 and 5 alternatives. Math. Soc. Sci. **64**(1), 67–73 (2012)
4. Aleskerov, F., Karabekyan, D., Ivanov, A., Yakuba, V.: Manipulability of majoritarian rules by coalitions with the same first-ranked alternative. Procedia Comput. Sci. **122**, 993–1000 (2017)
5. Gärdenfors, P.: Manipulation of social choice functions. J. Econ. Theory **13**(2), 217–228 (1976)
6. Gehrlein, W.V., Fishburn, P.C.: Constant scoring rules for choosing one among many alternatives. Qual. Quantity **15**(2), 203–210 (1981)
7. Gibbard, A.: Manipulation of voting schemes: a general result. Econometrica **41**(4), 587–601 (1973)
8. Kelly, J.S.: 4. Minimal manipulability and local strategy-proofness. Soc. Choice Welfare **5**(1), 81–85 (1988)
9. Lepelley, D., Valognes, F.: Voting rules, manipulability and social homogeneity. Public Choice **116**(1–2), 165–184 (2003)
10. Marchant, T.: The probability of ties with scoring methods: some results. Soc. Choice Welfare **18**(4), 709–735 (2001)
11. Nitzan, S.: The vulnerability of point-voting schemes to preference variation and strategic manipulation. Public Choice **47**(2), 349–370 (1985)
12. Pritchard, G., Wilson, M.C.: Exact results on manipulability of positional voting rules. Soc. Choice Welfare **29**(3), 487–513 (2007)
13. Reijngoud, A., Endriss, U.: Voter response to iterated poll information. In: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems, vol. 2, pp. 635–644 (2012)
14. Satterthwaite, M.A.: Strategy-proofness and Arrow's conditions: existence and correspondence theorems for voting procedures and social welfare functions. J. Econ. Theory **10**(2), 187–217 (1975)
15. Slinko, A.: How the size of a coalition affects its chances to influence an election. Soc. Choice Welfare **26**(1), 143–153 (2006)
16. Veselova, Y.A.: Does incomplete information reduce manipulability? Group Decis. Negotiation **29**(3), 523–548 (2020)

Rethinking Probabilistic Topic Modeling from the Point of View of Classical Non-Bayesian Regularization

Konstantin Vorontsov

1 Introduction

Topic modeling is a popular natural language processing technique, which has been actively developed since the late 1990s and still finds many applications [6, 10, 16, 30]. A probabilistic topic model reveals the latent thematic structure of a text document collection representing each topic by a probability distribution over words, and describing each document with a probabilistic mixture of topics.

Topic modeling can be considered as a soft clustering of documents. Unlike conventional hard clustering, a document is allocated among several topical clusters instead of belonging entirely to one cluster. Topic models are also called soft biclustering, since the words are also distributed over topics.

The problem of topic modeling of a text document collection is posed as a low-rank matrix factorization. This is an ill-posed problem, which may have infinitely many solutions. Regularizers are introduced to impose additional restrictions on the model and make the solution more stable [50]. In complex problems, there can be several regularizers.

Starting with the LDA, Latent Dirichlet Allocation model [7], Bayesian learning remains the dominant approach in topic modeling. Its main disadvantage is that the inference process is unique for each model, and the more complex the model, the more difficult its calculations. There are currently no easy ways to automate the inference as well as to construct complex models from the simpler ones. Bayesian regularization is introduced via prior distributions, however, the use of optimization criteria is more convenient and commonly accepted. Many models

K. Vorontsov (✉)
Federal Research Center "Computer Science and Control" of RAS and Institute of Artificial Intelligence of M.V.Lomonosov Moscow State University, Moscow, Russia
e-mail: voron@mlsa-iai.ru

B. Goldengorin, S. Kuznetsov (eds.), *Data Analysis and Optimization*, Springer Optimization and Its Applications 202,
https://doi.org/10.1007/978-3-031-31654-8_24

assume Dirichlet prior distributions, which simplifies Bayesian inference due to the conjugacy property. It was mathematical convenience that predetermined the special role of the Dirichlet distribution in topic modeling, despite the lack of convincing linguistic justifications. Finally, the Bayesian inference is inconvenient to combine with neural network learning procedures [64]. The above barriers prevent the topic modeling from the widespread adoption. Topic models more complicated than LDA are rarely used in the text analysis industry. Hundreds of models remain “the studies for one paper”.

The disadvantages mentioned above are overcome in the Additive Regularization of Topic Models (ARTM), which is an approach based on classical non-Bayesian regularization [53, 55]. As shown in [33], a wide class of Bayesian topic models can be restated in terms of ARTM. After that, it is possible to transfer regularizers from one model to another or to combine the regularizers from various models into a composite model with the required properties. For learning any ARTM models, a general algorithm is used, in which regularizers can be added as plug-ins. The modular technology for ARTM is implemented in the open source library `BigARTM`, http://bigartm.org [21, 57]. Let us emphasize that ARTM is a general framework for inferring and combining topic models rather than another model or method.

In this paper, an even more general approach is proposed. A theorem on the maximization of a smooth function on unit simplices is proven. From this theorem, a family of iterative EM-like algorithms can be inferred for learning topic models of various structures with arbitrary smooth regularizers. In fact, topic modeling becomes a theory of a single theorem.

An iteration of the general algorithm is not much different from the gradient step of a neural network learning process. This observation opens up new perspectives for learning neural topic models, as well as learning neural networks with non-negativity and normalization constraints imposed on some of the parameter vectors.

2 Maximization on Unit Simplices

Define the norm operator, which transforms an arbitrary numeric vector $(x_i)_{i\in I}$ into a non-negative normalized vector:

$$p_i = \operatorname*{norm}_{i\in I}(x_i) = \frac{(x_i)_+}{\sum\limits_{k\in I}(x_k)_+}, \text{ for all } i \in I,$$

where $(x)_+ = \max\{0, x\}$ is a positive part operation. If $x_i \leqslant 0$ for all $i \in I$, then the result of the norm operator is the null vector. Otherwise, the vector $(p_i)_{i\in I}$ lies on the unit simplex and defines a discrete probability distribution on a finite set I.

Theorem 1 *Let the function $f(\Omega)$ be continuously differentiable with respect to the set of vectors $\Omega = (\omega_j)_{j \in J}$, $\omega_j = (\omega_{ij})_{i \in I_j}$. If ω_j is the vector of the local extremum of the mathematical programming problem*

$$f(\Omega) \to \max_{\Omega}, \quad \sum_{i \in I_j} \omega_{ij} = 1, \quad \omega_{ij} \geqslant 0, \quad i \in I_j,\ j \in J$$

and if $\omega_{ij} \frac{\partial f}{\partial \omega_{ij}} > 0$ for some i, then ω_j satisfies the equations

$$\omega_{ij} = \operatorname*{norm}_{i \in I_j}\left(\omega_{ij} \frac{\partial f}{\partial \omega_{ij}}\right). \qquad (1)$$

Proof The Lagrangian of the optimization problem with non-negativity and normalization constraints is

$$\mathscr{L}(\Omega) = f(\Omega) - \sum_{j \in J} \lambda_j \left(\sum_{i \in I_j} \omega_{ij} - 1\right) + \sum_{j \in J} \sum_{i \in I_j} \mu_{ij} \omega_{ij},$$

with λ_j and μ_{ij} factors corresponding to normalization and nonnegativity constraints respectively. Equate the partial derivatives of the Lagrangian to zero, as required by the Karush–Kuhn–Tucker conditions:

$$\frac{\partial \mathscr{L}}{\partial \omega_{ij}} = \frac{\partial f}{\partial \omega_{ij}} - \lambda_i + \mu_{ij} = 0; \quad \mu_{ij} \omega_{ij} = 0. \qquad (2)$$

Multiplying both sides of Eq. (2) by ω_{ij}, one gets

$$\omega_{ij} \frac{\partial f}{\partial \omega_{ij}} = \omega_{ij} \lambda_j.$$

Denote the left side of the equality by A_{ij}. Then $A_{ij} = \omega_{ij} \lambda_j$. According to the condition of the theorem, there exists i such that $A_{ij} > 0$. Consequently, $\lambda_j > 0$. If $\frac{\partial f}{\partial \omega_{ij}} < 0$ for some i, then $\mu_{ij} = \lambda_i - \frac{\partial f}{\partial \omega_{ij}} > 0$, consequently, $\omega_{ij} = 0$.

Combining the equation $\omega_{ij} \lambda_t = A_{ij}$ for $A_{ij} > 0$ with a zero solution $\omega_{ij} = 0$ for $A_{ij} \leqslant 0$, we get $\omega_{ij} \lambda_j = (A_{ij})_+$. Summing these equations over i, express the dual variable: $\lambda_j = \sum_{i \in I_j} (A_{ij})_+$. Substituting λ_j into the formula $\omega_{ij} = \frac{1}{\lambda_j}(A_{ij})_+$, we get the required Eq. (1).

The theorem is proven.

The simple iteration method can be used to solve the system numerically. The update formula (1) is similar to the gradient maximization step $\omega_{ij} = \omega_{ij} + \eta \frac{\partial f}{\partial \omega_{ij}}$. In both cases, the gradient of $f(\Omega)$ is calculated. Three differences are worth noting: instead of an additive gradient step, a multiplicative update is used, the vector is projected onto the unit simplex by the norm operator, and the step size η is irrelevant.

Assuming that (1) is always applicable consider the iterative process

$$\omega_{ij}^{t+1} = \operatorname*{norm}_{i\in I_j}\left(\omega_{ij}^{t}\frac{\partial f(\Omega^t)}{\partial \omega_{ij}^t}\right), \quad t = 0, 1, 2, \ldots$$

Theorem 2 *Let $f(\Omega)$ be an upper bounded, continuously differentiable function, and all Ω^t, starting from some iteration t^0, satisfy the following conditions:*

- $\forall j \in J\ \ \forall i \in I_j\ \ \omega_{ij}^t = 0 \to \omega_{ij}^{t+1} = 0$ *(keeping zeros)*
- $\exists \epsilon > 0\ \ \forall j \in J\ \ \forall i \in I_j\ \ \omega_{ij}^t \notin (0, \epsilon)$ *(separation from zero)*
- $\exists \delta > 0\ \ \forall j \in J\ \ \exists i \in I_j\ \ \omega_{ij}^t \frac{\partial f(\Omega^t)}{\partial \omega_{ij}} \geqslant \delta$ *(nondegeneracy)*

Then $f(\Omega^{t+1}) > f(\Omega^t)$ and $\left|\omega_{ij}^{t+1} - \omega_{ij}^t\right| \to 0$ under $t \to \infty$.

This theorem was proved by I. A. Irkhin as a generalization of his convergence results for the EM-algorithm in topic modeling [27].

3 Probabilistic Topic Modeling

Consider the collection D of text documents composed of terms from a vocabulary W. The *terms* can be words, lemmatized words, n-grams or phrases, depending on the methods used for text preprocessing. Each document $d \in D$ is a sequence of terms $w_1, w_2, \ldots, w_{n_d}$, where n_d means the document length. Under the "bag of words" hypothesis, the order of terms does not matter, then the document d can be represented compactly by a conditional distribution $\hat{p}(w \,|\, d) = \frac{n_{dw}}{n_d}$, where n_{dw} counts how many times the term w occurs in the document d.

Conditional independence is the assumption that each topic generates terms regardless of the document: $p(w \,|\, t) = p(w \,|\, d, t)$. According to this assumption and the law of total probability,

$$p(w \,|\, d) = \sum_{t\in T} p(w \,|\, t)\; p(t \,|\, d) = \sum_{t\in T} \varphi_{wt}\theta_{td}. \tag{3}$$

Probabilistic Topic Model, PTM (3) describes how documents are generated from the known distributions $p(w \,|\, t)$ and $p(t \,|\, d)$. Learning PTM from data is an inverse problem: given a collection estimate model parameters $\varphi_{wt} = p(w \,|\, t)$ and $\theta_{td} = p(t \,|\, d)$. In the matrix form, $\Phi = (\varphi_{wt})_{W\times T}$ and $\Theta = (\theta_{td})_{T\times D}$.

Log-likelihood maximization is usual learning criterion for PTMs:

$$\ln \prod_{d\in D}\prod_{w\in d} p(w \,|\, d)^{n_{dw}} = \sum_{d\in D}\sum_{w\in d} n_{dw} \ln \sum_{t\in T} \varphi_{wt}\theta_{td} \;\to\; \max_{\Phi,\Theta} \tag{4}$$

with linear constraints that make columns nonnegative and normalized:

$$\sum_{w\in W} \varphi_{wt} = 1, \quad \varphi_{wt} \geqslant 0; \qquad \sum_{t\in T} \theta_{td} = 1, \quad \theta_{td} \geqslant 0. \tag{5}$$

For a better understanding of topic modeling consider the learning problem (4) and (5) from four points of view.

Firstly, it is a problem of approximate low-rank matrix factorization. The rank $|T|$ is usually much smaller than both $|D|$ and $|W|$ dimensions. The problem is ill-posed because its solution is not unique: $\Phi\Theta = (\Phi S)(S^{-1}\Theta)$ for infinitely many nonsingular S matrices. Regularization can be added to the main criterion in order to make the solution better defined and more stable using an extra knowledge or data.

Secondly, it is a document auto-encoder. The encoder $f_\Phi \colon \frac{n_{dw}}{n_d} \to \theta_d$ transforms $|W|$-dimensional sparse vector representation of the document $\hat{p}(w\,|\,d)$ into $|T|$-dimensional topical embedding $\theta_d = p(t\,|\,d)$. Linear decoder $g_\Phi \colon \theta_d \to \Phi\theta_d$ attempts to reconstruct the original representation as accurately as possible. Matrix Φ is a parameter of both encoder and decoder. The matrix $\Theta = (\theta_1, \ldots, \theta_D)$ is the result of all documents encoding. This important difference between the matrices Φ and Θ becomes obscure if considered only from the matrix factorization point of view.

Thirdly, it is a soft bi-clustering of both documents and terms by topical clusters T. Each document d and each term w are softly allocated to all clusters according to the distributions $p(t\,|\,d)$ and $p(t\,|\,w)$ respectively, instead of being hardly assigned to only one cluster. The model is also capable of estimating topic distribution for a term in a document $p(t\,|\,d, w)$, for a sentence $p(t\,|\,s)$, and for arbitrary text fragment. In general, we call a distribution $p(t\,|\,x)$ for an object x the *topical embedding* of x.

Fourth, it is a language model that predicts the occurrence of words in documents. Admittedly, conventional topic models are bad competitors in this role. Good word predictions are possible only from local contexts, however, they are violated by the bag-of-words hypothesis. In topic modeling, many ways have been proposed to go beyond this hypothesis and process text as a sequence of terms. Another flaw is more fatal: one can hardly expect that the appearance of a word is determined only by its topics, even if they were estimated from the local context. Deep neural networks based on attention models [51] and transformer architecture, such as BERT [17] and GPT-3 [11] capture the entire set of linguistic phenomena and predict words in a text much better than PTMs and even better than humans do. However, these models are non-interpretable: it is impossible to understand which phenomena are captured, and what each coordinate of the text embedding means.

In contrast to neural models, topical embeddings are interpretable. The topic can tell about itself addressing frequent words from the $p(w\,|\,t)$ distribution, or extracting topical phrases with automatic topic labeling [37] or summarization methods. Moreover, topical embedding $p(t\,|\,x)$ can tell about non-textual object x in words or phrases of natural language.

Thus, topic modeling is aimed not so much at predicting words in documents as revealing the thematic structure of a text collection, determining the semantics of documents and related objects, explaining topics in natural language.

4 Additive Regularization

To solve the ill-posed problem of stochastic matrix factorization, we add regularization criterion $R(\Phi, \Theta)$ to the log-likelihood (4), under non-negativity and normalization constraints (5):

$$L(\Phi, \Theta) = \sum_{d \in D} \sum_{w \in d} n_{dw} \ln \sum_{t \in T} \varphi_{wt} \theta_{td} + R(\Phi, \Theta) \;\to\; \max_{\Phi, \Theta}. \tag{6}$$

Generally, several requirements may be imposed, each formalized by a regularizer $R_i(\Phi, \Theta),\ i = 1, \dots, k$. The scalarization approach for multicriteria optimization leads to the *Additive Regularization for Topic Modeling* (ARTM), proposed in [53]:

$$R(\Phi, \Theta) = \sum_{i=1}^{k} \tau_i R_i(\Phi, \Theta),$$

where non-negative regularization coefficients $\tau_i, i = 1, \dots, k$, are hyperparameters of the learning algorithm.

Theorem 3 *Let the function $R(\Phi, \Theta)$ be continuously differentiable. Then the point (Φ, Θ) of the local extremum of the problem (6), (5) satisfies the system of equations with auxiliary variables $p_{tdw} = p(t \,|\, d, w)$, if zero columns of the matrices Φ, Θ are excluded from the solution:*

$$p_{tdw} = \operatorname*{norm}_{t \in T}\bigl(\varphi_{wt} \theta_{td}\bigr); \tag{7}$$

$$\varphi_{wt} = \operatorname*{norm}_{w \in W}\left(n_{wt} + \varphi_{wt} \frac{\partial R}{\partial \varphi_{wt}}\right); \qquad n_{wt} = \sum_{d \in D} n_{dw} p_{tdw}; \tag{8}$$

$$\theta_{td} = \operatorname*{norm}_{t \in T}\left(n_{td} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}\right); \qquad n_{td} = \sum_{w \in d} n_{dw} p_{tdw}. \tag{9}$$

Proof Proof can be found in [55], but it can be easier derived from Theorem 1. Let's rewrite (7) as follows:

$$p_{tdw} = \operatorname*{norm}_{t \in T}\bigl(\varphi_{wt} \theta_{td}\bigr) = \frac{\varphi_{wt} \theta_{td}}{\sum_s \varphi_{ws} \theta_{sd}} = \frac{\varphi_{wt} \theta_{td}}{p(w \,|\, d)}.$$

Let's apply the formula (1) to the function (6) and substitute the auxiliary variables p_{tdw} in the resulting expressions:

$$\begin{aligned}
\varphi_{wt} &= \operatorname*{norm}_{w\in W}\left(\varphi_{wt}\frac{\partial L}{\partial \varphi_{wt}}\right) = \operatorname*{norm}_{w\in W}\left(\varphi_{wt}\sum_{d\in D}\frac{n_{dw}\theta_{td}}{p(w\,|\,d)} + \varphi_{wt}\frac{\partial R}{\partial \varphi_{wt}}\right)\\
&= \operatorname*{norm}_{w\in W}\left(\sum_{d\in D} n_{dw}p_{tdw} + \varphi_{wt}\frac{\partial R}{\partial \varphi_{wt}}\right);\\
\theta_{td} &= \operatorname*{norm}_{t\in T}\left(\theta_{td}\frac{\partial L}{\partial \theta_{td}}\right) = \operatorname*{norm}_{t\in T}\left(\theta_{td}\sum_{w\in d}\frac{n_{dw}\varphi_{wt}}{p(w\,|\,d)} + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\right)\\
&= \operatorname*{norm}_{t\in T}\left(\sum_{w\in d} n_{dw}p_{tdw} + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\right).
\end{aligned}$$

Zero columns in the Φ and Θ matrices appear in those cases when the positive coordinate condition in Theorem 1 is not satisfied. Zero columns can be removed from the matrices, which is allowed by the condition of the theorem.

The theorem is proven.

A topic t is *degenerate* if $n_{wt} + \varphi_{wt}\frac{\partial R}{\partial \varphi_{wt}} \leqslant 0$ for all $w \in W$.

The degeneracy of the topic is a consequence of the excessively strong sparsing effect of the regularizer R. Zeroing the column of the matrix Φ means that the model prefers to abandon this topic. Reducing the number of topics can be a desirable side effect of regularization.

A document d is *degenerate* if $n_{td} + \theta_{td}\frac{\partial R}{\partial \theta_{td}} \leqslant 0$ for all $t \in T$.

The degeneracy of the document means that the model is not capable to describe it. May be, the document is too short or doesn't match the topical structure of the collection.

Learning a topic model is a numerical solution of the (7)–(9) system. The simple iteration method leads to the Expectation–Maximization (EM) algorithm, in which two steps are performed at each iteration: *E-step* (7) and *M-step* (8) and (9). With a rational implementation of this algorithm each iteration is performed in one linear pass through the collection. For each term w in each document d the topical embedding $p(t|d, w)$ is calculated by the E-step formula and is immediately used to update the counters n_{wt} and n_{td}.

Fast online algorithm, implemented in the `BigARTM` library [57], uses parallelization, splitting the collection into batches, controlling the update rate of Φ matrix, and a few more tricks to increase the computational speed [1, 21]. As a result, `BigARTM` outperforms other freely available topic modeling tools such as Gensim and Vowpal Wabbit by up to 20 times on some tasks [33].

Probabilistic Latent Semantic Analysis (PLSA) is historically the first probabilistic topic model [23]. In ARTM it corresponds to zero regularizer

$$R(\Phi, \Theta) = 0.$$

Latent Dirichlet Allocation (LDA) [7] is the first and most cited Bayesian model. It imposes restrictions on the columns of the Φ and Θ matrices in the form of Dirichlet prior distributions. In ARTM it corresponds to the cross-entropy regularizer [33]

$$R(\Phi, \Theta) = \sum_{t \in T} \sum_{w \in W} \beta_{wt} \ln \varphi_{wt} + \sum_{d \in D} \sum_{t \in T} \alpha_{td} \ln \theta_{td}. \qquad (10)$$

If the hyperparameters β_{wt}, α_{td} are positive, then the regularization smoothes the conditional distributions φ_{wt}, θ_{td} bringing them closer to the given vectors $\operatorname{norm}_w(\beta_{wt})$, $\operatorname{norm}_t(\alpha_{td})$. If β_{wt}, α_{td} are negative, then the effect of the regularizer is sparsing instead of smoothing, as can be seen from the M-step formulas:

$$\varphi_{wt} = \operatorname*{norm}_{w \in W}\bigl(n_{wt} + \beta_{wt}\bigr); \qquad \theta_{td} = \operatorname*{norm}_{t \in T}\bigl(n_{td} + \alpha_{td}\bigr).$$

In the Bayesian interpretation, hyperparameters are bounded from below: $\beta_{wt} > -1$, $\alpha_{td} > -1$, due to the properties of the Dirichlet distribution. Therefore, sparsing effect is restricted and weak. There are no such restrictions in the ARTM interpretation, since a priori Dirichlet distributions are not introduced into the model.

5 Comparison with Bayesian Learning

Let for generality X be the observed data set (e.g. the text documents collection), $p(X \mid \Omega)$ be a probabilistic data model with Ω parameters (e.g. the Φ and Θ matrices), $p(\Omega \mid \gamma)$ be *a priori distribution* of model parameters with hyperparameters γ (in the LDA model, the Dirichlet distributions with hyperparameters β_{wt}, α_{td}). Then the *posterior distribution* of Ω parameters is given by the Bayes' formula:

$$p(\Omega \mid X, \gamma) = \frac{p(\Omega, X \mid \gamma)}{p(X \mid \gamma)} \propto p(X \mid \Omega)\, p(\Omega \mid \gamma),$$

where the symbol $\propto$ means "equals up to normalization". Bayesian inference is useful in many data analysis problems where we do something with model parameters: testing statistical hypotheses, interval estimating, sampling, etc. However, in the practice of topic modeling Bayesian inference is performed only to get a point estimate of the Ω parameters:

$$\Omega := \arg\max_{\Omega} p(\Omega \mid X, \gamma).$$

Maximizing a posteriori (MAP) gives a point estimate for Ω, bypassing the intermediate step of the approximate and tedious posterior inference:

$$\Omega := \arg\max_{\Omega}\bigl(\ln p(X \,|\, \Omega) + \ln p(\Omega \,|\, \gamma)\bigr).$$

The logarithm of the prior distribution can be considered as a classical non-Bayesian regularization criterion $R(\Omega) = \ln p(\Omega \,|\, \gamma)$. In this form, it can be separated from a particular model and brought to another model.

Additive regularization generalizes log-priors to any regularizers, including those that do not have a probabilistic nature, as well as their linear combinations, without violating the convergence properties:

$$\Omega := \arg\max_{\Omega}\Bigl(\ln p(X \,|\, \Omega) + \sum_{i} \tau_i R_i(\Omega)\Bigr).$$

The main disadvantage of Bayesian inference is that it requires sophisticated calculations unique to each model, which makes it difficult to regularly combine multiple requirements and constraints. In Bayesian learning, there are no conventional regularization mechanisms based on criteria, since there is actually no optimization problem for Ω. Additional information can be introduced either through the prior distribution or through the very structure of the model. If the prior distributions are not Dirichlet distributions, then the inference becomes noticeably more complicated. Non-unified inference incur implementing and testing costs for each model.

The Dirichlet distribution plays a special role in Bayesian topic modeling. Although it has no convincing linguistic justification, most models are built on it in the literature. The reason is solely in the mathematical convenience of the Dirichlet prior conjugated with a multinomial distribution. In ARTM there is no reason to prefer the Dirichlet distribution to other regularizers.

The additivity of regularizers leads to a modular topic modeling technology, which is implemented in the `BigARTM` open source project [57]. In applications, composite models with desired properties can be built by adding ready-to-use regularizers from the library, without new mathematical calculations and coding. The development of such a technology within the Bayesian framework is hardly possible.

6 Overview of Models and Regularizers

Many topic models, originally formulated in the Bayesian paradigm can be reformulated in terms of classical non-Bayesian regularization [33].

Combination of smoothing, sparsing and decorrelation regularizers has proven itself well in practice in many studies [54, 55, 61]. Topic decorrelation regularizer

$$R(\Phi) = -\frac{\tau}{2}\sum_{t\in T}\sum_{s\in T\setminus t}\sum_{w\in W} \varphi_{wt}\varphi_{ws}$$

not only makes the topics more diverse, but also groups common words into separate background topics and purges all other topics from them [49].

Semi-supervised topic models use the smoothing regularizer of Φ matrix to set the seed words for some of the topics so that subject topics of interest can crystallize in their place during the iterative process. This technique has been used for searching rare topics in social media, such as symptoms, diseases, and their treatments [41, 42]; crime and extremism [36, 47]; ethnicities and interethnic relations [8, 34, 40]. For example, to search for a given number of ethno-relevant topics within the ARTM framework, smoothing regularization was applied using the vocabulary of ethnonyms. After that, the topic model was able to determine how topics are specialized by ethnicity [2, 3]. In particular, multi-ethnic topics were found, helping sociologists to identify the aspects of interethnic relations.

Multimodal topic model describes documents containing not only words, but also terms of other modalities: categories, authors, time, tags, entities, users, etc. Each modality $m \in M$ has its own dictionary of terms W^m, own matrix Φ^m with normalized columns, and own log-likelihood criterion. The problem is to maximize the weighted sum of these criteria over modalities:

$$\sum_{m\in M} \tau_m \sum_{d\in D} \sum_{w\in W^m} n_{dw} \ln \sum_{t\in T} \varphi^m_{wt}\theta_{td} + R\bigl(\{\Phi^m\}, \Theta\bigr) \;\to\; \max_{\Phi,\Theta}. \tag{11}$$

Multimodal data helps to determine the document topics more accurately. Conversely, the topic model can be used to reveal the semantics of modalities or predict missing modality metadata.

Classification topic model is a special case of the multimodal PTM with the modality C of categories or classes. The model predicts class probabilities for a document $p(c\,|\,d)$ with a linear classifier using topic probabilities $p(t\,|\,d)$ as features:

$$p(c\,|\,d) = \sum_{t\in T} p(c\,|\,t)p(t\,|\,d) = \sum_{t\in T} \varphi_{ct}\theta_{td}.$$

Experiments showed that this topic model outperforms conventional multiclass classification methods on large text collections with a large number of unbalanced, overlapping, interdependent classes [46]. Similar results on the same collections were reproduced for the multimodal ARTM in [56]. *Unbalanced* classes can contain both a small and a very large number of documents. *Overlapping* classes means that a document may belong to many classes. *Interdependent* classes share terms and topics, therefore, they can compete and interfere when classifying a document.

Multilingual topic model is another case of multimodal PTM, when languages act as modalities. Linking parallel texts into a common document is enough for synchronizing topics across languages in cross-language document search tasks [58]. Regularizers based on bilingual dictionaries have been proposed in [18], however, the parallel texts linking remains the main contribution to the search quality.

Triple matrix topic model arises from the assumption that topics are generated not by a document, but by one of the modalities, for example, categories, authors, or tags. The author-topic model ATM [45], the tag weighted topic model TWTM [35], and the model for detecting behaviour dynamics in video [24] can be viewed as triple matrix factorization:

$$p(w\,|\,d) = \sum_{t\in T} p(w\,|\,t) \sum_{a\in A} p(t\,|\,a)p(a\,|\,d) = \sum_{t\in T} \varphi_{wt} \sum_{a\in A} \psi_{ta}\pi_{ad},$$

where A is a dictionary of authors, tags, or behaviours respectively. The EM-like algorithm given in [33] for this model can be easily obtained as a corollary of the maximization theorem on unit simplices.

Hierarchical topic models divide topics into smaller subtopics recursively. There is a wide variety of approaches and methods for learning and evaluating topical hierarchies [62]. The top-down level-wise strategy based on ARTM has been proposed in [15] and improved in [5]. The hierarchy is built from top to bottom, each child level having greater number of topics than the parent level has. Each level is a conventional flat topic model, which is linked with the parent level by conditional probabilities $\psi_{st} = p(s\,|\,t)$ of subtopics $s \in S$ in parent topics $t \in T$. The regularizer tries to approximate parent topics φ_{wt} by a probabilistic mixture of child topics φ_{ws} with coefficients ψ_{st}:

$$R(\Phi, \Psi) = \tau \sum_{t\in T} \sum_{w\in W} n_{wt} \ln \sum_{s\in S} \varphi_{ws}\psi_{st}. \tag{12}$$

The maximization of $R(\Phi, \Psi)$ coincides up to notation with the main topic modeling task (4), with parent topics t considered as *pseudo-documents* with term frequencies $n_{wt} = n_t\varphi_{wt}$. This regularizer can be implemented by simply adding $|T|$ pseudo-documents to the collection before building each child level. The linking matrix Ψ is produced by the model in the columns of the Θ matrix corresponding to pseudo-documents.

Multimodal hierarchical topic models perform well in document-by-document topic-based search [25, 26]. Combining decorrelation, sparsing, and smoothing regularizers along with modalities of n-grams, authors and categories significantly improves search quality. In experiments with exploratory search in technology blogs, both precision and recall reach 90%. Optimal (in terms of search quality) dimension of topical embeddings at the third level of the hierarchy turned out to be several times higher than that of the flat model. This means that the gradual fragmentation of topics into smaller subtopics allows topical embeddings to keep more useful information about documents.

Topic model for mining polarized opinions is actually a two-level hierarchy, in which the upper level determines topics in news [20]. The second level is based on unusual modalities, dividing the topic into subtopics with polarized opinions about the topic. The modalities are: named entities with positive and negative sentiments, named entities with their semantic roles, triplets "subject, predicate, object". Experiments have shown that each of the three modalities is important for improving the polarized opinions detection. A similar two-level hierarchy has been proposed in [43], where syntactic modalities were used at the child level to divide parent level topics into more detailed client intents in the collection of contact center dialogs.

Hyperparameter optimization strategies. Additive regularization loses to Bayesian modeling in only one aspect. The more regularizers are used, and the more regularization coefficients have to be selected, the more careful balancing they require. Early studies have shown that regularizers can interfere with each other, and that understanding their interactions leads to sequential strategies of adding regularizers to the model [54].

Adding regularizers during the iteration process in the order $\{\Phi$ decorrelation, Θ sparsing, Φ smoothing$\}$ has been proven to be a successful strategy for topic-based exploratory search [25, 61]. In further experiments, the hyperparameter space was extended with modality weights, pseudo-document weights and the number of topics at each level in the hierarchical model [26]. When regularizer starts from a given iteration, learning algorithm must be restarted from this point many times with hyperparameter values iterated over a coarse grid. The model quality is controlled visually by multiple criteria during the iterative process.

Later, this technique was extended and implemented in TopicNet open source library, which operates on top of `BigARTM` hiding technical details from the user [12]. The user specifies only the high-level regularization strategy. TopicNet automates computational experiments on hyperparameter optimization, providing logging and visualization.

A more general framework for hyperparameter optimization in ARTM is based on evolutionary algorithms and representation of a learning process as a multi-stage strategy for changing hyperparameters [31]. Later this approach was extended by a surrogate model for PTM evaluation, which reduced the time for automatic selection of hyperparameters [32].

7 Hypergraph Topic Models of Transactional Data

Topic models of text collections describe occurrences of words in documents. Multimodal topic models describe documents that may contain the terms of several modalities: words, tags, categories, authors, etc. In all these cases, the model describes pairwise interactions between documents and terms. In more complex applications, the initial data may describe transactions between three or more objects. For example, "user u clicked ad b on page s" in an advertising network; "user u wrote word w on blog page d" in a social network; "buyer b bought item g from seller s" in a sales network; "client u departed from airport x to airport y by airline a" in passenger air transportation; "user u rated the film f in a contextual situation s" in a recommender system. Another modality could be transaction time.

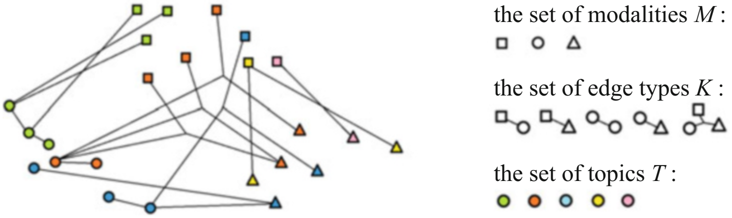


Fig. 1 An example of a hypergraph with vertices of three modalities, edges of five types, and five topics

In all of the examples above, a multi-object transaction can not be reduced to the pair interactions.

Transactional data can be represented by a hypergraph $\Gamma = \langle V, E\rangle$ defined by the set of term vertices V and the set of transaction edges E. Each edge e of E is a subset of two or more vertices, $e \subset V$. The task is to restore unknown topic distributions of vertices $p(t \,|\, v)$ from the observed dataset of transactions.

Each vertex has modality m from the set M. Denote by V_m the set of vertices having modality m. In conventional topic models, there are two modalities: terms $V_1 = W$ and documents $V_2 = D$; each edge transaction $e = (d, w)$ means that the term w occurs in the document d; thus, the hypergraph is a bipartite graph.

In more complicated applications, transactions can be of various types. For example, in the advertising network, along with triplet data "user u clicked ad b on page s", there may be pair data "user u visited page s", "page s contains term w", "ad b contains term w", "user's u query contains term w".

Let K be the set of transaction types. *Transactional data* of type k is a dataset of edges $E_k \subset E$. Each edge $e \in E_k$ occurs in the dataset n_{ke} times, having a latent topic $t \in T$. Figure 1 shows an example of a hypergraph.

Assume that each transaction $e \in E$ has one dedicated vertex d called *container*, and denote the edge by $e = (d, x)$, where x is the set of all other vertices of the edge. Similar to a document, a container has a distribution of topics $p(t \,|\, d)$. Denote the set of all containers by D.

We accept several hypotheses of conditional independence. Assume that neither the distribution of topics $p(t \,|\, d)$ in a container d, nor distributions of vertices in topics $p(v \,|\, t)$ depend on the type of the edge k. Next, suppose that the process of generating the edge $(d, x) \in E_k$ consists of two steps. First, a topic t is generated from the distribution $p(t \,|\, d)$. Then the set of vertices $x \subset V$ is generated so that each vertex $v \in x$ of the modality m is generated from the distribution $p(v \,|\, t)$ over the set V_m independently of the other edge vertices.

The topic model expresses the probabilities of hypergraph edge through conditional distributions associated with their vertices:

$$p(x \,|\, d) = \sum_{t \in T} p(t \,|\, d) \prod_{v \in x} p(v \,|\, t) = \sum_{t \in T} \theta_{td} \prod_{v \in x} \varphi_{vt}.$$

In matrix notation, the model parameters are matrices Θ and Φ_m, $m \in M$, as in the multimodal topic model (11).

Learning the hypergraph model is log-likelihoods maximization for all edge types k with weights τ_k, under the usual non-negativity and normalization constraints, improved by the regularizer $R(\Phi, \Theta)$:

$$\sum_{k\in K} \tau_k \sum_{dx\in E_k} n_{kdx} \ln\biggl(\sum_{t\in T} \theta_{td} \prod_{v\in x} \varphi_{vt}\biggr) + R(\Phi, \Theta) \to \max_{\Phi,\Theta}; \tag{13}$$

$$\sum_{v\in V_m} \varphi_{vt} = 1,\ \varphi_{vt} \geqslant 0; \qquad \sum_{t\in T} \theta_{td} = 1,\ \theta_{td} \geqslant 0.$$

Theorem 4 *Let the function $R(\Phi, \Theta)$ be continuously differentiable. Local maximum point (Φ, Θ) of the problem (13) satisfies the system of equations with respect to model parameters φ_{vt}, θ_{td} and auxiliary variables $p_{tdx} = p(t \,|\, d, x)$, if zero columns of the matrices Φ_m, Θ are excluded from the solution:*

$$p_{tdx} = \operatorname*{norm}_{t\in T}\biggl(\theta_{td} \prod_{v\in x} \varphi_{vt}\biggr); \tag{14}$$

$$\varphi_{vt} = \operatorname*{norm}_{v\in V_m}\biggl(\sum_{k\in K} \sum_{dx\in E_k} [v \in x]\, \tau_k n_{kdx} p_{tdx} + \varphi_{vt} \frac{\partial R}{\partial \varphi_{vt}}\biggr); \tag{15}$$

$$\theta_{td} = \operatorname*{norm}_{t\in T}\biggl(\sum_{k\in K} \sum_{dx\in E_k} \tau_k n_{kdx} p_{tdx} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}\biggr); \tag{16}$$

Proof Let us apply Theorem 1 on maximization on unit simplices, extracting the expression for the auxiliary variables p_{tdx} defined in (14):

$$\begin{aligned}
\varphi_{vt} &= \operatorname*{norm}_{v\in V_m}\biggl(\varphi_{vt} \sum_{k\in K} \tau_k \sum_{dx\in E_k} n_{kdx} \frac{\theta_{td}}{p(x \,|\, d)} \frac{\partial}{\partial \varphi_{vt}} \prod_{u\in x} \varphi_{ut} + \varphi_{vt} \frac{\partial R}{\partial \varphi_{vt}}\biggr) \\
&= \operatorname*{norm}_{v\in V_m}\biggl(\sum_{k\in K} \sum_{dx\in E_k} \tau_k n_{kdx} [v \in x] p_{tdx} + \varphi_{vt} \frac{\partial R}{\partial \varphi_{vt}}\biggr); \\
\theta_{td} &= \operatorname*{norm}_{t\in T}\biggl(\theta_{td} \sum_{k\in K} \tau_k \sum_{x\in d} n_{kdx} \frac{1}{p(x \,|\, d)} \prod_{v\in x} \varphi_{vt} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}\biggr) \\
&= \operatorname*{norm}_{t\in T}\biggl(\sum_{k\in K} \sum_{x\in d} \tau_k n_{kdx} p_{tdx} + \theta_{td} \frac{\partial R}{\partial \theta_{td}}\biggr).
\end{aligned}$$

The theorem is proven.

The hypergraph model is a broad generalization of conventional PTMs. Despite this, the derivation of the EM-algorithm out of Theorem 1 is no more difficult than in the conventional case. This algorithm is implemented in `BigARTM` project.

8 Hypergraph Recommender Topic Models

Let U be a finite set of users, I be a finite set of items that users can take or prefer. The probabilistic topic model predicts user preferences:

$$p(i \,|\, u) = \sum_{t \in T} p(i \,|\, t)\, p(t \,|\, u).$$

This model is equivalent to the topic model of a text collection, up to terminology: documents → users, terms → items, topics → interests. Following the analogy, the bag-of-words transforms into the bag-of-transactions hypothesis. In this case, dataset can be considered as n_{ui} counters of the user u transactions with the item i. Depending on the application, transactions may be purchases, visits, likes, etc.

There is a well-known "cold start" issue in recommender systems. Nothing to recommend to a new user, since there is no history of his preferences. Nobody to recommend a new item, since no one has chosen it yet. To solve this problem, additional data about users and items can be involved. In particular, these may be data n_{ua} on the user attributes $a \in A$ or data n_{ib} on the item properties $b \in B$. If items have text descriptions, then B is a dictionary of terms used in these descriptions. Such recommender systems are called, respectively, attribute-aware and content-aware.

Users' advice to each other can also be used as additional data. These are pairwise interactions between users $n_{uu'}$ or trust-aware data.

User preferences may change over time or depend on the situation. To take into account such information, two more modalities are introduced: the set of situations C and the set of time intervals J. Interactions between them are described by transactions of three or more terms, for example, n_{uic} for "user u selected item i in situation c", or n_{uicj} for "user u selected item i in situation c in time interval j. Such systems are called, respectively, context-aware and time-aware.

Many types of $***$-aware models were introduced separately in the literature [14]. The hypergraph model can combine them all and learn topical embeddings for any interacting terms regardless their nature, Fig. 2.

The recommender system data is different from the text collections as it has no natural analogue of a document or container. The set of transactions (u, i) may increase with time for both the user u and the item i, unlike unchanging documents.

Assume that the edges of the hypergraph $x \subset V$ do not contain container vertex. The edge generative process first generates a topic t from the distribution $\pi_t = p(t)$ which is common to the entire collection. Then the vertices $v \in x$ are generated independently of each other from distributions $\varphi_{vt} = p(v \,|\, t)$ over modalities V_m:

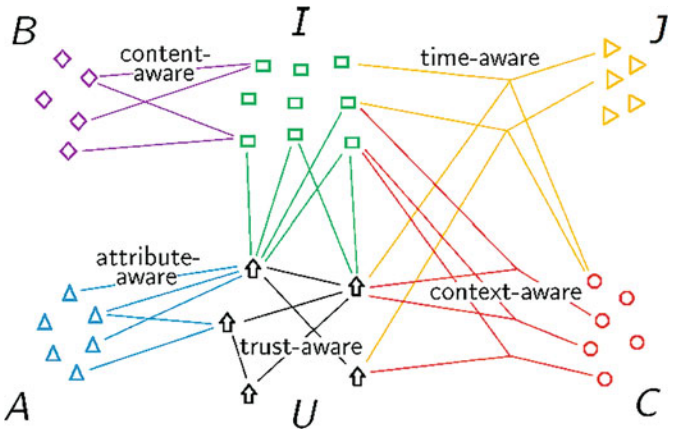


Fig. 2 Types of transactions between six modalities in a recommender system: users U, items I, user attributes A, item properties B, contextual situations C, time intervals J

$$p(x) = \sum_{t\in T} p(t) \prod_{v\in x} p(v\,|\,t) = \sum_{t\in T} \pi_t \prod_{v\in x} \varphi_{vt}.$$

Topic models, in which documents act as one of the modalities, are called symmetric [52]. As before, maximization problem is for regularized log-likelihood under normalization and non-negativity constraints:

$$\sum_{k\in K} \tau_k \sum_{x\in E_k} n_{kx} \ln\biggl(\sum_{t\in T} \pi_t \prod_{v\in x} \varphi_{vt}\biggr) \;+\; R(\Phi,\pi) \to \max_{\Phi,\pi}; \tag{17}$$

$$\sum_{v\in V_m} \varphi_{vt} = 1,\ \varphi_{vt} \geqslant 0; \qquad \sum_{t\in T} \pi_t = 1,\ \pi_t \geqslant 0.$$

Theorem 5 *Let the function $R(\Phi,\pi)$ be continuously differentiable. Local maximum point (Φ,π) of the problem (17) satisfies the system of equations with respect to model parameters φ_{vt}, π_t and auxiliary variables $p_{tx} = p(t\,|\,x)$, if zero columns of the Φ_m matrices are excluded from the solution:*

$$p_{tx} = \operatorname*{norm}_{t\in T}\biggl(\pi_t \prod_{v\in x} \varphi_{vt}\biggr). \tag{18}$$

$$\varphi_{vt} = \operatorname*{norm}_{v\in V_m}\biggl(\sum_{k\in K}\sum_{x\in E_k} [v\in x]\, \tau_k n_{kx} p_{tx} + \varphi_{vt}\frac{\partial R}{\partial \varphi_{vt}}\biggr); \tag{19}$$

$$\pi_t = \operatorname*{norm}_{t\in T}\biggl(\sum_{k\in K}\sum_{x\in E_k} \tau_k n_{kx} p_{tx} + \pi_t\frac{\partial R}{\partial \pi_t}\biggr). \tag{20}$$

Proof follows straightforwardly from the maximization theorem on unit simplices, as in the case of the previous theorem.

In BigARTM the symmetrized model is not implemented, but it is not difficult to simulate it. To to this, the collection is split in some way into documents (for example, by transaction time), then a strong regularizer is introduced for smoothing the columns of the Θ matrix towards the (n_t) vector summed over all documents.

9 Sequential Text Topic Models

The bag-of-words hypothesis is one of the most criticized assumptions in topic modeling. Many approaches was proposed in the literature in order to go beyond the bag-of-words restrictive assumption, either completely or partially.

Topic models with n-grams exploit the fact that stable combinations of n consecutive words often, though not always, represent subject domain terms or names. The n-grams may tell much more about topics than the same words treated independently. Topics built on the n-gram dictionary are better interpretable, than those built on unigrams [29, 60]. There are two approaches to using n-grams in topic modeling. In the first one, the dictionary of n-grams is built at the stage of text preprocessing using automatic extraction of terms, keywords, or collocations [19]. Then, the n-gram dictionary is used as a modality. The second approach is more complicated, in which topic modeling is combined with n-gram extraction [59, 60]. Concentration of distribution $p(t \mid w)$ in one or more topics is usually a strong indication that the n-gram w is a subject domain term.

Word network topic model predicts the appearance of a word nearby to another word, instead of predicting it in the document. “Nearby” means, say, no more than 10 words away or in one sentence. Define for each word $u \in W$ a pseudo-document d_u consisting of all words that occur nearby to the word u throughout the collection. Denote by n_{uw} the number of occurrences of the word w in a pseudo-document d_u.

The word network topic model WNTM [65] and the earlier word topic model WTM [13] predict a word in the neighborhood of other word:

$$p(w \mid u) = \sum_{t \in T} p(w \mid t)\, p(t \mid d_u) = \sum_{t \in T} \varphi_{wt}\theta_{tu}.$$

The log-likelihood can be used either as a regularizer for other topic model, or as the main learning criterion. In the first case, topic model is learned by the document collection augmented by pseudo-documents. In the second case, only pseudo-documents are used:

$$\sum_{u,w \in W} n_{uw} \ln \sum_{t \in T} \varphi_{wt}\theta_{tu} \;\to\; \max_{\Phi,\Theta}.$$

According to *the distributional hypothesis* the meaning of a word is determined by the distribution of all words, in whose environment it occurs [22]. Words found in similar contexts have similar semantics, and in the model they should receive similar embeddings. Word embeddings implemented in the `word2vec` program [38, 39] are also learned from word co-occurrence data. They encapsulate the meanings of words so well that paired associations turn into vector equalities:

$$\text{king} - \text{queen} = \text{man} - \text{woman};$$

$$\text{Moscow} - \text{Beijing} = \text{Russia} - \text{China}.$$

The additively regularized WNTM also has this property [44], unlike conventional topic models. Moreover, topical embeddings are coordinate-wise interpretable, unlike word2vec and neural embeddings.

Sentence topic model can be considered as a special case of hypergraph topic model. Vertices of the hypergraph are words, edges are sentences. This approach is equivalent to the sentence topic model senLDA [4] and Twitter-LDA short message model [63] first proposed in terms of Bayesian learning. The hypergraph representation gives a lot of freedom in defining edges. These can be not only sentences, but also noun phrases, syntagmas, lexical chains, and in general any group of words, with reasonable assumption that they are generated by a common topic.

E-step regularization. The idea behind using intradocument word order data is to impose regularization constraints on topical embeddings $p_{tdw} = p(t \,|\, d, w)$. They specialize topical embeddings $p(t \,|\, w)$ from the global context of the collection to the narrower document context. Further narrowing of the context to the local neighborhoods of words requires processing the document as a sequence of word embeddings.

Define the regularizer $R(\Pi, \Phi, \Theta)$ as a function of the matrices Φ, Θ and a three-dimensional matrix of auxiliary variables $\Pi = (p_{tdw})_{T \times D \times W}$. According to (7), the elements of Π matrix are functions of Φ and Θ matrices. Therefore, the regularizer has a form $\tilde{R}(\Phi, \Theta) = R(\Pi(\Phi, \Theta), \Phi, \Theta)$. Then, Theorem 3 can be applied to it. However, it is more convenient to write the system of equations in terms of the partial derivatives of the regularizer R rather than $\tilde{R}$.

Consider the problem of the regularized log-likelihood maximization under non-negativity and normalization constraints (5):

$$\sum_{d \in D} \sum_{w \in d} n_{dw} \ln \sum_{t \in T} \varphi_{wt} \theta_{td} + R(\Pi(\Phi, \Theta), \Phi, \Theta) \;\to\; \max_{\Phi, \Theta}. \tag{21}$$

Theorem 6 *Let the function $R(\Pi, \Phi, \Theta)$ be continuously differentiable and does not depend on p_{tdw} for all $w \notin d$. Then the point (Φ, Θ) of the local extremum of the problem (21), (5) satisfies the system of equations with auxiliary variables p_{tdw} and $\tilde{p}_{tdw}$, if zero columns of the matrices Φ, Θ are excluded from the solution:*

$$p_{tdw} = \operatorname*{norm}_{t\in T}\big(\varphi_{wt}\theta_{td}\big);$$

$$\tilde{p}_{tdw} = p_{tdw}\left(1 + \frac{1}{n_{dw}}\left(\frac{\partial R}{\partial p_{tdw}} - \sum_{z\in T} p_{zdw}\frac{\partial R}{\partial p_{zdw}}\right)\right); \tag{22}$$

$$\varphi_{wt} = \operatorname*{norm}_{w\in W}\left(\sum_{d\in D} n_{dw}\tilde{p}_{tdw} + \varphi_{wt}\frac{\partial R}{\partial \varphi_{wt}}\right); \tag{23}$$

$$\theta_{td} = \operatorname*{norm}_{t\in T}\left(\sum_{w\in d} n_{dw}\tilde{p}_{tdw} + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\right). \tag{24}$$

Proof First, we define the function $p_{zdw}(\Phi, \Theta) = \frac{\varphi_{wz}\theta_{zd}}{\sum_t \varphi_{wt}\theta_{td}}$ and find its partial derivatives. For any $t, z \in T$

$$\begin{aligned}\varphi_{wt}\frac{\partial p_{zdw}}{\partial \varphi_{wt}} &= \varphi_{wt}\frac{[z{=}t]\theta_{td}\sum_u \varphi_{wu}\theta_{ud} - \theta_{td}\varphi_{wz}\theta_{zd}}{(\sum_u \varphi_{wu}\theta_{ud})^2}\\ &= p_{tdw}[z{=}t] - p_{tdw}p_{zdw};\end{aligned} \tag{25}$$

$$\begin{aligned}\theta_{td}\frac{\partial p_{zdw}}{\partial \varphi_{td}} &= \theta_{td}\frac{[z{=}t]\varphi_{wt}\sum_u \varphi_{wu}\theta_{ud} - \varphi_{wt}\varphi_{wz}\theta_{zd}}{(\sum_u \varphi_{wu}\theta_{ud})^2}\\ &= p_{tdw}[z{=}t] - p_{tdw}p_{zdw};\end{aligned} \tag{26}$$

Note that the resulting expressions (25) and (26) are the same.

Let us introduce an auxiliary function Q of the variables Π, Φ, Θ:

$$Q_{tdw}(\Pi, \Phi, \Theta) = \frac{\partial R(\Pi, \Phi, \Theta)}{\partial p_{tdw}} - \sum_{z\in T} p_{zdw}\frac{\partial R(\Pi, \Phi, \Theta)}{\partial p_{zdw}}.$$

Let us differentiate the superposition $\tilde{R}(\Phi, \Theta) = R(\Pi(\Phi, \Theta), \Phi, \Theta)$, given that $\partial p_{zdw'}/\partial \varphi_{wt} = 0$ if $w \neq w'$; $\partial p_{zd'w}/\partial \theta_{td} = 0$ if $d \neq d'$; $\partial R/\partial p_{tdw} = 0$ if $w \notin d$:

$$\varphi_{wt}\frac{\partial \tilde{R}}{\partial \varphi_{wt}} = \varphi_{wt}\frac{\partial R}{\partial \varphi_{wt}} + \sum_{d\in D}\varphi_{wt}\sum_{z\in T}\frac{\partial R}{\partial p_{zdw}}\frac{\partial p_{zdw}}{\partial \varphi_{wt}}; \tag{27}$$

$$\theta_{td}\frac{\partial \tilde{R}}{\partial \theta_{td}} = \theta_{td}\frac{\partial R}{\partial \theta_{td}} + \sum_{w\in d}\theta_{td}\sum_{z\in T}\frac{\partial R}{\partial p_{zdw}}\frac{\partial p_{zdw}}{\partial \theta_{td}}. \tag{28}$$

Using (25) and (26), we get the identity

$$\varphi_{wt}\sum_{z\in T}\frac{\partial R}{\partial p_{zdw}}\frac{\partial p_{zdw}}{\partial \varphi_{wt}} = \theta_{td}\sum_{z\in T}\frac{\partial R}{\partial p_{zdw}}\frac{\partial p_{zdw}}{\partial \theta_{td}} = p_{tdw}Q_{tdw}.$$

Let us substitute the resulting expressions into (27) and (28), which we then substitute into the system of equations from Theorem 3:

$$p_{tdw} = \operatorname*{norm}_{t\in T}\bigl(\varphi_{wt}\theta_{td}\bigr);$$

$$\varphi_{wt} = \operatorname*{norm}_{w\in W}\biggl(\sum_{d\in D} n_{dw}p_{tdw} + \sum_{d\in D} Q_{tdw}p_{tdw} + \varphi_{wt}\frac{\partial R}{\partial \varphi_{wt}}\biggr); \tag{29}$$

$$\theta_{td} = \operatorname*{norm}_{t\in T}\biggl(\sum_{w\in d} n_{dw}p_{tdw} + \sum_{w\in d} Q_{tdw}p_{tdw} + \theta_{td}\frac{\partial R}{\partial \theta_{td}}\biggr). \tag{30}$$

Substituting of the auxiliary variable $\tilde{p}_{tdw}$ according (22) allows us to rewrite the equations (29) and (30) in the required form (23) and (24).

The theorem is proven.

In the EM-algorithm, topical embeddings $p_{tdw} = p(t|d, w)$ are calculated for each word w in the document d. Then they are transformed into new vectors $\tilde{p}_{tdw}$ and used at the M-step instead of p_{tdw}. We call this technique *E-step regularization* or *E-step post-processing*. This is an optional procedure, which does not affect the implementation of other computations in the EM-algorithm. This approach was used in [48] to improve the quality of topical segmentation of documents.

Moreover, the post-processing formula does not necessarily need to be derived from the regularization criterion. You can do the opposite: transform the sequence of topical embeddings p_{tdw} into $\tilde{p}_{tdw}$ using a heuristical post-processing, for example, smoothing, sparsing, or segmentation. In fact, this corresponds to some regularization criterion $R(\Pi)$, that does not have to be written explicitly.

Theorem 7 *Let vectors $(\tilde{p}^k_{tdw})_{t\in T}$ satisfying the normalization condition $\sum_t \tilde{p}^k_{tdw} = 1$ be substituted in the M-step formulas instead of topical embeddings $(p^k_{tdw})_{t\in T}$ for each (d, w): $n_{dw} > 0$ at the k-th iteration of the EM-algorithm. Then such a transformation is equivalent to adding a smoothing–sparsing regularizer:*

$$R(\Pi) = \sum_{d\in D}\sum_{w\in d} n_{dw} \sum_{t\in T}(\tilde{p}^k_{tdw} - p^k_{tdw}) \ln p_{tdw}. \tag{31}$$

The proof immediately follows from substitution of (31) into (22).

One-pass topic modeling. In the EM-algorithm, the computation of document topical embedding $\theta_d = (\theta_{td})_{t\in T}$ requires many iterations over the document. Nevertheless, θ_d can be calculated in a one linear pass through the document [28]. The explicit formula $\theta_{td}(\Phi)$ follows from the M-step equation or from the total probability formula, where the distribution $p(t)$ is assumed to be fixed:

$$\theta_{td}(\Phi) = \sum_{w\in d} p(t|w)\, p(w|d) = \sum_{w\in d} \frac{n_{dw}}{n_d} \operatorname*{norm}_{t\in T}\bigl(\varphi_{wt}\, p(t)\bigr).$$

Although formally this equality constraint is not an optimization criterion, in fact it plays the role of a regularizer and can be used in combination with other regularizers within the ARTM framework.

Theorem 8 *Let the functions $\theta_{td}(\Phi)$ and $R(\Phi, \Theta)$ be continuously differentiable. Then the point Φ of the local extremum of the problem (6), (5) with equality constraints $\theta_{td} = \theta_{td}(\Phi)$ satisfies the system of equations with auxiliary variables $p_{tdw} = p(t \,|\, d, w)$, n_{td}, and p'_{tdw}, if zero columns of the matrices Φ, Θ are excluded from the solution:*

$$\begin{aligned}
p_{tdw} &= \underset{t\in T}{\operatorname{norm}}\bigl(\varphi_{wt}\theta_{td}\bigr);\\
n_{td} &= \sum_{w\in d} n_{dw} p_{tdw} + \theta_{td}\frac{\partial R}{\partial \theta_{td}};\\
p'_{tdw} &= p_{tdw} + \frac{\varphi_{wt}}{n_{dw}} \sum_{s\in T} \frac{n_{sd}}{\theta_{sd}} \frac{\partial \theta_{sd}}{\partial \varphi_{wt}};\\
\varphi_{wt} &= \underset{w\in W}{\operatorname{norm}}\biggl(\sum_{d\in D} n_{dw} p'_{tdw} + \varphi_{wt}\frac{\partial R}{\partial \varphi_{wt}}\biggr).
\end{aligned}$$

Like the E-step post-processing, modification of the EM-algorithm leads to the transformation of the topical embeddings p_{tdw} into p'_{tdw}, which are substituted into the usual M-step equation for the Φ matrix, without affecting the implementation of the remaining computations.

Experiments on three text collections [28] have shown that the one-pass algorithm is not only much faster but also improves the model in terms of sparseness, difference, logLift and coherence topic quality measures. The `BigARTM` and TopicNet libraries were used for the experiments.

The one-pass topic modeling opens up possibilities for fast computation of local contextual topical embeddings and processing of text as a sequence of words beyond the bag-of-words restrictive assumption.

10 Discussion and Conclusions

Hundreds of Bayesian topic models described in thousands of papers over the past two decades, can be reformulated in terms of classical non-Bayesian regularization. After this, they can be inferred easily, literally by one line of calculations out of the theorem on the maximization of a smooth function on unit simplices. One may wonder why this opportunity has not been noticed over so long time, especially given that Bayesian inference is laborious and unique to each model, which brings many technical inconveniences to researchers.

Many areas of data analysis and machine learning including image and signal processing are being developed according to the same general scenario. First, the formal model and the optimization problem are stated; then various specific structures, auxiliary criteria and regularizers are added; and finally, the transition to Bayesian regularization takes place. This transition usually occurs when there is a practical need for evaluation not only the model parameters themselves, but also their posterior distributions.

In Probabilistic Topic Modeling, the typical development scenario was violated and the community moved to Bayesian learning skipping the natural stage of development within the classical regularization. The very paradox is that in the practice of topic modeling, posterior distributions are used only for maximum likelihood point estimation.

Additive regularization (ARTM) is an attempt to fill the gap, though it might be late as the focus of community interest has already shifted to deep neural networks, attention models, and transformer architectures. Topic modeling is now focused more on the fusion with neural networks in search of opportunities to combine the best of two worlds [64].

Both worlds of models, neural-based and topic-based, generate vector representations of words and texts.

Both worlds tend to models homogenization [9], that is, to have a unified vector space that embeds any heterogeneous objects of any nature based on data about their interactions. It was demonstrated above how the hypergraph topic models implement this idea.

Both worlds of models can generate global and local embeddings. It has been shown above how the topic models can process a sequential text. The neural network models are much more complicated, their embeddings are able to absorb all the information about the connections between words, but it is out of our understanding which connections and how exactly are taken into account. Topic models are much simpler, their embeddings take into account only the lexical co-occurrence of words, while retaining interpretability. The coordinate-wise interpretability is a direct consequence of the fact that topic embeddings are non-negative normalized vectors on a unit simplex.

Avoiding the Bayesian inference makes topic models closer to neural models, thus making their deeper integration possible. As soon as non-negativity and normalization constraints are imposed, any vector parameter of a neural network can be learned with the use of the multiplicative gradient steps from the theorem of maximization on unit simplices. These are the promising opportunities for future research.

This work was supported by the Russian Foundation for Basic Research (project no. 20-07-00936).

References

1. Apishev, M.A., Vorontsov, K.V.: Learning topic models with arbitrary loss. In: Proceeding of the 26th Conference of FRUCT (Finnish-Russian University Cooperation in Telecommunications) Association, pp. 30–37 (2020)
2. Apishev, M., Koltcov, S. Koltsova, O., Nikolenko, S., Vorontsov, K.: Additive regularization for topic modeling in sociological studies of user-generated text content. In: MICAI 2016, 15th Mexican International Conference on Artificial Intelligence, Springer, Lecture Notes in Artificial Intelligence, vol. 10061, pp. 166–181 (2016)
3. Apishev, M., Koltcov, S., Koltsova, O., Nikolenko, S., Vorontsov, K.: Mining ethnic content online with additively regularized topic models. Comput. Sist. **20**(3), 387–403 (2016)
4. Balikas, G., Amini, M., Clausel, M.: On a topic model for sentences. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 921–924. ACM, New York, NY, SIGIR '16 (2016)
5. Belyy, A.V., Seleznova, M.S., Sholokhov, A.K., Vorontsov K.V.: Quality evaluation and improvement for hierarchical topic modeling. In: Computational Linguistics and Intellectual Technologies. Dialogue 2018, pp. 110–123 (2018)
6. Blei, D.M.: Probabilistic topic models. Commun. ACM **55**(4), 77–84 (2012)
7. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
8. Bodrunova, S., Koltsov, S., Koltsova, O., Nikolenko, S.I., Shimorina, A.: Interval semi-supervised LDA: classifying needles in a haystack. In: Espinoza, F.C., Gelbukh, A.F., Gonzalez-Mendoza, M. (eds.) MICAI (1), Springer, Lecture Notes in Computer Science, vol. 8265, pp. 265–274 (2013)
9. Bommasani, R., Hudson, D.A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M.S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N.S., Chen, A.S., Creel, K.A., Davis, J., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L.E., Goel, K., Goodman, N.D., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D.E., Hong, J., Hsu, K., Huang, J., Icard, T.F., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P.W., Krass, M.S., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X.L., Li, X., Ma, T., Malik, A., Manning, C.D., Mirchandani, S.P., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J.C., Nilforoshan, H., Nyarko, J.F., Ogut, G., Orr, L., Papadimitriou, I., Park, J.S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y.H., Ruiz, C., Ryan, J., R'e, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K.P., Tamkin, A., Taori, R., Thomas, A.W., Tramèr, F., Wang, R.E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S.M., Yasunaga, M., You, J., Zaharia, M.A., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., Liang, P.: On the opportunities and risks of foundation models. CoRR abs/2108.07258 (2021). https://crfm.stanford.edu/assets/report.pdf
10. Boyd-Graber, J., Hu, Y., Mimno, D.: Applications of topic models. Found. Trends® Inf. Retrieval **11**(2–3), 143–296 (2017)
11. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901. Curran Associates (2020)

12. Bulatov, V., Egorov, E., Veselova, E., Polyudova, D., Alekseev, V., Goncharov, A., Vorontsov, K.: TopicNet: making additive regularisation for topic modelling accessible. In: Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pp. 6745–6752 (2020)
13. Chen, B.: Word topic models for spoken document retrieval and transcription. ACM Trans. Asian Lang. Inf. Process. **8**(1), 2:1–2:27 (2009)
14. Chen, R., Hua, Q., Chang, Y.S., Wang, B., Zhang, L., Kong, X.: A survey of collaborative filtering-based recommender systems: from traditional methods to hybrid methods based on social networks. IEEE Access **6**, 64301–64320 (2018). https://doi.org/10.1109/ACCESS.2018.2877208
15. Chirkova, N.A., Vorontsov, K.V.: Additive regularization for hierarchical multimodal topic modeling. J. Mach. Learn. Data Anal. **2**(2), 187–200 (2016)
16. Churchill, R., Singh, L.: The evolution of topic modeling. ACM Comput. Surv. **54**(10s), 1 (2022)
17. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019)
18. Dudarenko, M.A.: Regularization of multilingual topic models. Vychisl Metody Programm (Numerical methods and programming) **16**, 26–38 (2015)
19. El-Kishky, A., Song, Y., Wang, C., Voss, C.R., Han, J.: Scalable topical phrase mining from text corpora. Proc. VLDB Endowment **8**(3), 305–316 (2014)
20. Feldman, D.G., Sadekova, T.R., Vorontsov, K.V.: Combining facts, semantic roles and sentiment lexicon in a generative model for opinion mining. In: Computational Linguistics and Intellectual Technologies. Dialogue 2020, pp. 268–283 (2020)
21. Frei, O., Apishev, M.: Parallel non-blocking deterministic algorithm for online topic modeling. In: AIST'2016, Analysis of Images, Social networks and Texts, Springer International Publishing Switzerland, Communications in Computer and Information Science (CCIS), vol. 661, pp. 132–144 (2016)
22. Harris, Z.: Distributional structure. Word **10**(23), 146–162 (1954)
23. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 50–57. ACM, New York, NY (1999)
24. Hospedales, T., Gong, S., Xiang, T.: Video behaviour mining using a dynamic topic model. Int. J. Comput. Vision **98**(3), 303–323 (2012)
25. Ianina, A., Vorontsov, K.: Regularized multimodal hierarchical topic model for document-by-document exploratory search. In: Balandin, S., Niemi, V., Tutina, T. (eds.) Proceeding of the 25th Conference of FRUCT (Finnish-Russian University Cooperation in Telecommunications) Association. The Seminar on Intelligence, Social Media and Web (ISMW). Helsinki, Finland, November 5–8, 2019, pp. 131–138 (2019)
26. Ianina, A.O., Vorontsov, K.V.: Hierarchical interpretable topical embeddings for exploratory search and real-time document tracking. Int. J. Embedded Real-Time Commun. Syst. (IJERTCS) **11**(4), 134 (2020)
27. Irkhin, I.A., Vorontsov K.V.: Convergence of the algorithm of additive regularization of topic models. Trudy Instituta Matematiki i Mekhaniki UrO RAN **26**(3), 56–68 (2020)
28. Irkhin, I.A., Bulatov, V.G., Vorontsov, K.V.: Additive regularization of topic models with fast text vectorization. Comput. Res. Model. **12**(6), 1515–1528 (2020)
29. Jameel, S., Lam, W. An N-gram topic model for time-stamped documents. In: 35th European Conference on Information Retrieval, ECIR-2013, Moscow, Russia, 24–27 March 2013, Lecture Notes in Computer Science (LNCS), pp. 292–304. Springer Verlag-Germany (2013)
30. Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., Zhao, L.: Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey. Multimedia Tools Appl. **78**(11), 15169–15211 (2019)

31. Khodorchenko, M., Teryoshkin, S., Sokhin, T., Butakov, N.: Optimization of learning strategies for artm-based topic models. In: de la Cal, E.A., Villar Flecha, J.R., Quintián, H., Corchado, E. (eds.) Hybrid Artificial Intelligent Systems, pp. 284–296. Springer International Publishing (2020)
32. Khodorchenko, M., Butakov, N., Sokhin, T., Teryoshkin, S.: Surrogate-based optimization of learning strategies for additively regularized topic models. Logic J. IGPL (2022). https://doi.org/10.1093/jigpal/jzac019, https://academic.oup.com/jigpal/advance-article-pdf/doi/10.1093/jigpal/jzac019/43022305/jzac019.pdf
33. Kochedykov, D.A., Apishev, M.A., Golitsyn, L.V., Vorontsov K.V.: Fast and modular regularized topic modelling. In: Proceeding of the 21st Conference of FRUCT (Finnish-Russian University Cooperation in Telecommunications) Association. The Seminar on Intelligence, Social Media and Web (ISMW). Helsinki, Finland, November 6–10, 2017, pp. 182–193. IEEE (2017)
34. Koltcov, S., Koltsova, O., Nikolenko, S.: Latent Dirichlet allocation: stability and applications to studies of user-generated content. In: Proceedings of the 2014 ACM Conference on Web Science, pp. 161–165. ACM, New York, NY, WebSci'14 (2014)
35. Li, S., Li, J., Pan R.: Tag-weighted topic model for mining semi-structured documents. In: IJCAI'13 Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, pp. 2855–2861. AAAI Press (2013)
36. M A Basher, A.R., Fung, B.C.M.: Analyzing topics and authors in chat logs for crime investigation. Knowl. Inf. Syst. **39**(2), 351–381 (2014)
37. Mei, Q., Shen, X., Zhai, C.: Automatic labeling of multinomial topic models. In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 490–499. Association for Computing Machinery, New York, NY (2007)
38. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013)
39. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. CoRR abs/1310.4546 (2013)
40. Nikolenko, S.I., Koltcov, S., Koltsova, O.: Topic modelling for qualitative studies. J. Inf. Sci. **43**(1), 88–102 (2017)
41. Paul, M.J., Dredze, M.: Drug extraction from the web: Summarizing drug experiences with multi-dimensional topic models. In: Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9–14, 2013. Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA, pp. 168–178 (2013)
42. Paul, M.J., Dredze, M.: Discovering health topics in social media using topic models. PLoS One **9**(8), e103408 (2014)
43. Popov, A., Bulatov, V., Polyudova, D., Veselova, E.: Unsupervised dialogue intent detection via hierarchical topic model. In: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), pp. 932–938. INCOMA Ltd., Varna (2019)
44. Potapenko, A., Popov, A., Vorontsov, K.: Interpretable probabilistic embeddings: bridging the gap between topic models and neural networks. In: Communications in Computer and Information Science, vol. 789. AINL-6: Artificial Intelligence and Natural Language Conference, St. Petersburg, Russia, September 20–23, 2017, pp. 167–180. Springer, Cham (2017)
45. Rosen-Zvi, M., Griffiths, T., Steyvers, M., Smyth, P.: The author-topic model for authors and documents. In: Proceedings of the 20th conference on Uncertainty in artificial intelligence, pp. 487–494. AUAI Press, Arlington, Virginia, UAI '04 (2004)
46. Rubin, T.N., Chambers, A., Smyth, P., Steyvers, M.: Statistical topic models for multi-label document classification. Mach. Learn. **88**(1–2), 157–208 (2012)
47. Sharma, A., Pawar, D.M.: Survey paper on topic modeling techniques to gain useful forecasting information on violent extremist activities over cyber space. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **5**(12), 429–436 (2015)
48. Skachkov, N.A., Vorontsov, K.V.: Improving topic models with segmental structure of texts. In: Computational Linguistics and Intellectual Technologies. Dialogue 2018, pp. 652–661 (2018)

49. Tan, Y., Ou, Z.: Topic-weak-correlated latent Dirichlet allocation. In: 7th International Symposium Chinese Spoken Language Processing (ISCSLP), pp. 224–228 (2010)
50. Tikhonov, A.N., Arsenin, V.Y.: Solution of ill-posed problems. W. H. Winston, Washington, DC (1977)
51. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Lu., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 5998–6008. Curran Associates (2017)
52. Vinokourov, A., Girolami, M.: A probabilistic hierarchical clustering method for organising collections of text documents. In: Proceedings 15th International Conference on Pattern Recognition. ICPR-2000, vol. 2, pp. 182–185 (2000). https://doi.org/10.1109/ICPR.2000.906043
53. Vorontsov, K.V.: Additive regularization for topic models of text collections. Doklady Math. **89**(3), 301–304 (2014)
54. Vorontsov, K.V., Potapenko, A.A.: Tutorial on probabilistic topic modeling: Additive regularization for stochastic matrix factorization. In: AIST'2014, Analysis of Images, Social Networks and Texts, Communications in Computer and Information Science (CCIS), vol. 436, pp. 29–46. Springer International Publishing Switzerland (2014)
55. Vorontsov, K.V., Potapenko, A.A.: Additive regularization of topic models. Mach. Learn. Special Issue on Data Analysis and Intelligent Optimization with Applications **101**(1), 303–323 (2015)
56. Vorontsov, K., Frei, O., Apishev, M., Romov, P., Suvorova, M., Yanina, A: Non-bayesian additive regularization for multimodal topic modeling of large collections. In: Proceedings of the 2015 Workshop on Topic Models: Post-processing and Applications, pp. 29–37. ACM, New York, NY (2015)
57. Vorontsov, K.V., Frei, O.I., Apishev, M.A., Romov, P.A., Suvorova, M.A.: BigARTM: open source library for regularized multimodal topic modeling of large collections. In: AIST'2015, Analysis of Images, Social Networks and Texts, Communications in Computer and Information Science (CCIS), pp. 370–384. Springer International Publishing Switzerland (2015)
58. Vulic, I., De Smet, W., Tang, J., Moens, M.F.: Probabilistic topic modeling in multilingual settings: an overview of its methodology and applications. Inf. Process. Manag. **51**(1), 111–147 (2015)
59. Wallach, H.M.: Topic modeling: beyond bag-of-words. In: Proceedings of the 23rd International Conference on Machine Learning. ACM, New York, NY, ICML '06, pp. 977–984 (2006)
60. Wang, X., McCallum, A., Wei, X.: Topical n-grams: phrase and topic discovery, with an application to information retrieval. In: Proceedings of the 2007 Seventh IEEE International Conference on Data Mining, pp. 697–702. IEEE Computer Society, Washington, DC (2007)
61. Yanina, A., Golitsyn, L., Vorontsov, K.: Multi-objective topic modeling for exploratory search in tech news. In: Filchenkov, A., Pivovarova, L., Žižka, J. (eds.) Communications in Computer and Information Science, vol 789. AINL-6: Artificial Intelligence and Natural Language Conference, St. Petersburg, Russia, September 20–23, 2017, pp. 181–193. Springer International Publishing, Cham (2018)
62. Zavitsanos, E., Paliouras, G., Vouros, G.A.: Non-parametric estimation of topic hierarchies from texts with hierarchical Dirichlet processes. J. Mach. Learn. Res. **12**, 2749–2775 (2011)
63. Zhao, W.X., Jiang, J., Weng, J., He, J., Lim, E.P., Yan, H., Li, X.: Comparing Twitter and traditional media using topic models. In: Proceedings of the 33rd European Conference on Advances in Information Retrieval, pp. 338–349. Springer-Verlag, Berlin, Heidelberg, ECIR'11 (2011)
64. Zhao, H., Phung, D., Huynh, V., Jin, Y., Du, L., Buntine, W.: Topic modelling meets deep neural networks: a survey. In: Zhou, Z.H. (ed.) Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, pp. 4713–4720 (2021)
65. Zuo, Y., Zhao, J., Xu, K.: Word network topic model: a simple but general solution for short and imbalanced texts. Knowl. Inf. Syst. **48**(2), 379–398 (2016)