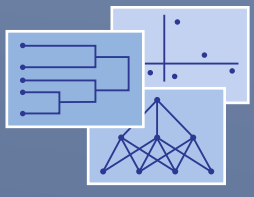
Studies in Classification, Data Analysis, and Knowledge Organization

Paula Brito · José G. Dias · Berthold Lausen · Angela Montanari · Rebecca Nugent *Editors*

Classification and Data Science in the Digital Age



OPEN ACCESS

Springer

Studies in Classification, Data Analysis, and Knowledge Organization

Studies in Classification, Data Analysis, and Knowledge Organization is a book series which offers constant and up-to-date information on the most recent developments and methods in the fields of statistical data analysis, exploratory statistics, classification and clustering, handling of information and ordering of knowledge. It covers a broad scope of theoretical, methodological as well as application-oriented articles, surveys and discussions from an international authorship and includes fields like computational statistics, pattern recognition, biological taxonomy, DNA and genome analysis, marketing, finance and other areas in economics, databases and the internet. A major purpose is to show the intimate interplay between various, seemingly unrelated domains and to foster the cooperation between mathematicians, statisticians, computer scientists and practitioners by offering well-based and innovative solutions to urgent problems of practice.

Paula Brito · José G. Dias · Berthold Lausen ·
Angela Montanari · Rebecca Nugent
Editors

Classification and Data Science in the Digital Age

Springer

Editors
Paula Brito
Faculty of Economics
University of Porto
Porto, Portugal

INESC TEC, Centre for Artificial Intelligence and Decision Support (LIAAD)
Porto, Portugal

Berthold Lausen
Department of Mathematical Sciences
University of Essex
Colchester, UK

Rebecca Nugent
Department of Statistics & Data Science
Carnegie Mellon University
Pittsburgh, PA, USA

José G. Dias
Business Research Unit
University Institute of Lisbon
Lisbon, Portugal

Angela Montanari
Department of Statistical Sciences
"Paolo Fortunati"
University of Bologna
Bologna, Italy

ISSN 1431-8814 ISSN 2198-3321 (electronic)
Studies in Classification, Data Analysis, and Knowledge Organization
ISBN 978-3-031-09033-2 ISBN 978-3-031-09034-9 (eBook)
https://doi.org/10.1007/978-3-031-09034-9

Mathematics Subject Classification: 62H30, 62H25, 62R07, 68T09, 62H86, 68T10, 94A16, 68T30

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

Preface

"Classification and Data Science in the Digital Age", the 17th Conference of the International Federation of Classification Societies (IFCS), is held in Porto, Portugal, from July 19th to July 23rd 2022, locally organised by the Faculty of Economics of the University of Porto and the Portuguese Association for Classification and Data Analysis, CLAD.

The International Federation of Classification Societies (IFCS), founded in 1985, is an international scientific organization with non-profit and non-political motives. Its purpose is to promote mutual communication, co-operation and interchange of views among all those interested in scientific principles, numerical methods, theory and practice of data science, data analysis, and classification in a broad sense and in as wide a range of applications as possible; to serve as an agency for the dissemination of scientific information related to these areas of interest; to prepare international conferences; to publish a newsletter and other publications. The scientific activities of the Federation are intended for all people interested in theory of classification and data analysis, and related methods and applications. IFCS 2022 – originally scheduled for August 2021, and postponed due to the Covid-19 pandemic – will be its 17th edition; previous editions were held in Thessaloniki (2019), Tokyo (2017) and Bologna (2015).

Keynote lectures are addressed by Genevera Allen (Rice University, USA), Charles Bouveyron (Université Côte d'Azur, Nice, France), Dianne Cook (Monash University, Melbourne, Australia), and João Gama (Faculty of Economics, University of Porto & LIAAD INESC TEC, Portugal). The conference program includes two tutorials: "Analysis of Data Streams" by João Gama (Faculty of Economics, University of Porto & LIAAD INESC TEC, Portugal) and "Categorical Data Analysis of Visualization" by Rosaria Lombardo (Università degli Studi della Campania Luigi Vanvitelli, Italy) and Eric Beh (University of Newcastle, Australia). IFCS 2022 has highlighted topics, which lead to Semi-Plenary Invited Sessions. The Conference program also includes Thematic Tracks on specific areas, as well as free contributed sessions in different topics (both oral communications and posters).

The Conference Scientific Program Committee is co-chaired by Paula Brito, José G. Dias, Berthold Lausen, and Angela Montanari, and includes representatives of the IFCS member societies: Adalbert Wilhelm – GfKl, Ahmed Moussa – MCS, Arthur White – IPRCS, Brian Franczak – CS, Eva Boj del Val – SEIO, Fionn Murtagh – BCS, Francesco Mola – CLADAG, Hyunjoong Kim – KCS, Javier Trejos Zelaya – SoCCCAD, Koji Kurihara – JCS, Krzysztof Jajuga – SKAD, Mark de Rooij – VOC, Mohamed Nadif – SFC, Niel le Roux – MDAG, Simona Korenjak Černe – SSS, Theodore Chadjipadelis – GSDA, who were responsible for the Conference Scientific Program, and whom the organisers wish to thank for their precious cooperation. Special thanks are also due to the chairs of the Thematic Tracks, for their invaluable collaboration.

The papers included in this volume present new developments in relevant topics of Data Science and Classification, constituting a valuable collection of methodological and applied papers that represent the current research in highly developing areas. Combining new methodological advances with a wide variety of real applications, this volume is certainly of great value for Data Science researchers and practitioners alike.

First of all, the organisers of the Conference and the editors would like to thank all authors, for their cooperation and commitment. We are specially grateful to all colleagues who served as reviewers, and whose work was decisive to the scientific quality of these proceedings. We also thank all those who have contributed to the design and production of this Book of Proceedings at Springer, in particular Veronika Rosteck, for her help concerning all aspects of publication.

The organisers would like to express their gratitude to the Portuguese Association for Classification and Data Analysis, CLAD, as well as to the Faculty of Economics of the University of Porto (FEP–UP), who enthusiastically supported the Conference from the very start, and contributed to its success. We cordially thank all members of the Local Organising Committee – Adelaide Figueiredo, Carlos Ferreira, Carlos Marcelo, Conceição Rocha, Fernanda Figueiredo, Fernanda Sousa, Jorge Pereira, M. Eduarda Silva, Paulo Teles, Pedro Campos, Pedro Duarte Silva, and Sónia Dias – and all people at FEP–UP who worked actively for the conference organisation, and whose work is much appreciated. We are very grateful to all our sponsors, for their generous support. Finally, we thank all authors and participants, who made the conference possible.

Porto,
July 2022

Paula Brito
José G. Dias
Berthold Lausen
Angela Montanari
Rebecca Nugent

Acknowledgements

The Editors are extremely grateful to the reviewers, whose work was determinant for the scientific quality of these proceedings. They were, in alphabetical order:

Adalbert Wilhelm
Agustín Mayo-Iscar
Alípio Jorge
André C. P. L. F. de Carvalho
Ann Maharaj
Anuška Ferligoj
Arthur White
Berthold Lausen
Brian Franczak
Carlos Soares
Christian Hennig
Conceição Amado
Eva Boj del Val
Francesco Mola
Francisco de Carvalho
Geoff McLAchlan
Gilbert Saporta
Glòria Mateu-Figueras
Hans Kestler
Hélder Oliveira
Hyunjoong Kim
Jaime Cardoso
Javier Trejos
Jean Diatta
José A. Lozano
José A. Vilar
José Matos
Koji Kurihara
Krzysztof Jajuga
Laura Palagi
Laura Sangalli
Lazhar Labiod
Luis Angel García-Escudero
Luis Teixeira
M. Rosário Oliveira
Margarida G. M. S. Cardoso
Mark de Rooij
Michelangelo Ceci
Mohamed Nadif
Niel Le Roux
Paolo Mignone
Patrice Bertrand
Pedro Campos
Pedro Duarte Silva
Pedro Ribeiro
Peter Filzmoser
Rosanna Verde
Rosaria Lombardo
Salvatore Ingrassia
Satish Singh
Simona Korenjak-Černe
Theodore Chadjipadelis
Veronica Piccialli
Vladimir Batagelj

Partners & Sponsors

We are extremely grateful to the following institutions whose support contributes to the success of IFCS 2022:

Sponsors

Banco de Portugal

Berd

Comissão de Viticultura da Região dos Vinhos Verdes

Indie Campers

INESC/TEC

Luso-American Development Foundation

PSE

Sociedade Portuguesa de Estatística

Instituto Nacional de Estatística/Statistics Portugal

Unilabs

Universidade do Porto

Partners

Associação Portuguesa para a Investigação Operacional

Associação Portuguesa de Reconhecimento de Padrões

Associação de Turismo do Porto e Norte

Centro Internacional de Matemática

Faculdade de Engenharia da Universidade do Porto

International Association of Statistical Computing

International Association of Statistical Education

Sociedade Portuguesa de Matemática

Springer

Organisation

CLAD - Associação Portuguesa de Classificação e Análise de Dados

Faculdade de Economia da Universidade do Porto

Contents

A Topological Clustering of Individuals 1
Rafik Abdesselam

Model Based Clustering of Functional Data with Mild Outliers 11
Cristina Anton and Iain Smith

A Trivariate Geometric Classification of Decision Boundaries for Mixtures of Regressions 21
Filippo Antonazzo and Salvatore Ingrassia

Generalized Spatio-temporal Regression with PDE Penalization 29
Eleonora Arnone, Elia Cunial, and Laura M. Sangalli

A New Regression Model for the Analysis of Microbiome Data 35
Roberto Ascari and Sonia Migliorati

Stability of Mixed-type Cluster Partitions for Determination of the Number of Clusters 43
Rabea Aschenbruck, Gero Szepannek, and Adalbert F. X. Wilhelm

A Review on Official Survey Item Classification for Mixed-Mode Effects Adjustment 53
Afshin Ashofteh and Pedro Campos

Clustering and Blockmodeling Temporal Networks – Two Indirect Approaches 63
Vladimir Batagelj

Latent Block Regression Model 73
Rafika Boutalbi, Lazhar Labiod, and Mohamed Nadif

Using Clustering and Machine Learning Methods to Provide Intelligent Grocery Shopping Recommendations 83
Nail Chabane, Mohamed Achraf Bouaoune, Reda Amir Sofiane Tighilt, Bogdan Mazoure, Nadia Tahiri, and Vladimir Makarenkov

COVID-19 Pandemic: a Methodological Model for the Analysis of Government's Preventing Measures and Health Data Records 93
Theodore Chadjipadelis and Sofia Magopoulou

pcTVI: Parallel MDP Solver Using a Decomposition into Independent Chains 101
Jaël Champagne Gareau, Éric Beaudry, and Vladimir Makarenkov

Three-way Spectral Clustering 111
Cinzia Di Nuzzo and Salvatore Ingrassia

Improving Classification of Documents by Semi-supervised Clustering in a Semantic Space 121
Jasminka Dobša and Henk A. L. Kiers

Trends in Data Stream Mining 131
João Gama

Old and New Constraints in Model Based Clustering 139
Luis A. García-Escudero, Agustín Mayo-Iscar, Gianluca Morelli, and Marco Riani

Clustering Student Mobility Data in 3-way Networks 147
Vincenzo Giuseppe Genova, Giuseppe Giordano, Giancarlo Ragozini, and Maria Prosperina Vitale

Clustering Brain Connectomes Through a Density-peak Approach 155
Riccardo Giubilei

Similarity Forest for Time Series Classification 165
Tomasz Górecki, Maciej Łuczak, and Paweł Piasecki

Detection of the Biliary Atresia Using Deep Convolutional Neural Networks Based on Statistical Learning Weights via Optimal Similarity and Resampling Methods 175
Kuniyoshi Hayashi, Eri Hoshino, Mitsuyoshi Suzuki, Erika Nakanishi, Kotomi Sakai, and Masayuki Obatake

Some Issues in Robust Clustering 183
Christian Hennig

Robustness Aspects of Optimized Centroids 193
Jan Kalina and Patrik Janáček

Data Clustering and Representation Learning Based on Networked Data 203
Lazhar Labiod and Mohamed Nadif

Towards a Bi-stochastic Matrix Approximation of k-means and Some Variants 213
Lazhar Labiod and Mohamed Nadif

Clustering Adolescent Female Physical Activity Levels with an Infinite Mixture Model on Random Effects 223
Amy LaLonde, Tanzy Love, Deborah R. Young, and Tongtong Wu

Unsupervised Classification of Categorical Time Series Through Innovative Distances 233
Ángel López-Oriona, José A. Vilar, and Pierpaolo D'Urso

Fuzzy Clustering by Hyperbolic Smoothing 243
David Masís, Esteban Segura, Javier Trejos, and Adilson Xavier

Stochastic Collapsed Variational Inference for Structured Gaussian Process Regression Networks 253
Rui Meng, Herbert K. H. Lee, and Kristofer Bouchard

An Online Minorization-Maximization Algorithm 263
Hien Duy Nguyen, Florence Forbes, Gersende Fort, and Olivier Cappé

Detecting Differences in Italian Regional Health Services During Two Covid-19 Waves 273
Lucio Palazzo and Riccardo Ievoli

Political and Religion Attitudes in Greece: Behavioral Discourses 283
Georgia Panagiotidou and Theodore Chadjipadelis

Supervised Classification via Neural Networks for Replicated Point Patterns 293
Kateřina Pawlasová, Iva Karafiátová, and Jiří Dvořák

Parsimonious Mixtures of Seemingly Unrelated Contaminated Normal Regression Models 303
Gabriele Perrone and Gabriele Soffritti

Penalized Model-based Functional Clustering: a Regularization Approach via Shrinkage Methods 313
Nicola Pronello, Rosaria Ignaccolo, Luigi Ippoliti, and Sara Fontanella

Emotion Classification Based on Single Electrode Brain Data: Applications for Assistive Technology 323
Duarte Rodrigues, Luis Paulo Reis, and Brígida Mónica Faria

The Death Process in Italy Before and During the Covid-19 Pandemic: a Functional Compositional Approach 333
Riccardo Scimone, Alessandra Menafoglio, Laura M. Sangalli, and Piercesare Secchi

Clustering Validation in the Context of Hierarchical Cluster Analysis: an Empirical Study 343
Osvaldo Silva, Áurea Sousa, and Helena Bacelar-Nicolau

An MML Embedded Approach for Estimating the Number of Clusters . . 353
Cláudia Silvestre, Margarida G. M. S. Cardoso, and Mário Figueiredo

Typology of Motivation Factors for Employees in the Banking Sector: An Empirical Study Using Multivariate Data Analysis Methods 363
Áurea Sousa, Osvaldo Silva, M. Graça Batista, Sara Cabral, and Helena Bacelar-Nicolau

A Proposal for Formalization and Definition of Anomalies in Dynamical Systems 373
Jan Michael Spoor, Jens Weber, and Jivka Ovtcharova

New Metrics for Classifying Phylogenetic Trees Using K-means and the Symmetric Difference Metric 383
Nadia Tahiri and Aleksandr Koshkarov

On Parsimonious Modelling via Matrix-variate t Mixtures 393
Salvatore D. Tomarchio

Evolution of Media Coverage on Climate Change and Environmental Awareness: an Analysis of Tweets from UK and US Newspapers 403
Gianpaolo Zammarchi, Maurizio Romano, and Claudio Conversano

A Topological Clustering of Individuals

Rafik Abdesselam

Abstract The clustering of objects-individuals is one of the most widely used approaches to exploring multidimensional data. The two common unsupervised clustering strategies are Hierarchical Ascending Clustering (HAC) and k-means partitioning used to identify groups of similar objects in a dataset to divide it into homogeneous groups. The proposed Topological Clustering of Individuals, or TCI, studies a homogeneous set of individual rows of a data table, based on the notion of neighborhood graphs; the columns-variables are more-or-less correlated or linked according to whether the variable is of a quantitative or qualitative type. It enables topological analysis of the clustering of individual variables which can be quantitative, qualitative or a mixture of the two. It first analyzes the correlations or associations observed between the variables in a topological context of principal component analysis (PCA) or multiple correspondence analysis (MCA), depending on the type of variable, then classifies individuals into homogeneous group, relative to the structure of the variables considered. The proposed TCI method is presented and illustrated here using a real dataset with quantitative variables, but it can also be applied with qualitative or mixed variables.

Keywords: hierarchical clustering, proximity measure, neighborhood graph, adjacency matrix, multivariate data analysis

1 Introduction

The objective of this article is to propose a topological method of data analysis in the context of clustering. The proposed approach, Topological Clustering of Individuals

Rafik Abdesselam (✉)
University of Lyon, Lyon 2, ERIC - COACTIS Laboratories
Department of Economics and Management, 69365 Lyon, France,
e-mail: rafik.abdesselam@univ-lyon2.fr

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_1

(TCI) is different from those that already exist and with which it is compared. There are approaches specifically devoted to the clustering of individuals, for example, the Cluster procedure implemented in SAS software, but as far as we know, none of these approaches has been proposed in a topological context.

Proximity measures play an important role in many areas of data analysis [16, 5, 9]. The results of any operation involving structuring, clustering or classifying objects are strongly dependent on the proximity measure chosen.

This study proposes a method for the topological clustering of individuals whatever type of variable is being considered: quantitative, qualitative or a mixture of both. The eventual associations or correlations between the variables partly depends on the database being used and the results can change according to the selected proximity measure. A proximity measure is a function which measures the similarity or dissimilarity between two objects or variables within a set.

Several topological data analysis studies have been proposed both in the context of factorial analyses (discriminant analysis [4], simple and multiple correspondence analyses [3], principal component analysis [2]) and in the context of clustering of variables [1], clustering of individuals [10] and this proposed TCI approach.

This paper is organized as follows. In Section 2, we briefly recall the basic notion of neighborhood graphs, we define and show how to construct an adjacency matrix associated with a proximity measure within the framework of the analysis of the correlation structure of a set of quantitative variables, and we present the principles of TCI according to continuous data. This is illustrated in Section 3 using an example based on real data. The TCI results are compared with those of the well-known classical clustering of individuals. Finally, Section 4 presents the concluding remarks on this work.

2 Topological Context

Topological data analysis is an approach based on the concept of the neighborhood graph. The basic idea is actually quite simple: for a given proximity measure for continuous or binary data and for a chosen topological structure, we can match a topological graph induced on the set of objects.

In the case of continuous data, we consider $E = \{x^1, \cdots, x^j, \cdots, x^p\}$, a set of p quantitative variables. We can see in [1] cases of qualitative or even mixed variables.

We can, by means of a proximity measure u, define a neighborhood relationship, V_u, to be a binary relationship based on $E \times E$. There are many possibilities for building this neighborhood binary relationship.

Thus, for a given proximity measure u, we can build a neighborhood graph on E, where the vertices are the variables and the edges are defined by a property of the neighborhood relationship.

Many definitions are possible to build this binary neighborhood relationship. One can choose the Minimal Spanning Tree (MST) [7], the Gabriel Graph (GG) [11] or, as is the case here, the Relative Neighborhood Graph (RNG) [14].

For any given proximity measure u, we can construct the associated adjacency binary symmetric matrix V_u of order p, where, all pairs of neighboring variables in E satisfy the following RNG property:

$$V_u(x^k\ ,\ x^l) = \begin{cases} 1 & \text{if } u(x^k\ ,\ x^l) \ \leq \ \max[u(x^k\ ,\ x^t), u(x^t\ ,\ x^l)]\ ; \\ & \qquad\qquad \forall x^k,\ x^l,\ x^t \in E,\ x^t \neq x^k \ \ and \ \ x^t \neq x^l \\ 0 & \text{otherwise.} \end{cases}$$

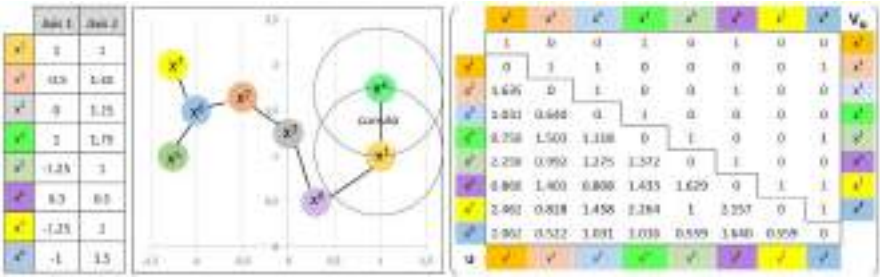


Fig. 1 Data - RNG structure - Euclidean distance - Associated adjacency matrix.

Figure 1 shows a simple illustrative example in $\mathbb{R}^2$ of a set of quantitative variables that verify the structure of the RNG graph with Euclidean distance as proximity measure: $u(x^k\ ,\ x^l) = \sqrt{\sum_{j=1}^{2}(x_j^k - x_j^l)^2}$.

This generates a topological structure based on the objects in E which are completely described by the adjacency binary matrix V_u.

2.1 Reference Adjacency Matrices

Three topological factorial approaches are described in [1] according to the type of variables considered: quantitative, qualitative or a mixture of both. We consider here the case of a set of quantitative variables.

We assume that we have at our disposal a set $E = \{x^j; j = 1, \cdots, p\}$ of p quantitative variables and n individuals-objects. The objective here is to analyze in a topological way, the structure of the correlations of the variables considered [2], from which the clustering of individuals will then be established.

We construct the reference adjacency matrix named $V_{u_\star}$ from the correlation matrix. Expressions of suitable adjacency reference matrices for cases involving qualitative variables or mixed variables are given in [1].

To examine the correlation structure between the variables, we look at the significance of their linear correlation. The reference adjacency matrix $V_{u_\star}$ associated with reference measure $u_\star$, can be written using the Student's t-test of the linear correlation coefficient ρ of Bravais-Pearson:

Definition 1 For quantitative variables, $V_{u_\star}$ is defined as:

$$V_{u_\star}(x^k, x^l) = \begin{cases} 1 \text{ if } \quad p\text{-value } = P[\ |\ T_{n-2}\ | > \text{t-value }\ ] \leq \alpha\ ;\ \forall k, l = 1, p \\ 0 \text{ otherwise.} \end{cases}$$

where the p-value is the significance test of the linear correlation coefficient for the two-sided test of the null and alternative hypotheses, $H_0 : \rho(x^k\ ,\ x^l) = 0$ vs. $H_1 : \rho(x^k\ ,\ x^l) \neq 0$.

Let T_{n-2} be a t-distributed random variable of Student with $\nu = n - 2$ degrees of freedom. In this case, the null hypothesis is rejected if the p-value is less than or equal to a chosen α significance level, for example, $\alpha = 5\%$. Using a linear correlation test, if the p-value is very small, it means that there is a very low likelihood that the null hypothesis is correct, and consequently we can reject it.

2.2 Topological Analysis - Selective Review

Whatever the type of variable set being considered, the built reference adjacency matrix $V_{u_\star}$ is associated with an unknown reference proximity measure $u_\star$.

The robustness depends on the α error risk chosen for the null hypothesis: no linear correlation in the case of quantitative variables, or positive deviation from independence in the case of qualitative variables, can be studied by setting a minimum threshold in order to analyze the sensitivity of the results. Certainly the numerical results will change, but probably not their interpretation.

We assume that we have at our disposal $\{x^k; k = 1, .., p\}$ a set of p homogeneous quantitative variables measured on n individuals. We will use the following notations:

- $X_{(n,p)}$ is the data matrix with n rows-individuals and p columns-variables,
- $V_{u_\star}$ is the symmetric adjacency matrix of order p, associated with the reference measure $u_\star$ which best structures the correlations of the variables,
- $\widehat{X}_{(n,p)} = XV_{u_\star}$ is the projected data matrix with n individuals and p variables,
- M_p is the matrix of distances of order p in the space of individuals,
- $D_n = \frac{1}{n} I_n$ is the diagonal matrix of weights of order n in the space of variables.

We first analyze, in a topological way, the correlation structure of the variables using a Topological PCA, which consists of carrying out the standardized PCA [6, 8] triplet $(\ \widehat{X}\ ,\ M_p\ ,\ D_n\ )$ of the projected data matrix $\widehat{X} = XV_{u_\star}$ and, for comparison, the duality diagram of the Classical standardized PCA triplet $(\ X\ ,\ M_p\ ,\ D_n\ )$ of the initial data matrix X. We then proceed with a clustering of individuals based on the significant principal components of the previous topological PCA.

Definition 2 TCI consist of performing a HAC, based on the Ward criterion[1] [15], on the significant factors of the standardized PCA of the triplet $(\widehat{X}, M_p, D_n)$.

[1] Aggregation based on the criterion of the loss of minimal inertia.

3 Illustrative Example

The data used [13] to illustrate the TCI approach describe the renewable electricity (RE) of the 13 French regions in 2017, described by 7 quantitative variables relating to RE. The growth of renewable energy in France is significant. Some French regions have expertise in this area; however, the regions' profiles appear to differ.

The objective is to specify regional disparities in terms of RE by applying topological clustering to the French regions in order to identify which were the country's greenest regions in 2017. Statistics relating to the variables are displayed in Table 1.

Table 1 Summary statistics of renewable energy variables.

Variable	Frequency	Mean	Standard Deviation (N)	Coefficient of variation (%)	Min	Max
Total RE production (TWH)	13	6.84	6.58	96.19	0.59	2.34
Total RE consumption (TWH)	13	3.70	1.87	50.67	2.18	7.06
Coverage RE consumption (%)	13	0.18	0.11	59.01	0.02	0.36
Hydroelectricity (%)	13	0.34	0.30	87.47	0.01	0.89
Solar electricity (%)	13	0.13	0.09	72.57	0.02	0.31
Wind electricity (%)	13	0.39	0.29	76.12	0.01	0.86
Biomass electricity (%)	13	0.15	0.19	130.54	0.01	0.79

Table 2 Correlation matrix (p-value) - Reference adjacency matrix $V_{u_\star}$.

Production	**1.000**					
Consumption	**0.575**	**1.000**				
	(0.040)					
Coverage	**0.798**	0.090	**1.000**			
	(0.001)	(0.771)				
Hydroelectricity	**0.720**	0.138	**0.872**	**1.000**		
	(0.006)	(0.653)	**(0.000)**			
Solar	-0.272	-0.477	0.105	0.168	**1.000**	
	(0.369)	(0.099)	(0.734)	(0.582)		
Wind	-0.408	-0.305	-0.524	**-0.772**	-0.395	**1.000**
	(0.167)	(0.311)	(0.066)	**(0.002)**	(0.181)	
Biomass	-0.365	0.489	**-0.609**	-0.459	-0.149	-0.135	**1.000**
	(0.220)	(0.090)	**(0.027)**	(0.114)	(0.627)	(0.660)	

Significance level: p−value $\leq \alpha = 5\%$

$$V_{u_\star} = \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & 0 & 0 & 0 \\ \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & 0 & 0 \\ \mathbf{1} & 0 & \mathbf{1} & \mathbf{1} & 0 & 0 & \mathbf{-1} \\ \mathbf{1} & 0 & \mathbf{1} & \mathbf{1} & 0 & \mathbf{-1} & 0 \\ 0 & 0 & 0 & 0 & \mathbf{1} & 0 & 0 \\ 0 & 0 & 0 & \mathbf{-1} & 0 & \mathbf{1} & 0 \\ 0 & 0 & \mathbf{-1} & 0 & 0 & 0 & \mathbf{1} \end{pmatrix}$$

The adjacency matrix $V_{u_\star}$, associated with the proximity measure $u_\star$, adapted to the data considered, is built from the correlations matrix Table 2 according to Definition 1. Note that in this case, which uses quantitative variables, it is considered that two positively correlated variables are related and that two negatively correlated variables are related but remote. We will therefore take into account any sign of correlation between variables in the adjacency matrix.

We first carry out a Topological PCA to identify the correlation structure of the variables. A HAC, according to Ward's criterion, is then applied to the significant principal components of the PCA of the projected data. We then compare the results of a topological and a classical PCA.

Figure 2 presents, for comparison on the first factorial plane, the correlations between principal components-factors and the original variables.

We can see that these correlations are slightly different, as are the percentages of the inertias explained on the first principal planes of Topological and Classic PCA.

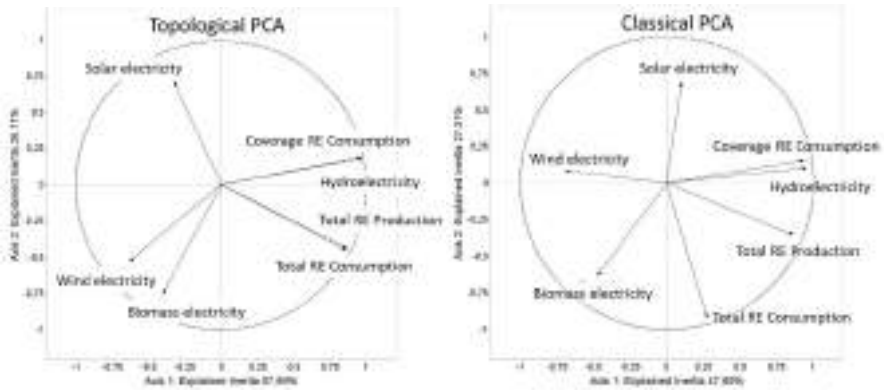


Fig. 2 Topological & Classical PCA of RE of the French regions.

The two first factors of the Topological PCA explain 57.89% and 26.11%, respectively, accounting for 83.99% of the total variation in the data set; however, the two first factors of the Classical PCA add up to 75.20%. Thus, the first two factors provide an adequate synthesis of the data, that is, of RE in the French regions. We restrict the comparison to the first significant factorial axes.

For comparison, Figure 3 shows dendrograms of the Topological and Classical clustering of the French regions according to their RE. Note that the partitions chosen in 5 clusters are appreciably different, as much by composition as by characterization. The percentage variance produced by the TCI approach, $R^2 = 86.42\%$, is higher than that of the classic approach, $R^2 = 84.15\%$, indicating that the clusters produced via the TCI approach are more homogeneous than those generated by the Classical one.

Based on the TCI analysis, the Corse region alone constitutes the fourth cluster, and the Nouvelle-Acquitaine region is found in the second cluster with the Grand-Est, Occitanie and Provence-Alpes-Côte-d'Azur (PACA) regions; however, in the Classical clustering, these two regions - Corse and Nouvelle-Aquitaine - together constitute the third cluster.

Figure 4 summarizes the significant profiles (+) and anti-profiles (-) of the two typologies; with a risk of error less than or equal to 5%, they are quite different.

The first cluster produced via the TCI approach, consisting of a single region, Auvergne-Rhônes-Alpes (AURA), is characterized by high share of hydroelectricity, a high level of coverage of regional consumption, and high RE production and consumption. The second cluster - which groups together the four regions of Grand-Est, Occitanie, Provence-Alpes-Côte-d'Azur (PACA) and Nouvelle-Aquitaine - is considered a homogeneous cluster, which means that none of the seven RE characteristics differ significantly from the average of these characteristics across all regions. This cluster can therefore be considered to reflect the typical picture of RE in France.

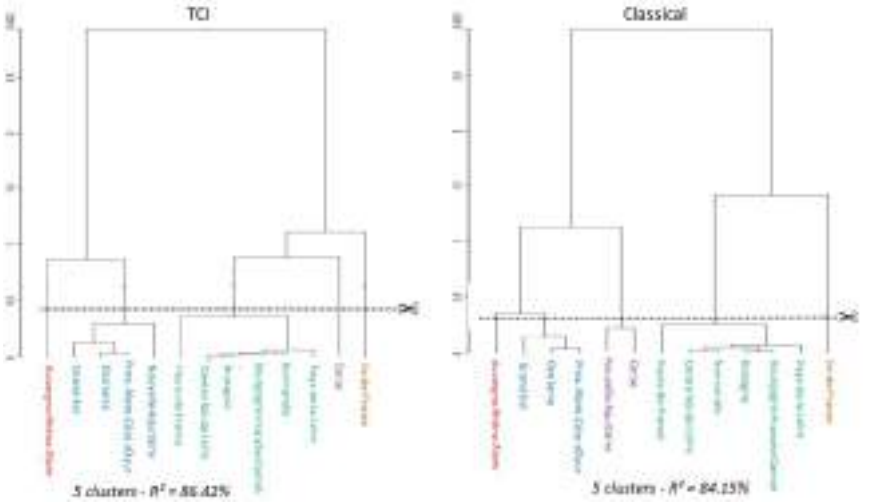


Fig. 3 Topological and Classical dendrograms of the French regions.

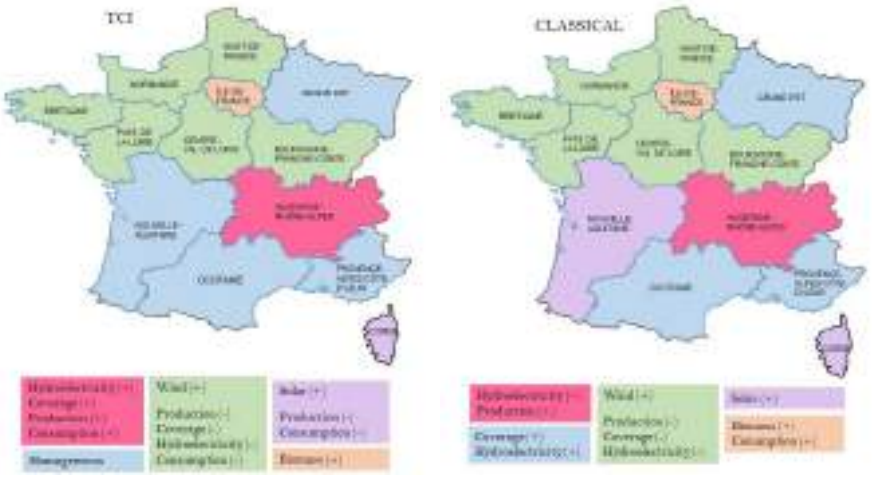


Fig. 4 Typologies - Characterization of TCI & Classical clusters

Cluster 3, which consists of six regions, is characterized by a high degree of wind energy, a low degree of hydroelectricity, low coverage of regional consumption, and low production and consumption of RE compared to the national average. Cluster 4, represented by the Corse region, is characterized by a high share of solar energy and low production and consumption of RE. The last class, represented by the Ile-de-France region, is characterized by a high share of biomass energy. Regarding the other types of RE, their share is close to the national average.

4 Conclusion

This paper proposes a new topological approach to the clustering of individuals which can enrich classical data analysis methods within the framework of the clustering of objects. The results of the topological clustering approach, based on the notion of a neighborhood graph, are as good - or even better, according to the R-squared results - than the existing classical method. The TCI approach is be easily programmable from the PCA and HAC procedures of SAS, SPAD or R software. Future work will involve extending this topological approach to other methods of data analysis, in particular in the context of evolutionary data analysis.

References

1. Abdesselam, R.: A topological clustering of variables. Journal of Mathematics and System Science. Accepted (2022)
2. Abdesselam, R.: A topological approach of Principal Component Analysis. International Journal of Data Science and Analysis. **77**(2), 20–31 (2021)
3. Abdesselam, R.: A topological Multiple Correspondence Analysis. Journal of Mathematics and Statistical Science, ISSN 2411-2518, **5**(8), 175–192 (2019)
4. Abdesselam, R.: A topological Discriminant Analysis. Data Analysis and Applications 2, Utilization of Results in Europe and Other Topics, Vol.3, Part 4. pp. 167–178 Wiley, (2019)
5. Batagelj, V., Bren, M.: Comparing resemblance measures. Journal of Classification, **12**(1), 73–90 (1995)
6. Caillez, F., Pagès, J. P.: Introduction à l'Analyse des Données. S.M.A.S.H., Paris (1976)
7. Kim, J. H. and Lee, S.: Tail bound for the minimal spanning tree of a complete graph. In Statistics & Probability Letters, **4**(64), 425–430 (2003)
8. Lebart, L.: Stratégies du traitement des données d'enquêtes. La Revue de MODULAD, **3**, 21–30 (1989)
9. Lesot, M. J., Rifqi, M., Benhadda, H.: Similarity measures for binary and numerical data: a survey. In: IJKESDP, **1**(1), 63–84 (2009)
10. Panagopoulos, D.: Topological data analysis and clustering. Chapter for a book, Algebraic Topology (math.AT) arXiv:2201.09054, Machine Learning (2022)
11. Park, J. C., Shin, H., Choi, B. K.: Elliptic Gabriel graph for finding neighbors in a point set and its application to normal vector estimation. Computer-Aided Design Elsevier, **38**(6), 619–626 (2006)
12. SAS Institute Inc. SAS/STAT Software, the Cluster Procedure, Available via DIALOG. `https://support.sas.com/documentation/onlinedoc/stat/142/cluster.pdf`
13. Selectra: Electricité renouvelable: quelles sont les régions les plus vertes de France ? `http://selectra.info/energie/actualites/expert/electricite-renouvelable-regions-plus-vertes-france` (2020)
14. Toussaint, G. T.: The relative neighbourhood graph of a finite planar set. Pattern Recognition, **12**(4) 261–268 (1980)
15. Ward, J. R.: Hierarchical grouping to optimize an objective function. Journal of the American Statistical Association, **58**(301), 236–244 (1963)
16. Zighed, D., Abdesselam, R., Hadgu, A.: Topological comparisons of proximity measures. In: Tan et al. (Eds). In Proc. 16th PAKDD 2012 Conference, pp. 379–391. Springer, (2012)

Model Based Clustering of Functional Data with Mild Outliers

Cristina Anton and Iain Smith

Abstract We propose a procedure, called CFunHDDC, for clustering functional data with mild outliers which combines two existing clustering methods: the functional high dimensional data clustering (FunHDDC) [1] and the contaminated normal mixture (CNmixt) [3] method for multivariate data. We adapt the FunHDDC approach to data with mild outliers by considering a mixture of multivariate contaminated normal distributions. To fit the functional data in group-specific functional subspaces we extend the parsimonious models considered in FunHDDC, and we estimate the model parameters using an expectation-conditional maximization algorithm (ECM). The performance of the proposed method is illustrated for simulated and real-world functional data, and CFunHDDC outperforms FunHDDC when applied to functional data with outliers.

Keywords: functional data, model-based clustering, contaminated normal distribution, EM algorithm

1 Introduction

Recently, model-based clustering for functional data has received a lot of attention. Real data are often contaminated by outliers that affect the estimations of the model parameters. Here we propose a method for clustering functional data with mild outliers. Mild outliers are usually sampled from a population different from the

Cristina Anton (✉)
MacEwan University, 10700 – 104 Avenue Edmonton, AB, T5J 4S2, Canada,
e-mail: popescuc@macewan.ca

Iain Smith
MacEwan University, 10700 – 104 Avenue Edmonton, AB, T5J 4S2, Canada,
e-mail: smithi23@mymacewan.ca

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_2

assumed model, so we need to choose a model flexible enough to accommodate them.

Functional data live in an infinite dimensional space and model-based methods for clustering are not directly available because the notion of probability density function generally does not exist for such data. A first approach is to use a two-step method and first do a discretization or a decomposition of the functional data in a basis of functions (such as Fourier series, B-splines, etc.), and then directly apply multivariate clustering methods to the discretization or the basis coefficients. A second approach, which allows the interaction between the discretization and the clustering steps, is based on a probabilistic model for the basis coefficients [1, 2].

We follow the second approach, and we propose a method, called CFunHDDC, which extends the functional high dimensional data clustering (FunHDDC) [1] to clustering functional data with mild outliers. There are several methods to detect outliers of functional data and a robust clustering methodology based on trimming is presented in [4]. Our approach does not involve trimming the outliers and it is inspired by the method CNmixt [3] for clustering multivariate data with mild outliers. We propose a model for the basis coefficients based on a mixture of contaminated multivariate normal distributions. A multivariate contaminated normal distribution is a two-component normal mixture in which the bad observations (outliers) are represented by a component with a small prior probability and an inflated covariance matrix.

In the next section we present the model and its parsimonious variants. Parameter estimation is included in Section 3. In Section 4 we present applications to simulated and real-world data. The last section includes the conclusions.

2 The Model

We suppose that we observe n curves $\{x_1, \ldots, x_n\}$ and we want to cluster them in K homogeneous groups. For each curve x_i we have access to a finite set of values $x_{ij} = x_i(t_{ij})$, where $0 \leq t_{i1} < t_{i2} < \ldots < t_{im_i} \leq T$. We assume that the observed curves are independent realizations of a L^2- continuous stochastic process $X = \{X(t)\}_{t\in[0,T]}$ for which the sample paths are in $L^2[0,T]$. To reconstruct the functional form of the data we assume that the curves belong to a finite dimensional space spanned by a basis of functions $\{\xi_1, \ldots, \xi_p\}$, so we have the expansion for each curve

$$x_i(t) = \sum_{j=1}^{p} \gamma_{ij}\xi_j(t).$$

Here we assume that the dimension p is fixed and known. We consider a model based on a mixture of multivariate contaminated normal distributions for the coefficients vectors $\{\gamma_1, \ldots, \gamma_n\} \subset \mathbb{R}^p$, $\gamma_i = (\gamma_{i1}, \ldots, \gamma_{ip})^\top \in \mathbb{R}^p$, $i = 1, \ldots, n$.

We suppose that there exists two unobserved random variables $Z = (Z_1, \ldots, Z_K)$, $\Upsilon = (\Upsilon_1, \ldots, \Upsilon_K) \in \{0,1\}^K$ where Z indicates the cluster membership and Υ

whether an observation is good or bad (outlier). $Z_k = 1$ if $X \in k$th cluster and $Z_k = 0$ otherwise, and $\Upsilon_k = 1$ if $X \in k$th cluster and it is a good observation, and $\Upsilon_k = 0$ otherwise. For clustering we need to predict the value $z_i = (z_{i1}, \ldots, z_{iK})$ of Z, and to determine the bad observations we need to predict the value $\nu_i = (\nu_{i1}, \ldots, \nu_{iK})$ of Υ for each observed curve x_i, $i = 1, \ldots, n$.

We consider a set of n_k observed curves of the kth cluster with the coefficients $\{\gamma_1, \ldots, \gamma_{n_k}\} \subset \mathbb{R}^p$. We assume that $\{\gamma_1, \ldots, \gamma_{n_k}\}$ are independent realizations of a random vector $\Gamma \in \mathbb{R}^p$, and that the stochastic process associated with the kth cluster can be described in a lower dimensional subspace $\mathbb{E}^k[0,T] \subset L^2[0,T]$ with dimension $d_k \leq p$ and spanned by the first d_k elements of a group specific basis of functions $\{\phi_{kj}\}_{j=1,\ldots,d_k}$ that can be obtained from $\{\xi_j\}_{j=1,\ldots,p}$ by a linear transformation

$$\phi_{kj} = \sum_{l=1}^{p} q_{k,jl}\xi_l,$$

with an $p \times p$ orthogonal matrix $Q_k = (q_{k,jl})$. In [1] for FunHDDC the assumption is that the distribution of Γ for the kth cluster is $\Gamma \sim N(\mu_k, \Sigma_k)$, $\Sigma_k = Q_k \Delta_k Q_k^\top$, where

$$\Delta_k = \left(\begin{array}{ccc|ccc} a_{k1} & & 0 & & & \\ & \ddots & & & \mathbf{0} & \\ 0 & & a_{kd_k} & & & \\ \hline & & & b_k & & 0 \\ & \mathbf{0} & & & \ddots & \\ & & & 0 & & b_k \end{array}\right) \Bigg\} p$$

with $a_{ki} > b_k$, $i = 1, \ldots, d_k$. We can say that the variance of the actual data in the kth cluster is modeled by $a_{k1}, \ldots, a_{kd_k}$ and the parameter b_k models the variance of the noise [1].

We follow the approach in [3] and we assume that Γ for the kth cluster has the multivariate contaminated normal distribution with density

$$f(\gamma_i; \theta_k) = \alpha_k \phi(\gamma_i; \mu_k, \Sigma_k) + (1 - \alpha_k)\phi(\gamma_i; \mu_k, \eta_k \Sigma_k), \tag{1}$$

where $\alpha_k \in (0.5, 1)$, $\eta_k > 1$, $\theta_k = \{\alpha_k, \mu_k, \Sigma_k, \eta_k\}$, and $\phi(\gamma_i; \mu_k, \Sigma_k)$ is the density for the $p-$variate normal distribution $N(\mu_k, \Sigma_k)$:

$$\phi(\gamma_i; \mu_k, \Sigma_k) = (2\pi)^{-p/2} |\Sigma_k|^{-1/2} \exp\left(-\frac{1}{2}(\gamma_i - \mu_k)^\top \Sigma_k^{-1} (\gamma_i - \mu_k)\right) \tag{2}$$

Here α_k defines the proportion of uncontaminated data in the $k$the cluster and η_k represents the degree of contamination. We can see η_k as an inflation parameter that measures the increase in variability due to the bad observations.

Each curve x_i has a basis expansion with coefficient γ_i such that γ_i is a random vector whose distributions is a mixture of contaminated Gaussians with density

$$p(\gamma;\theta)=\sum_{k=1}^{K}\pi_k f(\gamma;\theta_k) \tag{3}$$

where $\pi_k = P(Z_k = 1)$ is the prior probability of the kth the cluster and $\theta=\bigcup_{k=1}^{k}(\theta_k\cup\{\pi_k\})$ is the set formed by all the parameters. We refer to this model as FCLM$[a_{kj}, b_k, Q_k, d_k]$ (functional contaminated latent mixture). As in [1] we consider the parsimonious sub-models: FCLM$[a_{kj}, b, Q_k, d_k]$, FCLM$[a_k, b_k, Q_k, d_k]$, FCLM$[a, b_k, Q_k, d_k]$, FCLM$[a_k, b, Q_k, d_k]$, FCLM$[a, b, Q_k, d_k]$.

3 Model Inference

To fit the models we use the ECM algorithm [3], which is a variant of the EM algorithm. In the ECM algorithm we replace the M-step in the EM algorithm by two simpler CM-steps given by the partition of the set with the parameters $\theta=\{\Psi_1,\Psi_2\}$, where $\Psi_1=\{\pi_k,\alpha_k,\mu_k,a_{kj},b_k,q_{kj},k=1,\ldots,K,j=1,\ldots,d_k\}$, $\Psi_2=\{\eta_k,k=1,\ldots,K\}$, and q_{kj} is the jth column of Q_k.

We have two sources of missing data: the clusters' labels and the type of observation (good or bad). Thus the complete data are given by $S=\{\gamma_i,z_i,\nu_i\}_{i=1,\ldots,n}$, and the complete-data likelihood is

$$L_c(\theta;S)=\prod_{i=1}^{N}\prod_{k=1}^{K}\left\{\pi_k\left[\alpha_k\phi(\gamma_i;\mu_k,\Sigma_k)\right]^{\nu_{ik}}\left[(1-\alpha_k)\phi(\gamma_i;\mu_k,\eta_k\Sigma_k)\right]^{1-\nu_{ik}}\right\}^{z_{ik}}$$

We denote the complete-data log-likelihood by $l_c(\theta;S)=\log(L_c(\theta;S))$.

Next we present the ECM algorithm for the model FCLM$[a_{kj}, b_k, Q_k, d_k]$. At the q iteration of the ECM algorithm in the E-step we calculate $E[l_c(\theta^{(q-1)};S)|\gamma_1,\ldots,\gamma_n,\theta^{(q-1)}]$, given the current values of the parameters $\theta^{(q-1)}$. This reduces to the calculation of $z_{ik}^{(q)}:=E[Z_{ik}|\gamma_i,\theta^{(q-1)}]$, $\nu_{ik}^{(q)}:=E[\Upsilon_{ik}|\gamma_i,z_i,\theta^{(q-1)}]$.

In the first CM step in the q iteration of the ECM algorithm we calculate $\Psi_1^{(q)}$ as the value of Ψ_1 that maximize $l_c^{(q-1)}$ with Ψ_2 fixed at $\Psi_2^{(q-1)}$. We obtain

$$\pi_k^{(q)}=\frac{\sum_{i=1}^{n}z_{ik}^{(q)}}{n},\quad \alpha_k^{(q)}=\frac{\sum_{i=1}^{n}z_{ik}^{(q)}\nu_{ik}^{(q)}}{\sum_{i=1}^{n}z_{ik}^{(q)}},\quad \mu_k^{(q)}=\frac{\sum_{i=1}^{n}z_{ik}^{(q)}\left(\nu_{ik}^{(q)}+\frac{1-\nu_{ik}^{(q)}}{\eta_k^{(q-1)}}\right)\gamma_i}{\sum_{i=1}^{n}z_{ik}^{(q)}\left(\nu_{ik}^{(q)}+\frac{1-\nu_{ik}^{(q)}}{\eta_k^{(q-1)}}\right)} \tag{4}$$

$$\Sigma_k^{(q)}=\frac{1}{\sum_{i=1}^{n}z_{ik}^{(q)}}\sum_{i=1}^{n}z_{ik}^{(q)}\left(\nu_{ik}^{(q)}+\frac{1-\nu_{ik}^{(q)}}{\eta_k^{(q-1)}}\right)(\gamma_i-\mu_k^{(q)})(\gamma_i-\mu_k^{(q)})^{\top} \tag{5}$$

We introduce a value α^* and we constrain $\alpha_k\in(\alpha^*,1)$. If the estimation $\alpha_k^{(q)}$ in (4) is less than α^*, we use the *optimize*() function in the *stats* package in R to do a numerical search for $\alpha_k^{(q)}$.

As in [1] we get the updated values $a_{kj}^{(q)}, b_k^{(q)}, q_{kj}^{(q)}, k = 1, \ldots, K, j = 1, \ldots, d_k$ from the sample covariance matrix $\Sigma_k^{(q)}$ of cluster k, using also the matrix of inner products between the basis functions $W = (w_{jl})_{1 \leq j,l \leq p}$, where $w_{jl} = \int_0^T \xi_j(t)\xi_l(t)dt$.

In the second CM step in the q iteration of the ECM algorithm we calculate $\eta_k^{(q)}$ as the value that maximize $l_c^{(q-1)}$ with Ψ_1 fixed at $\Psi_1^{(q)}$.

At the end of the ECM algorithm, we do a two-step classification to provide the expected clustering. If q_f is the last iteration of the algorithm before convergence, an observation $\gamma_i \in \mathbb{R}^p$ is assigned to the cluster $k_0 \in \{1, \ldots, K\}$ with the largest $z_{ik}^{(q_f)}$. Next, an observation γ_i that was assigned to the cluster k_0 is considered good if $v_{ik_0}^{(q_f)} > 0.5$, and it is considered bad otherwise. After the classification step we can eliminate the bad observations and run FunHDDC to re-cluster the remaining observations.

The class specific dimension d_k is selected through the scree-test of Cattell by comparison of the difference between eigenvalues with a given threshold [1]. The number of clusters K as well as the parsimonious model are selected using the BIC criterion.

4 Applications

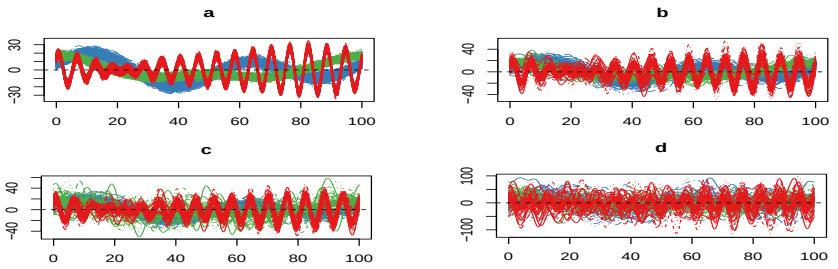


Fig. 1 Smooth data simulated without oultiers (a), according to scenario A (b), scenarion B (c), and scenario C (d), coloured by group for one simulation.

We simulate 1000 curves based on the model FCLM$[a_k, b_k, Q_k, d_k]$. The number of clusters is fixed to $K = 3$ and the mixing proportions are equal $\pi_1 = \pi_2 = \pi_3 = 1/3$. We consider the following values of the parameters

Group 1: $d = 5$, $a = 150$, $b = 5$, $\mu = (1, 0, 50, 100, 0, \ldots, 0)$
Group 2: $d = 20$, $a = 15$, $b = 8$, $\mu = (0, 0, 80, 0, 40, 2, 0, \ldots, 0)$

Group 3: $d = 10$, $a = 30$, $b = 10$, $\mu = (0, \ldots, 0, 20, 0, 80, 0, 0, 100)$,

where d is the intrinsic dimension of the subgroups, μ is the mean vector of size 70, a is the value of the d-first diagonal elements of Δ, and b the value of the $70 - d$- last ones. Curves are smoothed using 35 Fourier basis functions. We repeat the simulation 100 times. A sample of theses data is plotted in Figure 1 a. We consider the following contamination schemes where the scores are simulated from contaminated normal distributions with the previous parameters and

A: $\alpha_i = 0.9$, $i = 1, \ldots, 3$, and $\eta_1 = 7$, $\eta_2 = 10$, $\eta_3 = 17$.
B: $\alpha_i = 0.9$, $i = 1, \ldots, 3$, and $\eta_1 = 5$, $\eta_2 = 50$, $\eta_3 = 15$.
C: $\alpha_i = 0.9$, $i = 1, \ldots, 3$, and $\eta_1 = 100$, $\eta_2 = 70$, $\eta_3 = 170$.

Samples for data generated according to scenarios A, B, C are plotted in Figure 1 b, c, d, respectively. We notice that there is more overlapping between the 3 groups when we increase the values of η.

Table 1 Mean (and standard deviation) of ARI for BIC best model on 100 simulations. Bold values indicates the highest value for each method.

Scenario	Method	α^*	ϵ	ARI	ARI Outliers
A	FunHDDC	-	0.05	**0.519 (0.11)**	-
A	FunHDDC	-	0.1	0.499(0.05)	-
A	FunHDDC	-	0.2	0.494 (0.01)	-
A	CFunHDDC	0.75	0.05	0.769 (0.23)	0.959(0.04)
A	CFunHDDC	0.75	0.1	0.986(0.08)	0.998(0.01)
A	CFunHDDC	0.75	0.2	**0.9995 (0.001)**	**1 (0)**
B	FunHDDC	-	0.05	**0.861 (0.23)**	-
B	FunHDDC	-	0.1	0.754(0.25)	-
B	FunHDDC	-	0.2	0.52 (0.09)	-
B	CFunHDDC	0.75	0.05	0.807 (0.22)	0.961(0.05)
B	CFunHDDC	0.75	0.1	0.948 (0.14)	0.99(0.03)
B	CFunHDDC	0.75	0.2	**0.990 (0.062)**	**0.971 (0.149)**
C	FunHDDC	-	0.05	0.490 (0.02)	-
C	FunHDDC	-	0.1	0.491(0.02)	-
C	FunHDDC	-	0.2	**0.494 (0.01)**	-
C	CFunHDDC	0.75	0.05	0.736 (0.23)	0.928(0.10)
C	CFunHDDC	0.75	0.1	0.911 (0.18)	0.958(0.15)
C	CFunHDDC	0.75	0.2	**0.965 (0.11)**	**0.994 (0.03)**

The quality of the estimated partitions obtained using FunHDDC and CFunHDDC is evaluated using the Adjusted Rand Index (ARI) [3], and the results are included in Table 1. For FunHDDC we use the library *funHDDC* in R. We run both algorithms for $K = 3$ with all 6 sub-models and the best solution in terms of the highest BIC value for all those submodels is returned. The initialization is done with the k-means

Table 2 Correct classification rates for each method.

Method	ϵ	CCR	Method	α^*	ϵ	CCR	Method	α^*	CCR
FunHDDC	0.01	0.68	CFunHDDC	0.85	0.01	0.67	CNmixt	0.5	0.67
FunHDDC	0.05	0.64	CFunHDDC	0.85	0.05	0.70	CNmixt	0.75	0.66
FunHDDC	0.1	0.59	CFunHDDC	0.85	0.1	0.70	CNmixt	0.85	0.67
FunHDDC	0.2	0.57	CFunHDDC	0.85	0.2	0.6	CNmixt	0.9	0.66

strategy with 50 repetitions, and the maximum number of iterations is 200 for the stopping criterion. We use $\epsilon \in \{0.05, 0.1, 0.2\}$ in the Cattell test.

We notice that CFunHDDC outperforms FunHDDC, and it gives excellent results even in Scenario C. For CFunHDDC the best results are obtained for $\epsilon = 0.2$ in the Catell test, and the values of the ARI are close to 1.

Next, we consider the NOx data available in the *fda.usc* library in R and representing daily curves of Nitrogen Oxides (NOx) emissions in the neighborhood of the industrial area of Poblenou, Barcelona (Spain). The measurements of NOx (in $\mu g/m^3$) were taken hourly resulting in 76 curves for "working days" and 39 curves for "non-working days" (see Figure 2 a). Since NOx is a contaminant agent, the detection of outlying emission is useful for environmental protection. This data set has been used for testing methods for the detection of outliers and to illustrate robust clustering based on trimming for functional data [4].

We apply CFunHDDC, FunHDDC, and CNmixt to the NOx data. Curves are smoothed using a basis of 8 Fourier functions, and we run the algorithms for $K = 2$ clusters. For CFunHDDC, FunHDDC we use $\epsilon \in \{0.001, 0.05, 0.1, 0.2\}$ in the Cattell test and the rest of the settings are the same as in the simulation study. We run CNmixt for all 14 models from the *ContaminatedMixt* R library, based on the coefficients in the Fourier basis, with 1000 iterations for the stopping criteria, and initialization done with the k-means method. The correct classification rates (CCR) are reported in Table 2.

The CCR for CFunHDDC are slightly better than the ones for FunHDDC and CNmixt, and are comparable with the ones reported in Table 1 in [4] for Funclust,

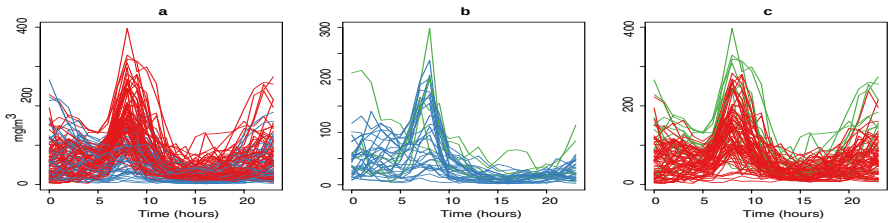


Fig. 2 a.Daily NOx curves for 115 days; b. c. Clustering obtained with CFunHDDC, $\epsilon = 0, 05, \alpha^* = 0.85$; Non-working days (blue), working days (red), outliers (green).

RFC, and TrimK. In Figure 2 b, c we present the clusters and the detected outliers for $\epsilon = 0.05$ and $\alpha^* = 0.85$. The curves that are detected as outliers (green lines) exhibit different patterns from the rest of the curves.

One of the advantages of extending the FunHDDC to CFunHDDC is the outlier detection. For $\alpha^* = 0.85$ and $\epsilon = 0.05$, CFunHDDC detects 16 outliers, which are the same with the outliers mentioned in [4]. For the data without outliers, CFunHDDC becomes equivalent to FunHDDC, and for the trimmed data the CCR increases to 0.79.

5 Conclusion

We propose a new method, CFunHDDC, that extends the FunHDDC functional clustering method to data with mild outliers. Unlike other robust functional clustering algorithms, CFunHDDC does not involve trimming the data. CFunHDDC is based on a model formed by a mixture of contaminated multivariate normal distributions, which makes parameter estimation more difficult than for FunHDDC, so we use an ECM instead of an EM algorithm. The clustering and outlier detection performance of CFunHDDC is tested for simulated data and the NOx data and it always outperforms FunHDDC. Moreover, CFunHDDC has a comparable performance with robust functional clustering methods based on trimming, such as RFC and TrimK, and it has similar or better performance when compared to a two-step method based on CNmixt. Although there are several model-based methods for multivariate data with outliers that can be used to construct two-step methods for functional data, as observed in [1], these two-step methods always suffers from the difficulty to choose the best discretization. CFunHDDC can be extended to multivariate functional data, and recently, independently of our work, a similar approach was followed in [5], but without considering the parsimonious models and the value α^*.

References

1. Bouveyron, C., Jacques, J.: Model-based clustering of time series in group-specific functional subspaces. Adv. Data. Anal. Classif. **5**(4), 281–300 (2011)
2. Jacques, J., Preda, C.: Funclust: a curves clustering method using functional random variables density approximation. Neurocomputing **112**, 164–171 (2013)
3. Punzo, A., McNicholas, P. D.: Parsimonious mixtures of multivariate contaminated normal distributions. Biom. J. **58**, 1506–1537 (2016)
4. Rivera-Garcia, D., Garcia-Escudero, L. A., Mayo-Iscar, A., Ortega, J.: Robust clustering for functional data based on trimming and constraints. Adv. Data Anal. Classif. **13**, 201–225 (2019)
5. Amovin-Assagba, M., Gannaz, I., Jacques, J.: Outlier detection in multivariate functional data through a contaminated mixture model. (2021)
https://doi.org/10.48550/arXiv.2106.07222

A Trivariate Geometric Classification of Decision Boundaries for Mixtures of Regressions

Filippo Antonazzo and Salvatore Ingrassia

Abstract Mixtures of regressions play a prominent role in regression analysis when it is known the population of interest is divided into homogeneous and disjoint groups. This typically consists in partitioning the observational space into several regions through particular hypersurfaces called decision boundaries. A geometrical analysis of these surfaces allows to highlight properties of the used classifier. In particular, a geometrical classification of decision boundaries for the three most used mixtures of regressions (with fixed covariates, with concomitant variables and random covariates) was provided in case of one and two covariates, under Gaussian assumptions and in presence of only one real response variable. This work aims to extend these results to a more complex setting where three independent variables are considered.

Keywords: mixtures of regressions, decision boundaries, hyperquadrics, model-based clustering

1 Introduction

Linear regression is commonly employed to model the relationship between a d-dimensional real vector of covariates $\mathbf{X}$ and a real response variable Y. It is well suited if we can assume that regression coefficients are fixed over all possible realizations $(\mathbf{x}, y) \in \mathbb{R}^{d+1}$ of the couple $(\mathbf{X}, Y)$. This assumption falls if it is a-priori known that realizations come from a population Ω which can be partitioned into G disjoint

Filippo Antonazzo (✉)
Inria, Université de Lille, CNRS, Laboratoire de mathématiques Painlevé 59650 Villeneuve d'Ascq, France, e-mail: filippo.antonazzo@inria.fr

Salvatore Ingrassia
Dipartimento di Economia e Impresa, Università di Catania, Corso Italia 55, 95129 Catania, Italy, e-mail: salvatore.ingrassia@unict.it

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_3

homogeneous groups $\Omega_g, g = 1, \ldots, G$. In this case, a mixture of linear regressions (or clusterwise regression) is a more appropriate statistical tool. According to their degree of flexibility and generality, we can distinguish three types of mixtures of regressions: mixtures of regressions with fixed covariates (MRFC) [3]; mixtures of regressions with concomitant variables (MRCV) [6] and mixtures of regressions with random covariates (MRRC), also referred to in literature as cluster-weighted models [3, 4].

Mixtures of regressions can also be employed from a classification point of view to identify the group membership of each observation. In this case, the generated classifier divides the real space into G regions through particular $\mathbb{R}^{d+1}$ surfaces called decision boundaries. In [5], the decision boundaries generated by each type of mixture are analyzed from a geometrical point of view, especially in those cases where $d = 1, 2$ and $G = 2$. The aim of the present work is to extend the results presented in the aforementioned paper to a higher dimensional case where $d = 3$, giving more insight into the properties of these classifiers. The rest of the paper is organized as follows. In Section 2 we summarize the main ideas about mixtures of regressions. In Section 3 decision boundaries will be defined, finally proposing a geometrical classification in Section 4 when $d = 3$ and $G = 2$. In Section 5, we will conclude investigating with practical example the shape of three-dimensional decision boundaries in presence of variables following heavy-tailed t-distributions.

2 Mixtures of Regressions

Below we briefly define three types of mixtures of regressions, ordered according to their generality and flexibility, given by an increasing number of parameters.

MRFC. Mixtures of regressions with fixed covariates have the following density:

$$p(y|\mathbf{x}; \psi) = \sum_{g=1}^{G} \pi_g f(y|\mathbf{x}; \theta_g). \tag{1}$$

The density $f(y|\mathbf{x}; \theta_g)$ is indexed by a parameter vector θ_g belonging to an Euclidean parametric space Θ_g. Moreover, every π_g is positive and $\sum_{g=1}^{G} \pi_g = 1$. The vector $\psi = (\pi_1, \ldots, \pi_G, \theta_1, \ldots, \theta_G)$ denotes the set of all the parameters of the model.

MRCV. The density of a mixture of regressions with concomitant variables is:

$$p(y|\mathbf{x}; \psi) = \sum_{g=1}^{G} f(y|\mathbf{x}; \theta_g) p(\Omega_g|\mathbf{x}; \alpha), \tag{2}$$

where the vector $\psi = (\theta_1, \ldots, \theta_G, \alpha)$ contains all parameters indexing the model. More specifically, $p(\Omega_g|\mathbf{x}; \alpha)$ is a function depending on $\mathbf{x}$ according to a vector of real parameters α. Typically, the probability $p(\Omega_g|\mathbf{x}; \alpha)$ is a multinomial logistic

density with $\alpha = (\alpha_1^t, \ldots, \alpha_G^t)^t$ and $\alpha_g = (\alpha_{g0}, \alpha_{g1}^t)^t \in \mathbb{R}^{d+1}$, i.e.:

$$p(\Omega_g|\mathbf{x};\alpha) = \frac{\exp(\alpha_{g0} + \alpha_{g1}^t\mathbf{x})}{\sum_{g=1}^{G}\exp(\alpha_{g0} + \alpha_{g1}^t\mathbf{x})}.$$

Due to identifiability reasons, it is necessary to add the constraint $\alpha_1 = \mathbf{0}$, see [2].

MRRC. Mixtures of regressions with random covariates propose the following decomposition for the conjoint density $p(\mathbf{x}, y;\psi)$:

$$p(\mathbf{x}, y;\psi) = \sum_{g=1}^{G} f(y|\mathbf{x},\theta_g)p(\mathbf{x};\xi_g)\pi_g, \tag{3}$$

where $\pi_g > 0$ and $\sum_{g=1}^{G}\pi_g = 1$. Furthermore, the model is totally parametrized by the vector $\psi = (\pi_1, \ldots, \pi_G, \theta_1, \ldots, \theta_G, \xi_1, \ldots, \xi_G)$, where each θ_g indexes the conditional density $f(y|\mathbf{x},\theta_g)$, while each ξ_g refers to the density of $\mathbf{X}$ in the group Ω_g, denoted with $p(\mathbf{x};\xi_g)$.

In particular, under Gaussian assumptions it results $Y|\mathbf{x},\Omega_g \sim N(\beta_{g0}+\beta_{g1}^t\mathbf{x},\sigma_g^2)$, where each $\beta_g = (\beta_{g0}, \beta_{g1})$ is a vector of real parameters. Only for MRRC model, we will further assume $\mathbf{X}|\mathbf{\Omega}_g \sim N(\mu_g, \Sigma_g)$ for all $g = 1, \ldots, G$, where μ_g denotes the mean of the Gaussian distribution, while Σ_g is its covariance matrix. Denoting with $\phi(\cdot)$ the Gaussian density function, equations (1)-(3) can be, respectively, rewritten as

$$p(y|\mathbf{x};\psi) = \sum_{g=1}^{G}\phi(y;\beta_{g0} + \beta_{g1}^t\mathbf{x};\sigma_g^2)\pi_g, \tag{4}$$

$$p(y|\mathbf{x};\psi) = \sum_{g=1}^{G}\phi(y;\beta_{g0} + \beta_{g1}^t\mathbf{x};\sigma_g^2)p(\Omega_g|\mathbf{x};\alpha), \tag{5}$$

$$p(\mathbf{x}, y;\psi) = \sum_{g=1}^{G}\phi(y,\beta_{g0} + \beta_{g1}^t\mathbf{x};\sigma_g^2)\phi(\mathbf{x};\mu_g,\Sigma_g)\pi_g. \tag{6}$$

Maximum likelihood estimate for ψ are usually obtained with the Expectation-Maximization (EM) algorithm. Then, the final estimate is used to build classifiers which group observations into G disjoint classes.

3 Decision Boundaries: Generality

There are different ways to build classifiers. One of the best known is the method of discriminant functions. The aim of this procedure is to define G functions $D_g(\mathbf{x}, y;\psi)$ and a decision rule to divide the real space $\mathbb{R}^{d+1}$ into G decision regions, named

$\mathcal{R}_1, \ldots, \mathcal{R}_G$. The decision regions have a one-to-one relationship with the subgroups Ω_g, i.e., if an observation $(\mathbf{x}, y) \in \mathbb{R}^{d+1}$ is assigned to $\mathcal{R}_g$, it is classified as part of Ω_g. Among all possible decision rules, the most used one consists in assigning $(\mathbf{x}, y)$ to $\mathcal{R}_g$ if:

$$D_g(\mathbf{x}, y; \psi) > D_j(\mathbf{x}, y; \psi) \quad \forall j \neq g. \tag{7}$$

Then, decision boundaries are defined as the surfaces in $\mathbb{R}^{d+1}$ separating the decision regions $\mathcal{R}_g$, where observations cannot be uniquely classified. Formally, each decision boundary is a hypersurface represented by the mathematical equation $D_j(\mathbf{x}, y; \psi) - D_k(\mathbf{x}, y; \psi) = 0$, $j \neq k$.

Different choices for discriminant functions are possible: under Gaussian assumptions it is convenient to define $D_g(\cdot)$ as the logarithm of the g-th component mixture density, as it conveys useful computational simplification [5]. So, we can define, for all the three models, these discriminant functions:

$$MRFC: \quad D_g(\mathbf{x}, y; \psi) = \ln[\phi(y; \beta_{g0} + \beta_{g1}^t \mathbf{x}, \sigma_g^2)\pi_g] \tag{8}$$

$$MRCV: \quad D_g(\mathbf{x}, y; \psi) = \ln[\phi(y; \beta_{g0} + \beta_{g1}^t \mathbf{x}, \sigma_g^2) \exp(\alpha_{g0} + \alpha_{g1}^t \mathbf{x})] \tag{9}$$

$$MRRC: \quad D_g(\mathbf{x}, y; \psi) = \ln[\phi(y; \beta_{g0} + \beta_{g1}^t \mathbf{x}, \sigma_g^2)\phi(\mathbf{x}; \mu_g, \Sigma_g)\pi_g] \tag{10}$$

3.1 The Case with $G = 2$

In the case of interest where $G = 2$, there is a single decision boundary defined by the equation $D(\mathbf{x}, y; \psi) = D_2(\mathbf{x}, y; \psi) - D_1(\mathbf{x}, y; \psi) = 0$. Thus, the assignment rule for every point $(\mathbf{x}, y) \in \mathbb{R}^{d+1}$ is based on the sign of $D(\mathbf{x}, y; \psi)$. It assigns $(\mathbf{x}, y)$ to Ω_2 if $D(\mathbf{x}, y; \psi) > 0$; to Ω_1, otherwise.

In [5] the geometrical properties of the hypersurfaces, defined by the equation $D(\mathbf{x}, y; \psi) = 0$, have been investigated up to dimension $d = 2$, providing the following propositions for quadrics.

Proposition 1 (MRFC quadrics) *The decision boundary between Ω_1 and Ω_2 is always a degenarate quadric.*

Proposition 2 (MRCV quadrics) *If $\alpha^t(\beta_{21} - \beta_{11}) \neq 0$, then the decision boundary between Ω_1 and Ω_2 is a paraboloid; otherwise it is a degenarate quadric.*

Proposition 3 (MRRC quadrics) *Under convenient conditions, the decision boundary between Ω_1 and Ω_2 can be a degenerate quadric but it can be also assume any of the general quadric forms.*

These results show that models with more flexibility, i.e. with more parameters, can generate more varieties of decision boundaries. In the following section, we will extend these statements to dimension $d = 3$.

4 Geometrical Classification of Decision Boundaries with $G = 2$ and $D = 3$

In this section we extend previous results for mixtures of regression in presence of two classes and $d = 3$, where decision boundaries reveal to be hyperquadrics in $\mathbb{R}^4$. Mathematical proofs of results for MRFC and MRCV models are based on an algebraic analysis of the matrices representing these hyperquadrics.

MRFC. Mixtures of regressions with fixed covariates are characterized by a low degree of flexibility. Indeed, all decision boundaries are degenerate hyperquadrics as the following result shows.

Proposition 4 (MRFC hyperquadrics) *The decision boundary between* Ω_1 *and* Ω_2 *is a degenerate hyperquadric of rank at most equal to* 3. *The rank is less than 3 if* $\beta_{11} = \beta_{21}$ *or* $\frac{1-\pi_1}{\pi_1} = \frac{\sigma_2}{\sigma_1}$.

MRCV. A MRCV allows more degrees of freedom than a MRFC. A consequence is that the obtained decision boundaries are higher rank hyperquadrics, as the following result states.

Proposition 5 (MRCV hyperquadrics) *The decision boundary between* Ω_1 *and* Ω_2 *is a degenerate hyperquadric with rank at most equal to 4. In particular, rank is equal to 4 if* $\alpha^t(\beta_{21} - \beta_{11}) \neq 0$. *In addition, if* $\alpha^t(\beta_{21} - \beta_{11}) = 0$ *and* $\sigma_1^2 = \sigma_2^2$; *the matrix has rank at most equal to 2, therefore the hyperquadric is reducible.*

MRRC. Proposition 3 shows MRRC exhibit a high number of possible types of conics and quadrics [5]. This fact is confirmed in dimension $d = 3$, even if a strong theoretical result is difficult to obtain with simple algebra due to the mathematical complexity of the MRRC hyperquadric matrix. Indeed, it is possible to show such flexibility by building several practical examples (not displayed here), where hyperquadrics of various shapes arise.

Analyzing the provided results, we can note that they perfectly match the hierarchy established in dimension $d = 2$. Indeed, a MRFC can generate only degenerate hyperquadrics of rank 3; the surfaces generated by a MRCV, which has more parameters, are still degenerate, but with a higher rank (equal to 4) depending on the same mathematical condition of Proposition 2; finally a MRRC, the most flexible model in terms of number of parameters, can give rise to various hyperquadrics, as in $d = 2$.

5 Beyond Gaussian Assumptions: t-distribution in $d = 2$

In [5], Gaussian assumptions were crossed by illustrating the case of a simple linear regression ($G = 2$ and $d = 1$) where more general t-distributions were required

for robustness reasons. It is shown that the generated decision boundaries are more flexible than their Gaussian counterparts, as they can assume more various shapes, although these surfaces can be calculated only numerically. In this section, we continue the exploration of the t-distribution case adding one more variable, thus $d = 2$. Under these more general assumptions, discriminant functions (8) – (10) become:

$$MRFC\text{-}t: \quad D_g(\mathbf{x}, y; \psi) = \ln[q(y; \beta_{g0} + \beta_{g1}^t \mathbf{x}, \sigma_g^2, \eta_g)\pi_g], \tag{11}$$

$$MRCV\text{-}t: \quad D_g(\mathbf{x}, y; \psi) = \ln[q(y; \beta_{g0} + \beta_{g1}^t \mathbf{x}, \sigma_g^2, \eta_g) \exp(\alpha_{g0} + \alpha_{g1}^t \mathbf{x})], \tag{12}$$

$$MRRC\text{-}t: \quad D_g(\mathbf{x}, y; \psi) = \ln[q(y; \beta_{g0} + \beta_{g1}^t \mathbf{x}, \sigma_g^2, \eta_g) q(\mathbf{x}; \mu_g, \mathbf{\Sigma}_g, \nu_g)\pi_g], \tag{13}$$

where $q(y; \beta_{g0} + \beta_{g1}^t \mathbf{x}, \sigma_g^2, \eta_g)$ denotes a generalized t-distribution density, with non-centrality parameter equal to $\beta_{g0} + \beta_{g1}^t \mathbf{x}$, scaling parameter equal to σ_g^2 and degrees of freedom given by η_g. Similarly, $q(\mathbf{x}; \mu_g, \mathbf{\Sigma}_g, \nu_g)$ is a multivariate generalized t-distribution density, where μ_g is the non-centrality parameter, $\mathbf{\Sigma}_g$ denotes the scaling and ν_g represents the degrees of freedom. Figure 1-2 display the decision boundaries for the three considered models whose parameters are presented in Table 1: they clearly show the gain in flexibility given by the more general distributional assumptions. Moreover, t-boundaries with $\eta_1 = \eta_2 = 10$ (Figure 2; red curves) seem to be closer to Gaussian ones (blue curves) than those with $\eta_1 = \eta_2 = 3$ (Figure 1; orange curves): this is coherent with standard probabilistic theory.

Table 1 Parameters used in Figure 1-2. MRRC: covariance matrices $\mathbf{\Sigma_1}$ and $\mathbf{\Sigma_2}$ are equal to the identity matrix $\mathbf{I}_2$.

Model	Group	π_g	β_{g0}	β_{g1}	σ_g^2	α_{g0}	α_{g1}	μ_g	ν_g
MRFC	1	0.3	1	(2,-3)	0.5				
	2	0.7	1	(-4,3)	0.5				
MRCV	1	0.3	1	(2,-3)	0.5				
	2	0.7	1	(-4,3)	0.5	1	(-1,0.5)		
MRRC	1	0.3	1	(2,-3)	0.5			(1,2)	5
	2	0.7	1	(-4,3)	0.5	1	(-1,0.5)	(-1,-2)	5

6 Conclusions

This work has provided a trivariate multivariate geometrical classification for the decision boundaries generated by mixtures of regressions in presence of two classes. Under Gaussian assumptions, our results confirmed the same hierarchy that was shown in $d = 2$, as MRRC turns out to exhibits a huge variety of decision boundaries, while other models generate only degenerate surfaces. This is coherent with its high degree of flexibility given by its very general parametrization. The provided results

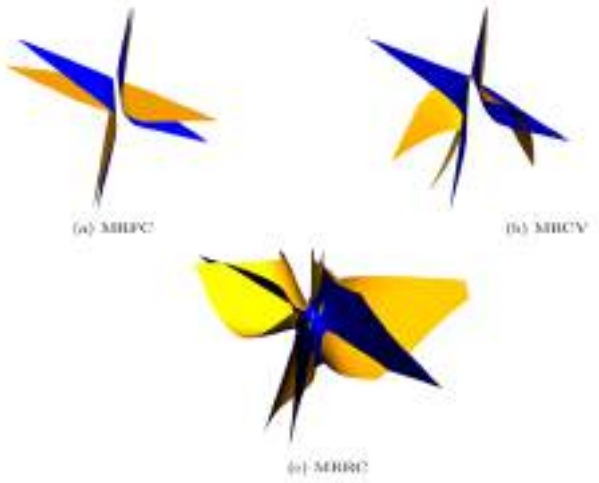


Fig. 1 Decision boundaries under assumptions of Gaussian (in blue) and t-distributed variables with $\eta_1 = \eta_2 = 3$ (in orange) for the three considered mixtures of regressions.

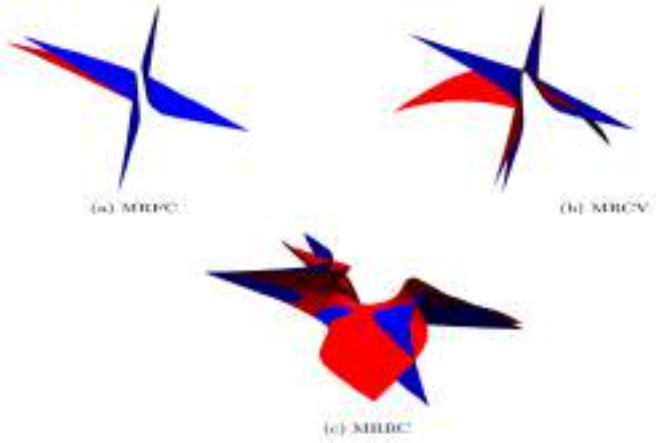


Fig. 2 Decision boundaries under assumptions of Gaussian (in blue) and t-distributed variables with $\eta_1 = \eta_2 = 10$ (in red) for the three considered mixtures of regressions.

could help to select the right model depending on the shape of data. For example, if in a descriptive analysis data turn out to be approximately separated by a simple degenerate hyperquadric, it will be better to estimate a MRFC or a MRCV instead of a complex MRRC. On the contrary, if the separation surface seems to be non-degenerate, then it will be preferable to fit a general MRRC. Moreover, this work also showed that the degree of flexibility (thus, the variety of possible decision boundaries) can be enhanced by go further Gaussianity, assuming, for example, t-distributed variables. This encourage additional extensions where more general

distributions can be included, allowing a better comprehension of mixtures and possible applications to generalized linear models where categorical variables are considered.

References

1. DeSarbo, W. S., Cron, W. L.: A maximum likelihood methodology for clusterwise linear regression. J. Classif. **5**, 249–282 (1988)
2. Grun, B., Leisch, F.: FlexMix version 2: finite mixtures with concomitant variables and varying and constant parameters. J. Stat. Softw. **28**, 1–35 (2008)
3. Hennig, C.: Identifiablity of models for clusterwise linear regression. J. Classif. **17**, 273–296 (2000)
4. Ingrassia, S., Minotti, S. C., Vittadini, G.: Local Statistical Modeling via a Cluster-Weighted Approach with Elliptical Distributions. J. Classif. **29**, 363-401 (2012)
5. Ingrassia, S., Punzo, A.: Decision boundaries for mixtures of regressions. J. Korean Stat. Soc. **45**, 295-306 (2016)
6. Wedel, M.: Concomitant variables in finite mixture models. Stat. Neerl. **56**, 362–375 (2002)

Generalized Spatio-temporal Regression with PDE Penalization

Eleonora Arnone, Elia Cunial, and Laura M. Sangalli

Abstract We develop a novel generalised linear model for the analysis of data distributed over space and time. The model involves a nonparametric term f, a smooth function over space and time. The estimation is carried out by the minimization of an appropriate penalized negative log-likelihood functional, with a roughness penalty on f that involves space and time differential operators, in a separable fashion, or an evolution partial differential equation. The model can include covariate information in a semi-parametric setting. The functional is discretized by means of finite elements in space, and B-splines or finite differences in time. Thanks to the use of finite elements, the proposed method is able to efficiently model data sampled over irregularly shaped spatial domains, with complicated boundaries. To illustrate the proposed model we present an application to study the criminality in the city of Portland, from 2015 to 2020.

Keywords: functional data analysis, spatial data analysis, semiparametric regression with roughness penalty

Eleonora Arnone (✉)
Dipartimento di Scienze Statistiche, Università di Padova, Via Cesare Battisti, 241, 35121 Padova, Italy, e-mail: eleonora.arnone@unipd.it

Elia Cunial
Dipartimento di Matematica, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milano, Italy, e-mail: elia.cunial@mail.polimi.it

Laura M. Sangalli
Dipartimento di Matematica, Politecnico di Milano, Piazza Leonardo da Vinci 32, 20133 Milano, Italy, e-mail: laura.sangalli@polimi.it

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_4

1 Introduction

In this work we develop a novel generalised linear model for the analysis of data distributed over space and time. Let Y be a real-valued variable of interest, and $\mathbf{W}$ a vector of q covariates, observed in n spatio-temporal locations $\{\mathbf{p}_i, t_i\}_{i=1,\dots,n} \in \Omega \times T$, where $\Omega \subset \mathbb{R}^2$ is a bounded spatial domain, and $T \subset \mathbb{R}$ a temporal interval. We assume that the expected value of Y, conditional on the covariates and the location of observation, can be modeled as:

$$g(\mathbb{E}[Y|W, \mathbf{p}, t]) = \mathbf{W}^\top \boldsymbol{\beta} + f(\mathbf{p}, t)$$

where g is a known monotone link function, chosen on the basis of the stochastic nature of Y, $\boldsymbol{\beta} \in \mathbb{R}^q$ is an unknown vector of regression coefficients, and $f : \Omega \times T \to \mathbb{R}$ is an unknown deterministic function, which captures the spatio-temporal variation of the phenomenon under study. Starting from the values $\{y_i, \mathbf{w}_i\}_{i=1,\dots,n}$, of the observed response variable and covariates, we estimates $\boldsymbol{\beta}$ and f in a semiparametric fashion. In particular, following the approach in [9], that consider a similar problem for data scattered over space only, we minimize the functional

$$\ell\Big(\{y_i, \mathbf{w}_i, \mathbf{p}_i, t_i\}_{i=1,\dots,n}; \boldsymbol{\beta}, f\Big) + \mathcal{P}(f)$$

where ℓ is the appropriate negative log-likelihood, and $\mathcal{P}(f)$ is a penalty that enforces f to be a regular function.

Similarly to the regression methods in [1, 2, 3, 4, 5, 7, 8], the roughness penalty on f, $\mathcal{P}(f)$, involves some partial differential operators. In particular, our aim is to extend the Spatial-Temporal regression with partial differential equations regularization (ST-PDE), developed in [2, 3, 4], to generalized linear model settings, further broadening the class of regression models with PDE regularization reviewed in [6]. Hence, likewise ST-PDE, the proposed generalized linear model has a roughness penalty that involves a second order linear differential operator L applied to f. Specifically, as in [4], we may consider the penalty

$$\mathcal{P}(f) = \lambda_T \int_\Omega \int_0^T \left(\frac{\partial^2 f}{\partial t^2}\right)^2 + \lambda_S \int_\Omega \int_0^T (Lf)^2,$$

where the first term accounts for the regularity of the function in time, while the second accounts for the regularity of the function in space; the importance of each term is controlled by two smoothing parameters λ_T and λ_S. Alternatively, as in [2], we may consider a single penalty which accounts for the spatial and temporal regularity:

$$\mathcal{P}(f) = \lambda \int_\Omega \int_0^T \left(\frac{\partial f}{\partial t} + Lf - u\right)^2.$$

Differently from the models in [2, 3, 4], the estimation functional to be minimized is not quadratic. This poses increased difficulties from the computational point

of view. The minimization is performed via a functional version of the penalized iterative reweighted least square algorithm.

The estimation problem is appropriately discretized. In particular, in time, the discretization involves either cubic B-splines, for the two-penalties case, or finite differences, when the single penalty is employed. The discretization in space is performed via finite elements, on a triangulation of the spatial domain of interest. This enables to appropriately considered spatial domains with complicated boundaries, such as the one considered in the following section, concerning the study of criminality data over the city of Portland.

2 Application to Criminality Data

This section describes the Portland criminality data, that will be used to illustrate the proposed methodology. We will present a Poisson model to count the crimes in the city, and study their evolution from April 2015 to November 2020. In addition, we shall consider as a covariate the population of the city neighborhoods. The crime data are publicly available on the website of the Police Bureau of the city[1].

The crimes counts are aggregated by trimesters and at a neighborhoods level. Figure 1 shows the city neighborhoods, each neighborhood colored according to its total population. The bottom part of the same figure shows the temporal evolution of the crimes in each neighborhood. Each curve corresponds to a neighborhood and is colored according to the neighborhood population. In both panels, the three neighborhoods with the highest number of crimes are indicated by numbers 1, 2 and 3. The figure highlights the presence of some correlation between neighborhood population and the number of crimes. However, criminality is not fully explained by population. For instance, neighborhoods 1 and 3 present an high number of crimes with a moderate population. This raises the interest towards a semiparametric generalized linear model, as the one introduced in Section 1, with a nonparametric term accounting for the spatio-temporal variability in the phenomenon, that cannot be explained by population or other census quantities. Figure 2 shows the same data for four different trimesters on the Portland map. As already pointed out, the three area with the highest number of crimes are in the city center, and in the Hazelwood neighborhood, in the east part of the city.

From Figures 1 and 2 we can see that the shape of the domain is complicated; the city is indeed crossed by a river, with few bridges connecting the two parts, most of them placed downtown. Therefore, neighborhoods at opposite side of the river and far from the center, where most bridges are located, are close in euclidean distance, but far apart in reality. This particular morphology influences the phenomenon under study, for example, in the north of the city, the east side of the river is characterized by an higher number of crimes with respect to the west part. Due to this characteristics of the data and the domain, is is of crucial importance to take into account the shape

[1] Police Bureau crime data: `https://www.portlandoregon.gov/police/71978`

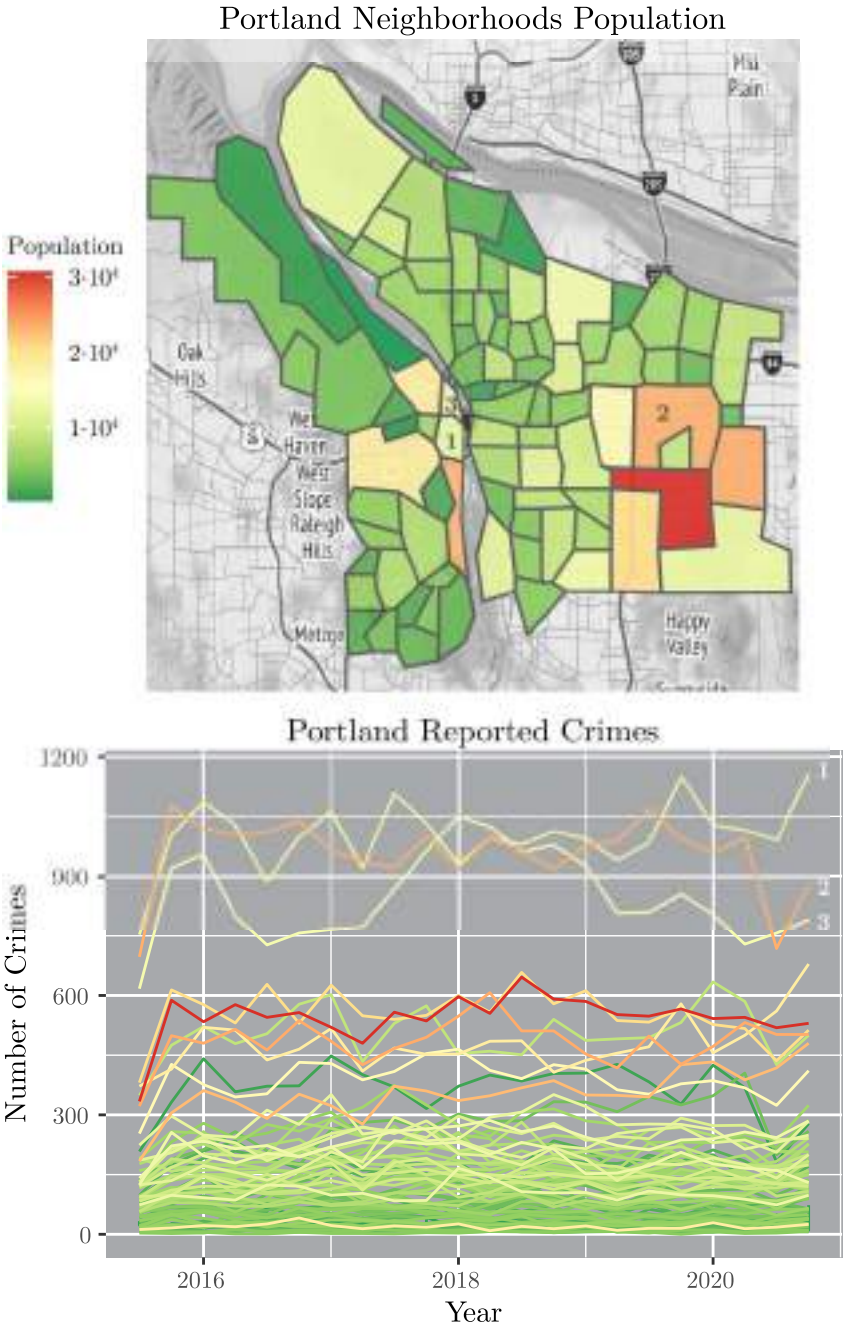


Fig. 1 Top: the city of Portland divided into neighborhoods, each neighborhood colored according to the total population. Bottom: the total crimes over time for each neighborhood; each curve corresponds to a neighborhood and is colored according to the neighborhood's population. The three neighborhoods with the highest number of crimes are indicated by numbers 1, 2 and 3.

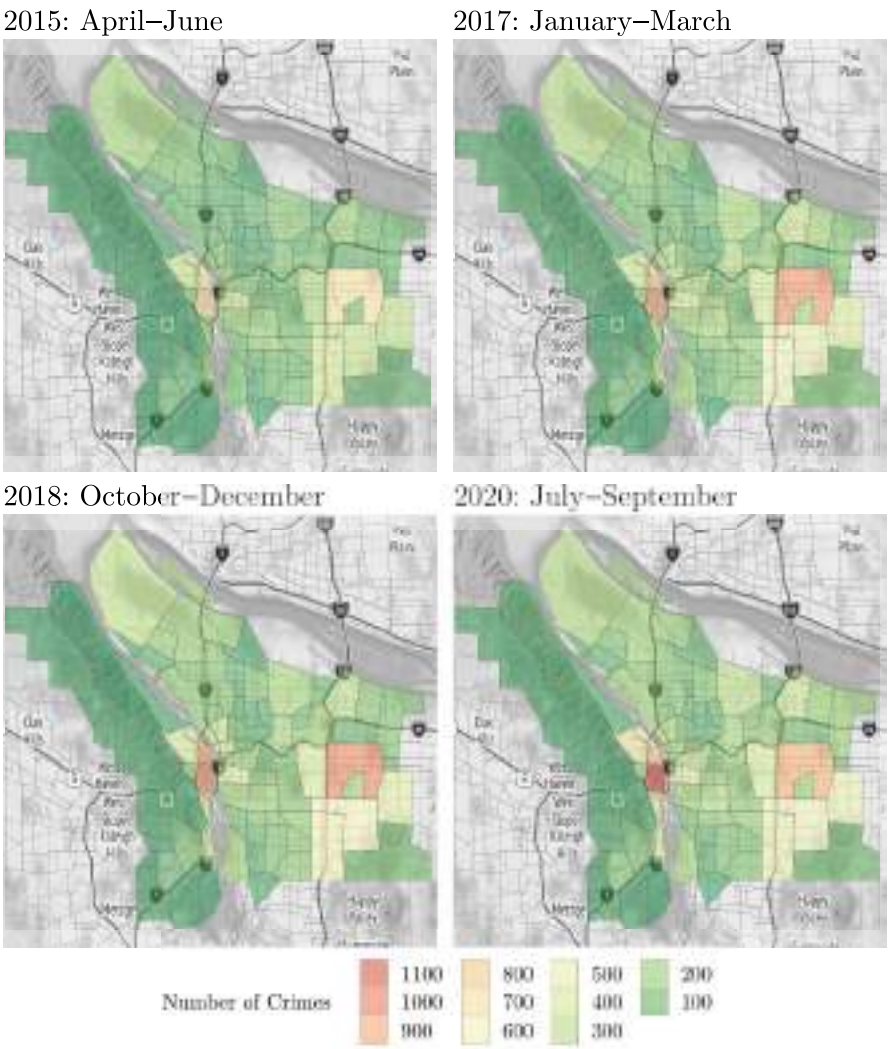


Fig. 2 Total crime counts per neighborhood per trimester; green indicates lower number of crimes, red indicates a higher number of crimes.

of the domain during the estimation process. For this reason, estimation based on classical semiparametric models, such as those based on thin-plate splines, would give poor results, while the proposed method is particularly well suited, being able to complying the nontrivial form of the domain.

References

1. Aguilera-Morillo, M. C., Durbán, M., Aguilera, A. M.: Prediction of functional data with spatial dependence: a penalized approach. Stoch. Environ. Res. Risk Assess. **31**, 7–22 (2017)
2. Arnone, E., Azzimonti, L., Nobile, F., Sangalli, L. M.: Modeling spatially dependent functional data via regression with differential regularization. J. Multivariate Anal. **170**, 275–295 (2019)
3. Arnone, E., Sangalli, L. M., Vicini, A.: Smoothing spatio-temporal data with complex missing data patterns. Stat. Model. Int. J. (2021)
4. Bernardi, M. S., Sangalli, L. M., Mazza, G., Ramsay, J. O.: A penalized regression model for spatial functional data with application to the analysis of the production of waste in Venice province. Stoch. Environ. Res. Risk Assess. **31**, 23–38 (2017)
5. Marra, G., Miller, D. L., Zanin, L.: Modelling the spatiotemporal distribution of the incidence of resident foreign population. Statistica Neerlandica **66**(2) 133–160 (2012)
6. Sangalli, L. M.: Spatial regression with partial differential equation regularization. Int. Stat. Rev. **89**(3), 505–531 (2021)
7. Ugarte, M. D., Goicoa, T., Militino, A. F., Durbán, M.: Spline smoothing in small area trend estimation and forecasting. Comput. Stat. Data Anal. **53**(10), 3616–3629 (2009)
8. Ugarte, M. D., Goicoa, T., Militino, A. F.: Spatio-temporal modeling of mortality risks using penalized splines. Environmetrics **21**, 270–289 (2010)
9. Wilhelm M., Sangalli L. M.: Generalized spatial regression with differential regularization. J. Stat. Comput. Simulat. **86**(13), 2497–2518 (2016)

A New Regression Model for the Analysis of Microbiome Data

Roberto Ascari and Sonia Migliorati

Abstract Human microbiome data are becoming extremely common in biomedical research due to the relevant connections with different types of diseases. A widespread discrete distribution to analyze this kind of data is the Dirichlet-multinomial. Despite its popularity, this distribution often fails in modeling microbiome data due to the strict parameterization imposed on its covariance matrix. The aim of this work is to propose a new distribution for analyzing microbiome data and to define a regression model based on it. The new distribution can be expressed as a structured finite mixture model with Dirichlet-multinomial components. We illustrate how this mixture structure can improve a microbiome data analysis to cluster patients into "enterotypes", which are a classification based on the bacteriological composition of gut microbiota. The comparison between the two models is performed through an application to a real gut microbiome dataset.

Keywords: count data, Bayesian inference, mixture model, multivariate regression

1 Introduction

The human microbiome is defined as the set of genes associated with the microbiota, i.e. the microbial community living in the human body, including bacteria, viruses and some unicellular eukaryotes [1, 8]. The mutualistic relationship between microbiota and human beings is often beneficial, though it can sometimes

Roberto Ascari (✉)
Department of Economics, Management and Statistics (DEMS), University of Milano-Bicocca, Milan, Italy, e-mail: roberto.ascari@unimib.it

Sonia Migliorati
Department of Economics, Management and Statistics (DEMS), University of Milano-Bicocca, Milan, Italy, e-mail: sonia.migliorati@unimib.it

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_5

become detrimental for several health outcomes. For example, changes in the gut microbiome composition can be associated with diabetes, cardiovascular disesase, obesity, autoimmune disease, anxiety and many other factors impacting on human health [1, 5, 12, 14]. Moreover, the development of next-generation sequencing technologies allows nowadays to survey the microbiome composition using direct DNA sequencing of either marker genes or the whole metagenomics, without the need for isolation and culturing. These are the two main reasons for the recent explosion of research on microbiome, and highlight the importance of understanding the association between microbiome composition and biological and environmental covariates.

A widespread distribution for handling microbiome data is the Dirichlet-multinomial (DM) (e.g., see [4, 16]), a generalization of the multinomial distribution obtained by assuming that, instead of being fixed, the underlying taxa proportions come from a Dirichlet distribution. This allows to model overdispersed data counts, that is data showing a variance much larger than that predicted by the multinomial model. Despite its popularity, the DM distribution is often inadequate to model real microbiome datasets due to the strict covariance structure imposed by its parameterization, which hinders the description of co-occurrence and co-exclusion relationships between microbial taxa.

The aim of this work is to propose a new distribution that generalizes the DM, namely the flexible Dirichlet-multinomial (FDM), and a regression model based on it. The new model provides a better fit to real microbiome data, still preserving a clear interpretation of its parameters. Moreover, being a finite mixture with DM components, it enables to account for the data latent group structure, and thus to identify clusters sharing similar biota compositions.

2 Statistical Models for Microbiome Data

In this section, we define a new distribution for multivariate counts and a regression model based on it, that allows to link microbiome abundances with covariates. Note that, once the DNA sequence reads have been aligned to the reference microbial genomes, the abundances of microbial taxa can be quantified. Thus, microbiome data represent the count composition of D bacterial taxa in a specific biological sample, and a microbiome dataset is a sequence of D-dimensional vectors $\mathbf{Y}_1, \mathbf{Y}_2, \ldots, \mathbf{Y}_N$, where Y_{ir} counts the number of occurrences of taxon r in the i-th sample ($i = 1, \ldots, N$ and $r = 1, \ldots, D$). Since the i-th sample contains a number n_i of bacteria, microbiome observations are subject to a fixed-sum constraint, that is $\sum_{r=1}^{D} Y_{ir} = n_i$.

2.1 Count Distributions

Following a compound approach, we assume that $\mathbf{Y}|\mathbf{\Pi} = \boldsymbol{\pi} \sim \text{Multinomial}(n, \boldsymbol{\pi})$, and we consider suitable distributions for the vector of probabilities $\mathbf{\Pi} \in \mathcal{S}^D$. The set $\mathcal{S}^D = \{\boldsymbol{\pi} = (\pi_1, \ldots, \pi_D)^\intercal : \pi_r > 0, \sum_{r=1}^D \pi_r = 1\}$ is the D-part simplex and it is the proper support of continuous compositional vectors. A distribution for $\mathbf{Y}$ is obtained by marginalizing the joint distribution of $(\mathbf{Y}, \mathbf{\Pi})^\intercal$. A common choice for this distribution is the mean-precision parameterized Dirichlet, whose probability density function (p.d.f.) is

$$f_{\text{Dir}}(\boldsymbol{\pi}; \boldsymbol{\mu}, \alpha^+) = \frac{\Gamma(\alpha^+)}{\prod_{r=1}^D \Gamma(\alpha^+ \mu_r)} \prod_{r=1}^D \pi_r^{(\alpha^+ \mu_r)-1},$$

where $\boldsymbol{\mu} = \mathbb{E}[\mathbf{\Pi}] \in \mathcal{S}^D$, and $\alpha^+ > 0$ is a precision parameter. Compounding the multinomial distribution with the Dirichlet one leads to the DM distribution, widely used in microbiome data analysis, whose probability mass function (p.m.f.) is

$$f_{\text{DM}}(\mathbf{y}; n, \boldsymbol{\mu}, \alpha^+) = \frac{n!\Gamma(\alpha^+)}{\Gamma(\alpha^+ + n)} \prod_{r=1}^D \frac{\Gamma(\alpha^+ \mu_r + y_r)}{(y_r!)\Gamma(\alpha^+ \mu_r)}.$$

The mean vector of a DM distribution is $\mathbb{E}[\mathbf{Y}] = n\boldsymbol{\mu}$, so that the parameter $\boldsymbol{\mu} = \mathbb{E}[\mathbf{Y}]/n$ can be thought of as a scaled mean vector. Moreover, its covariance matrix is

$$\mathbb{V}[\mathbf{Y}] = n\mathbf{M}\left[1 + \frac{n-1}{\alpha^+ + 1}\right], \tag{1}$$

where $\mathbf{M} = (\text{Diag}(\boldsymbol{\mu}) - \boldsymbol{\mu}\boldsymbol{\mu}^\intercal)$. Equation (1) highlights how the additional parameter α^+ allows to increase flexibility in the variability structure with respect to the standard multinomial distribution.

We propose to take advantage of an alternative sound distribution defined on $\mathcal{S}^D$, namely the flexible Dirichlet (FD) [7, 9]. The latter is a structured finite mixture with Dirichlet components, entailing some constraints among the components' parameters to ensure model identifiability. Thanks to its mixture structure, the p.d.f. of a FD-distributed random vector can be expressed as

$$f_{\text{FD}}(\boldsymbol{\pi}; \boldsymbol{\mu}, \alpha^+, w, \mathbf{p}) = \sum_{j=1}^D p_j f_{\text{Dir}}\left(\boldsymbol{\pi}; \boldsymbol{\lambda}_j, \frac{\alpha^+}{1-w}\right), \tag{2}$$

where

$$\boldsymbol{\lambda}_j = \boldsymbol{\mu} - w\mathbf{p} + w\mathbf{e}_j \tag{3}$$

is the mean vector of the j-th component, $\boldsymbol{\mu} = \mathbb{E}[\mathbf{\Pi}] \in \mathcal{S}^D$, $\alpha^+ > 0$, $\mathbf{p} \in \mathcal{S}^D$, $0 < w < \min\left\{1, \min_{r \in \{1,\ldots,D\}}\left\{\frac{\mu_r}{p_r}\right\}\right\}$, and $\mathbf{e}_j$ is a vector with all elements equal to zero except for the j-th which is equal to one.

Equation (2) points that the Dirichlet components have different mean vectors and a common precision parameter, the latter being determined by α^+ and w. In particular, inspecting Equation (3), it is easy to observe that any two vectors $\boldsymbol{\lambda}_r$ and $\boldsymbol{\lambda}_h$, $r \neq h$, coincide in all the elements except for the r-th and the h-th.

If $\boldsymbol{\Pi}$ is supposed to be FD distributed, a new discrete distribution for count vectors can be defined (we shall call flexible Dirichlet-multinomial (FDM)). The p.m.f. of the FDM can be expressed as

$$\begin{aligned} f_{\text{FDM}}(\mathbf{y}; n, \boldsymbol{\mu}, \alpha^+, \mathbf{p}, w) &= \sum_{j=1}^{D} p_j f_{\text{DM}}\left(\mathbf{y}; n, \boldsymbol{\lambda}_j, \frac{\alpha^+}{1-w}\right) \\ &= \sum_{j=1}^{D} p_j \frac{n!\Gamma(\frac{\alpha^+}{1-w})}{\Gamma(\frac{\alpha^+}{1-w}+n)} \prod_{r=1}^{D} \frac{\Gamma(\frac{\alpha^+}{1-w}\lambda_{jr}+y_r)}{(y_r!)\Gamma(\frac{\alpha^+}{1-w}\lambda_{jr})}, \end{aligned} \tag{4}$$

where $\boldsymbol{\lambda}_j$ is defined in Equation (3). Interestingly, it is possible to recognize the flexible beta-binomial (FBB) [3] distribution as a special case of the FDM. The FBB is a generalization of the binomial distribution successful in dealing with overdispersion. Moreover, note that when $\mathbf{p} = \boldsymbol{\mu}$ and $w = 1/(\alpha^+ + 1)$ the DM distribution is recovered.

Equation (4) shows that the FDM is a finite mixture with DM components displaying a common precision parameter and different scaled mean vectors $\boldsymbol{\lambda}_j$, $j = 1, \ldots, D$. The overall mean vector and the covariance matrix of the FDM can be expressed as

$$\begin{aligned} \mathbb{E}[\mathbf{Y}] &= n\boldsymbol{\mu}, \\ \mathbb{V}[\mathbf{Y}] &= n\mathbf{M}\left[1 + \frac{n-1}{\phi+1}\right] + n\frac{(n-1)\phi w^2}{\phi+1}\mathbf{P}, \end{aligned} \tag{5}$$

where $\mathbf{M} = (\text{Diag}(\boldsymbol{\mu}) - \boldsymbol{\mu}\boldsymbol{\mu}^\intercal)$, $\mathbf{P} = (\text{Diag}(\mathbf{p}) - \mathbf{p}\mathbf{p}^\intercal)$, and $\phi = \alpha^+/(1-w)$ is the common precision parameter of the DM components. A comparison between Equations (5) and (1) points out that the covariance matrix of the FDM distribution is a very easily interpretable extension of the DM's covariance matrix. Indeed, it is composed of two terms, the first one coinciding with the DM's covariance matrix, whereas the second one depends on the mixture structure of the FDM model. In particular, the FDM covariance matrix has D additional parameters with respect to the DM, namely $D-1$ distinct elements in the vector of mixing weights $\mathbf{p}$, and the parameter w which controls the distance among the components' barycenters [7]. This is the key element explaining the better ability of the FDM in modeling a wide range of scenarios.

2.2 Regression Models

With the aim of performing a regression analysis, let $\mathbf{Y} = (\mathbf{Y}_1, \ldots, \mathbf{Y}_N)^\intercal$ be a set of independent multivariate responses collected on a sample of N subjects/units. For the i-th subject, $\mathbf{Y}_i$ counts the number of times that each of D possible taxa occurred among n_i trials, and $\mathbf{x}_i$ is a $(K+1)$-dimensional vector of covariates.

A parameterization of the FDM useful in a regression perspective is the one based on $\boldsymbol{\mu}$, $\mathbf{p}$, α^+, and $\tilde{w}$, where

$$\tilde{w} = \frac{w}{\min\left\{1, \min_r\left\{\frac{\mu_r}{p_r}\right\}\right\}} \in (0,1). \tag{6}$$

We can define the FDM regression (FDMReg) and the DM regression (DMReg) models assuming that $\mathbf{Y}_i$ follows an FDM$(n_i, \boldsymbol{\mu}_i, \alpha^+, \mathbf{p}, \tilde{w})$ or a DM$(n_i, \boldsymbol{\mu}_i, \alpha^+)$ distribution, respectively. Even if the FDM and DM distributions do not belong to the dispersion-exponential family, we can follow a GLM-type approach, [6] by linking the parameter $\boldsymbol{\mu}_i$ to the linear predictor through a proper link function such as the multinomial logit link function, that is

$$g(\mu_{ir}) = \log\left(\frac{\mu_{ir}}{\mu_{iD}}\right) = \mathbf{x}_i^\intercal \boldsymbol{\beta}_r, \qquad r = 1, \ldots, D-1, \tag{7}$$

where $\boldsymbol{\beta}_r = (\beta_{r0}, \beta_{r1}, \ldots, \beta_{rK})^\intercal$ is a vector of regression coefficients for the r-th element of $\boldsymbol{\mu}_i$. Note that the last category has been conventionally chosen as baseline category, thus $\boldsymbol{\beta}_D = \mathbf{0}$.

The parameterization of the FDMReg based on $\boldsymbol{\mu}$, $\mathbf{p}$, α^+, and $\tilde{w}$ defines a variation independent parameter space, meaning that no constraints exist among parameters. In a Bayesian framework, this allows to assume prior independence, and, consequently, we can specify a prior distribution for each parameter separately. In order to induce minimum impact on the posterior distribution, we select weakly-informative priors: (i) $\boldsymbol{\beta}_r \sim N_{K+1}(\mathbf{0}, \Sigma)$, where $\mathbf{0}$ is the $(K+1)$-vector with zero elements, and Σ is a diagonal matrix with 'large' variance values, (ii) $\alpha^+ \sim Gamma(g_1, g_2)$ for small values of g_1 and g_2, (iii) $\tilde{w} \sim Unif(0,1)$, and (iv) a uniform prior on the simplex for $\mathbf{p}$.

Inferential issues are dealt with by a Bayesian approach through a Hamiltonian Monte Carlo (HMC) algorithm [10], which is a popular generalization of the Metropolis-Hastings algorithm. The Stan modeling language [13] allows implementing an HMC method to obtain a simulated sample from the posterior distribution.

To compare the fit of the models we use the Watanabe-Akaike information criterion (WAIC) [15, 17], a fully Bayesian criterion that balances between goodness-of-fit and complexity of a model: lower values of WAIC indicate a better fit.

3 A Gut Microbiome Application

In this section, we fit the DM and the FDM regression models to a microbiome dataset analyzed by Xia et al. [19] and previously proposed by Wu et al. [18]. They collected gut microbiome data on 98 healthy volunteers. In particular, the counts of three bacteria genera were recorded, namely Bacteroides, Prevotella, and Ruminococcus. Arumugam et al. [2] used these three bacteria to define three groups they called enterotypes. These enterotypes provide information about the human's body ability to produce vitamins.

Wu et al. analyzed the same dataset conducting a cluster analysis via the 'partitioning around medoids' (PAM) approach. They detected only two of the three enterotypes defined in the work by Arumugam et al. Moreover, these two clusters are characterized by different frequencies: 86 out of the 98 samples were allocated to the first enterotype, whereas only 12 samples were clustered into enterotype 2. This is due to the small number of subjects with a high abundance of Prevotella (i.e., only 36 samples showed a Prevotella count greater than 0).

Besides the bacterial data, we consider also $K = 9$ covariates, representing information on micro-nutrients in the habitual long-term diet collected using a food frequency questionnaire. These 9 additional variables have been selected by Xia et al. using a l_1 penalized regression approach.

Table 1 shows the posterior mean and 95% credible set (CS) of each parameter involved in the DMReg and the FDMReg models. Though the significant covariates are the same across the models, the FDMReg shows a lower WAIC, thus being the best model in terms of fit. This is due to the additional set of parameters involved in the mixture structure that help in providing information on this dataset.

The mixture structure of the FDMReg model can be exploited to cluster observations into groups through a model-based approach. More specifically, each observation can be allocated to the mixture component that most likely generated it. Indeed, note that the mixing weights estimates (0.637, 0.357 and 0.006, from Table 1) confirm the presence of two out of the three enterotypes defined by Arumugam et al. [2]. To further illustrate the benefits of the FDReg model in a microbiome data analysis, we compare the clustering profile obtained by the FDMReg model and the one obtained with the PAM approach used by Wu et al. In particular, Table 2 summarizes this comparison in a confusion matrix. Despite the clustering generated by the FDMReg being based on some distributional assumptions (i.e., the response is FDM distributed), it highly agrees with the one obtained by the PAM algorithm for 84% of the observations. This percentage is obtained using the covariates selected by Xia et al. in a logistic normal multinomial regression model context. Clearly, the results could be improved by developing an ad hoc variable selection procedure for the FDMReg model. The main advantage to considering the FDMReg (that is a model-based clustering approach) is that, besides the clustering of the data points, it provides also some information on the detected clusters (e.g., their size and a measure of their distance) and the relationship between the response and the set of covariates. This additional information may increase the insight we can gain from

Table 1 Posterior mean and 95% CS for the parameters of the DMReg and FDMReg models. Regression coefficients in bold are related to 95% CS's not containing the zero value.

		DM		FDM	
		Post. Mean	95% CS	Post. Mean	95% CS
Bacteroides	Intercept	**2.197**	**(1.844, 2.546)**	**2.642**	**(2.215, 3.034)**
	Proline	-0.039	(-0.344, 0.273)	-0.036	(-0.325, 0.261)
	Sucrose	-0.257	(-0.555, 0.039)	-0.208	(-0.471, 0.064)
	Vitamin E, food fortification	-0.016	(-0.351, 0.336)	-0.043	(-0.351, 0.299)
	Beta cryptoxanthin	-0.073	(-0.357, 0.237)	-0.059	(-0.334, 0.214)
	Added germa from wheats	-0.147	(-0.477, 0.196)	-0.042	(-0.411, 0.271)
	Vitamin C	0.300	(-0.031, 0.771)	0.267	(-0.035, 0.673)
	Maltose	-0.031	(-0.311, 0.260)	0.034	(-0.237, 0.302)
	Palmitelaidic trans fatty acid	0.019	(-0.292, 0.328)	-0.044	(-0.336, 0.251)
	Acrylamide	0.133	(-0.167, 0.455)	0.184	(-0.094, 0.474)
Prevotella	Intercept	**-1.196**	**(-1.715, -0.699)**	-0.402	(-1.094, 0.245)
	Proline	-0.053	(-0.571, 0.443)	-0.018	(-0.663, 0.546)
	Sucrose	0.029	(-0.437, 0.476)	0.126	(-0.335, 0.591)
	Vitamin E, food fortification	0.109	(-0.355, 0.548)	0.113	(-0.473, 0.574)
	Beta cryptoxanthin	0.263	(-0.230, 0.762)	0.349	(-0.386, 0.812)
	Added germa from wheats	0.280	(-0.137, 0.701)	0.121	(-0.298, 0.604)
	Vitamin C	-0.169	(-1.196, 0.623)	-0.021	(-1.131, 0.738)
	Maltose	**0.640**	**(0.164, 1.126)**	**0.877**	**(0.260, 1.400)**
	Palmitelaidic trans fatty acid	**-0.530**	**(-1.008, -0.043)**	**-0.716**	**(-1.209, -0.140)**
	Acrylamide	**0.780**	**(0.362, 1.206)**	**0.800**	**(0.382, 1.231)**
	α^+	1.541	(1.104, 2.040)	2.275	(1.489, 3.208)
	p_1	—	—	0.637	(0.420, 0.797)
	p_2	—	—	0.357	(0.197, 0.570)
	p_3	—	—	0.006	(0.000, 0.027)
	$\tilde{w}$	—	—	0.914	(0.791, 0.991)
	WAIC	1686.2		1662.3	

data. Further improvements could be obtained considering an even more flexible distribution for $\mathbf{\Pi}$, that is the extended flexible Dirichlet [11].

Table 2 Confusion matrix for clustering based on the FDMReg model compared to the PAM algorithm.

		FDMReg	
		1	2
PAM	1	70	16
	2	0	12

References

1. Amato, K.: An introduction to microbiome analysis for human biology applications. Am. J. Hum. Biol. **29** (2017)

2. Arumugam, M. et al.: Enterotypes of the human gut microbiome. Nature. **473**, 174–180 (2011)
3. Ascari, R., Migliorati, S.: A new regression model for overdispersed binomial data accounting for outliers and an excess of zeros. Stat. Med. **40**(17), 3895–3914 (2021)
4. Chen, J., Li, H.: Variable selection for sparse Dirichlet-multinomial regression with an application to microbiome data analysis. Ann. Appl. Stat. **7**(1), 418–442 (2013)
5. Koeth, R. A. et al.: Intestinal microbiota metabolism of L-carnitine, a nutrient in red meat, promotes atherosclerosis. Nat. Med. **19**(5) (2013)
6. McCullagh, P., Nelder, J. A.: Generalized Linear Models. Chapman & Hall (1989)
7. Migliorati, S., Ongaro, A., Monti, G. S.: A structured Dirichlet mixture model for compositional data: inferential and applicative issues. Stat. Comput. 27(4), 963–983, 2017.
8. Morgan, X. C., Huttenhower, C.: Human microbiome analysis. PloS Computational Biology. **8**(12) (2012)
9. Ongaro, A., Migliorati, S.: A generalization of the Dirichlet distribution. J. Multivar. Anal. **114**, 412–426 (2013)
10. Neal, R. M.: An improved acceptance procedure for the hybrid Monte Carlo algorithm. Tech. Rep. (1994)
11. Ongaro, A., Migliorati, S., Ascari, R.: A new mixture model on the simplex. Stat. Comput. **30**(4), 749–770 (2020)
12. Qin, J., Li, Y., Cai, Z., Li, S., Zhu, J., Zhang, F., Liang, S., Zhang, W., Guan, Y., Shen, D., Peng, Y.: A metagenome-wide association study of gut microbiota in type 2 diabetes. Nature. 490 (2012)
13. Stan Development Team: Stan Modeling Language Users Guide and Reference Manual (2017)
14. Turnbaugh, P. J. et al.: A core gut microbiome in obese and lean twins. Nature. 457 (2009)
15. Vehtari, A., Gelman, A., Gabry, J.: Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. Stat. Comput. **27**(5), 1413–1432 (2017)
16. Wadsworth, W. D., Argiento, R., Guindani, M., Galloway-Pena, J., Shelburne, S. A., Vannucci, M.: An integrative Bayesian Dirichlet-multinomial regression model for the analysis of taxonomic abundances in microbiome data. BMC Bioinformatics. **18**(94) (2017)
17. Watanabe, S.: A widely applicable Bayesian information criterion. J. Mach. Learn. Tech. **14**(1), 867–897 (2013)
18. Wu., G. D. et al.: Linking long-term dietary patterns with gut microbial enterotypes. Science. 334, 105–109 (2011)
19. Xia, F., Chen, J., Fung, W. K., Li, H.: A logistic normal multinomial regression model for microbiome compositional data analysis. Biometrics. **69**(4), 1053–1063 (2013)

Stability of Mixed-type Cluster Partitions for Determination of the Number of Clusters

Rabea Aschenbruck, Gero Szepannek, and Adalbert F. X. Wilhelm

Abstract For partitioning clustering methods, the number of clusters has to be determined in advance. One approach to deal with this issue are stability indices. In this paper several stability-based validation methods are investigated with regard to the *k-prototypes* algorithm for mixed-type data. The stability-based approaches are compared to common validation indices in a comprehensive simulation study in order to analyze preferability as a function of the underlying data generating process.

Keywords: cluster stability, cluster validation, mixed-type data

1 Introduction

In cluster analysis practice, it is common to work with mixed-type data (i.e. numerical and categorical variables), while in theoretical development the research is traditionally often restricted to numerical data. A comprehensive overview on cluster analysis based on mixed-type data is given in [1]. To cluster these mixed-type data, a popular approach is the *k-prototypes* algorithm, an extension of the popular *k-means* algorithm, as proposed in [2] and implemented in [3].

As for all partitioning clustering methods, the number of clusters has to be specified in advance. In the past, several validation methods have been identified for the

Rabea Aschenbruck (✉)
Stralsund University of Applied Sciences, Zur Schwedenschanze 15, 18435 Stralsund, Germany,
e-mail: rabea.aschenbruck@hochschule-stralsund.de

Gero Szepannek
Stralsund University of Applied Sciences, Zur Schwedenschanze 15, 18435 Stralsund, Germany,
e-mail: gero.szepannek@hochschule-stralsund.de

Adalbert F.X. Wilhelm
Jacobs University Bremen, Campus Ring 1, 28759 Bremen, Germany,
e-mail: A.Wilhelm@jacobs-university.de

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_6

k-prototypes algorithm to enable the rating of clusters and to determine the index optimal number of clusters. A brief overview is given in Section 2, followed by an examination of the investigated stability indices to improve clustering mixed-type data[1]. In Section 3, a simulation study has been conducted in order to compare the performance of stability indices as well as a new proposed adjustment, and additionally to rate the performance with respect to internal validation indices. Finally, a summary, which does not state a superiority of the stability-based approaches over internal validation indices in general, and an outlook are given in Section 4.

2 Stability of Cluster Partitions

The assessment of cluster quality can be used for the comparison of clusters resulting from different methods or from the same method but with different input parameters, e.g., with a different number of clusters. Especially the latter has already been an important issue in partitioning clustering many decades ago [5]. Since then, some work has been done on this subject. Hennig [6] points out, that nowadays some literature uses the term *cluster validation* exclusively for methods that decide about the optimal number of clusters, in the following named *internal validation*. An overview of internal validation indices is given, e.g., in [7] or [8]. In [9], a set of internal cluster validation indices for mixed-type data to determine the number of clusters for the *k-prototypes* algorithm was derived and analyzed. In the following, stability indices are presented, before they are compared to each other and additionally to internal validation indices in Section 3. Since cluster stability is a model agnostic method, the indices are applicable to any clustering algorithm and not limited to numerical data [10].

A partition S splits data $Y = \{y_1, \ldots, y_n\}$ into K groups $S_1, \ldots, S_K \subseteq Y$. The focus of this paper is on the evaluation and rating of cluster partitions with so-called stability indices. To calculate these, as discussed by Dolnicar and Leisch [11] or mentioned by Fang and Wang [12], $b \in \{1, \ldots, B\}$ bootstrap samples Y^b (with replacement, see e.g. [13]) from the original data set Y are drawn. For every bootstrap sample Y^b, a cluster partition $S^b = \{S_1^b, \ldots, S_{L_b}^b\}$ is determined. For the validation of the different results of these bootstrap samples, the set of points from the original data set that are also part of the b-th bootstrap sample $X^b = Y \cap Y^b$ is used, where n_b is the size of X^b. Furthermore $C^b = \{S_k \cap X^b | k = 1, \ldots, K\}$ and $D^b = \{S_l^b \cap X^b | l = 1, \ldots, L_b\}$, with $B_C^\star$ being the number of bootstrap samples for which $C^b \neq \emptyset$, and n_{S_k}, $n_{C_k^b}$, $n_{S_l^b}$, and $n_{D_l^b}$ with $k \in \{1, \ldots, K\}, l \in \{1, \ldots, L_b\}$ are the numbers of objects in cluster group S_k, C_k^b, S_l^b and D_l^b, respectively.

In 2002, Ben-Hur et al. [14] presented stability-based methods, which can be used to define the optimal number of clusters. In their work, the basis for the calculation of the stability indices is a binary matrix P^{C^b}, which represents the cluster partition C^b in the following way

[1] The mentioned and analyzed stability indices will extend the R package `clustMixType` [4].

$$P_{ij}^{C^b} = \begin{cases} 1, \text{ if objects } x_i^b, x_j^b \in X^b \text{ are in the same cluster and } i \neq j, \\ 0, \text{ otherwise.} \end{cases} \quad (1)$$

With P^{D^b} defined analogously, the dot product of the two cluster partitions C^b and D^b is defined as $D(P^{C^b}, P^{D^b}) = \sum_{i,j} P_{ij}^{C^b} P_{ij}^{D^b}$. This leads to a Jaccard coefficient based index of two cluster partitions C^b and D^b

$$Stab_{\mathrm{J}}(P^{C^b}, P^{D^b}) = \frac{D(P^{C^b}, P^{D^b})}{D(P^{C^b}, P^{C^b}) + D(P^{D^b}, P^{D^b}) - D(P^{C^b}, P^{D^b})}. \quad (2)$$

Hennig proposed a so-called local stability measure for every cluster group in a cluster partition based on the Jaccard coefficient as well [15]. To obtain one stability value $Stab_{\mathrm{J;cw}}$ for the whole partition, the weighted mean of the cluster-wise values with respect to the size of the cluster groups is determined. Another stability-based index presented by Ben-Hur et al., based on the simple matching coefficient, is called Rand index [16] and defined as

$$Stab_{\mathrm{R}}(P^{C^b}, P^{D^b}) = 1 - \frac{1}{n^2} \| P^{C^b} - P^{D^b} \|^2. \quad (3)$$

Additionally, they present the stability index based on a similarity measure, which was originally mentioned by Fowlkes and Mallows [17],

$$Stab_{\mathrm{FM}}(P^{C^b}, P^{D^b}) = \frac{D(P^{C^b}, P^{D^b})}{\sqrt{D(P^{C^b}, P^{C^b}) D(P^{D^b}, P^{D^b})}}. \quad (4)$$

For determination of the number of clusters, Ben-Hur et al. proposed the analysis of the distribution of index values calculated between pairs of clustered sub-samples, where high pairwise similarities indicate a stable partition. The authors' suggested aim is examining the transition from a stable to an unstable clustering state. In the simulation study, this qualitative criterion was numerically approximated by the differences in the areas under these curves. Furthermore, von Luxburg [18] published an approach to obtain the cluster partition stability based on the minimal matching distance, where the minimum is taken over all permutations of the K labels of clusters. Straightforward, the distances are summarized by their mean to obtain $Instab_{\mathrm{L}}(P^{C^b}, P^{D^b})$ respectively $Stab_{\mathrm{L}}(P^{C^b}, P^{D^b}) = 1 - Instab_{\mathrm{L}}(P^{C^b}, P^{D^b})$.

3 Simulation Study

In order to compare the stability indices of the cluster partition and afterwards with respect to the internal validation indices, a simulation study was conducted. In the following, the setup and execution of this simulation study starting with the data generation is briefly presented, and subsequently the results are evaluated.

3.1 Data Generation and Execution of Simulation Study

The simulation study is based on artificial data, which are generated for different scenarios. In Table 1, the features that define the data scenarios and their corresponding parameter values are listed. Since a full factorial design is used, there are 120 different data settings in the conducted simulation study.[2] The selection of the considered features follow the characteristics of the simulation study in [19] and were extended with respect to the ratio of the variable types as in [20].

Table 1 Features and the associated feature specifications used to generate the data scenarios.

data parameter	feature specification	short
number of clusters	2, 4, 8	`nC`
clusters of equal size (`FALSE`: randomly drawn sizes)	`TRUE`, `FALSE`	`symm`
number of variables	2, 4, 8	`nV`
ratio of factor to numerical variables	0.25, 0.5, 0.75	`fac_prop`
overlap between cluster groups	0, 0.05, 0.1	`overlap`

The clusters of the 200 observations are defined by the the feature settings. Each variable can either be *active* or *inactive*. For the numerical variables, *active* means drawing values from the normal distribution $X_1 \sim \mathcal{N}(\mu_1, 1)$, with random $\mu_1 \in \{0, ..., 20\}$, and *inactive* means drawing from $X_0 \sim \mathcal{N}(\mu_0, 1)$ with $\mu_0 = 2 \cdot q_{1-\frac{v}{2}} - \mu_1$, where q_α is the α-quantile of $\mathcal{N}(\mu_1, 1)$ and $v \in \{0.05, 0.1\}$. This results in an overlap of v for the two normal distributions. To achieve an overlap of $v = 0$, the inactive variable is drawn from $\mathcal{N}(\mu_1 - 10, 1)$. Furthermore, each factor variable has two levels, l_0 and l_1. The probability for drawing l_0 for an active variable is v and $(1 - v)$ for level l_1. For an inactive variable, the probability for l_0 is $(1 - v)$ and v for l_1.

Below, the code structure of the simulation study is presented. For each of the 120 data scenarios, a repetition of $N = 10$ runs was performed. This should mitigate the influence of the random initialization of the *k-prototypes* algorithm. For the range of two up to nine cluster groups, the stability indices are determined based on bootstrap samples as suggested in [21]. In order to rank the performance of the stability-based indices, the internal validation indices were also determined on the same data.

Pseudo-Code Simulation Study

```
for(every data situation){
  for(i in 1:N){ # 10 iterations to mitigate/soften random influences
    data <- create.data(data situation)
    for(q in 2:9){
      output <- kproto(data, k = q, nstarts = 20)
      # stability-based indices determined with the usage of 100 bootstrap samples
      stab_val_method <- stab_kproto(output, B = 100, method)
      int_val_method <- validation_kproto(output, method) # internal validation
    }
    # determine optimal cluster size for every method
    cs_method <- max/min(int_val_method or stab_val_method)
  }
}
```

[2] There is no data scenario with two variables and eight cluster groups. Additionally, if there are two variables, obviously only the 0.5 ratio between factor and numerical variables is possible.

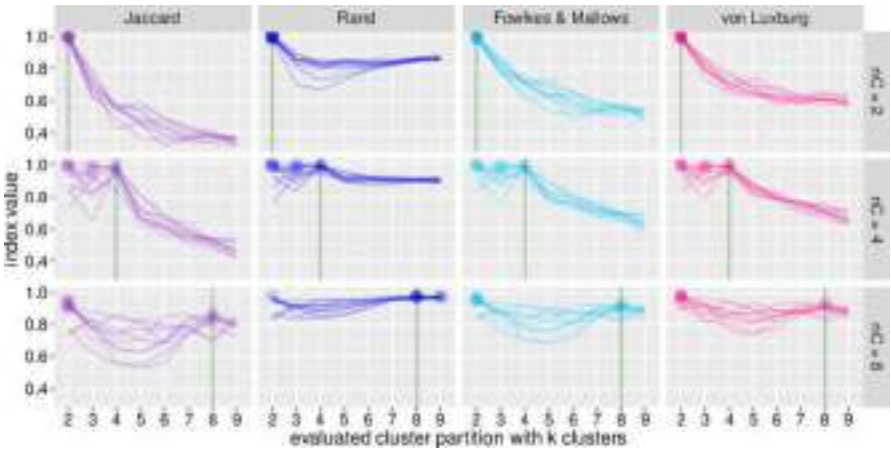


Fig. 1 The evaluations of the four stability-based cluster indices are presented. There are ten repetitions of rating the data situation for k clusters in the range of two to nine and the index-optimal number of clusters is highlighted. The parameters of the underlying data structure are `nV = 8`, `fac_prop = 0.5`, `overlap = 0.1` and `symm = FALSE`. The number of clusters `nC` in the data structure varies row-wise.

3.2 Analysis of the Results

Figure 1 shows exemplary results of the simulation study for three different data scenarios over the 10 repetitions. Each row of the figure shows a different data scenario and each column shows one of the four stability-based indices. The first row is related to a data scenario with two clusters (marked by a vertical green line). Each plot shows the examined number of clusters and the determined index value for the 10 repetitions. The maximum index value for each repetition is highlighted with a larger dot and marks the index-optimal number of clusters of this repetition. It can be seen that all of the four different indices detected the two clusters in the underlying data structure. Rows two and three show the evaluations of data with cluster partitions of four and eight clusters, respectively. It can be seen that the generated number of clusters is not always rated as index optimal (for example, with four clusters, two or three clusters were often also evaluated as optimal). Since the results shown here are representative for all scenarios, the four cluster indices and their interpretation were examined in more detail.

In the left part of Figure 2, different transformations of the index values are presented. Besides the standard index values (green line), the numerical approximation of the approach of Ben-Hur et al. mentioned above is also shown (red line). For the Jaccard-based evaluation, the proposed cluster-wise stability determination by Hennig is presented in orange. Additionally, we propose an adjustment of the index values (hereinafter referred to as *new adjust*), similar to [22], to take into account not only the magnitude of the index but also the local slope: The index value scaled with the geometric mean of the changes to the neighbor values is presented in dark green.

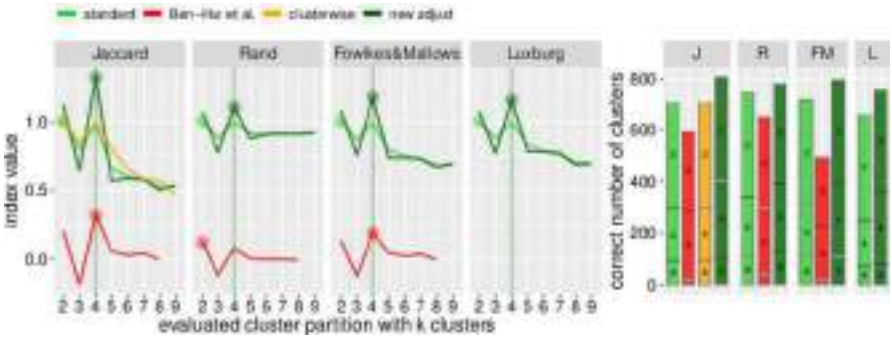


Fig. 2 *Left:* Example of the variations of the index values at an iteration of the data scenario with the parameters `nC = 4`, `nV = 8`, `fac_prop = 0.5`, `overlap = 0.1` and `symm = FALSE`. *Right:* Proportion of correct determinations, partitioned according to the different number of clusters in the underlying data structure.

Again, for each variation of the indices, the index optimal value is highlighted. The numerically determined index values according to the approach of Ben-Hur et al. gain no benefit, thus it can be concluded that the quantification is not appropriate for the purpose and that further research is required. The cluster-wise stability determination of the Jaccard index also does not seem to improve the determination of the number of clusters to a large extent. Obviously, the local slope in the example in Figure 2 is strengthened for four evaluated cluster groups by the new adjustment that leads to a determination of four cluster groups (which is the generated number of clusters). Since only one iteration of one data scenario is shown on the left, the sum of correct determined number of clusters with respect to the generated number of clusters is shown on the right hand side of Figure 2. These sums for two, four and eight clusters in the underlying data structure point out the improvement of the proposed adjustment of the index values. Especially for more than two clusters, the rate of correctly determined numbers of clusters can be increased.

Finally, the internal validation indices were comparatively examined. For analyzing the outcome of the simulation study, the determined index optimal numbers of clusters are shown in Table 2. While the comparison for two clusters in the underlying data shows a slight advantage for the stability-based indices, especially for eight clusters the preference is in favor of the internal validation indices. To gain a better understanding of the mean success rate of determining the correct number of clusters for each data scenario, Figure 3 further shows the results of a linear regression on the various data parameters. It can be seen that in most cases there is not too much difference between the considered methods. The stability-based indices do a better job of determining the number of clusters for data with equally large cluster groups. Obviously, a larger number of variables causes a better determination of the number of clusters. The largest variation in the influence on the proportion of correct determination can be seen for the parameter *number of clusters*. The more cluster groups are available in the underlying data structure, the worse the determination becomes (especially for the stability-based indices and the indices Ptbiserial and Tau).

Table 2 Determined number of clusters for all data scenarios with $\mathtt{nC} \in \{2, 4, 8\}$, summarized by the stability-based as well as internal validation indices and the evaluated number of clusters.

clusters	**2**	3	4	5	6	7	8	9	2	3	**4**	5	6	7	8	9	2	3	4	5	6	7	**8**	9
J_{newadj}	**403**	17	0	0	0	0	0	0	47	74	**298**	1	0	0	0	0	90	70	27	16	16	26	**104**	11
R_{newadj}	**391**	18	5	0	1	1	1	3	56	99	**258**	3	2	0	0	2	38	68	22	17	16	32	**133**	34
FM_{newadj}	**402**	17	1	0	0	0	0	0	50	80	**289**	1	0	0	0	0	88	71	26	16	15	26	**106**	12
L_{newadj}	**394**	21	5	0	0	0	0	0	53	83	**282**	2	0	0	0	0	100	97	31	20	16	16	**76**	4
CIndex	**313**	13	2	2	1	4	18	67	7	27	**344**	13	3	2	5	19	2	0	2	4	22	28	**211**	91
Dunn	**386**	24	4	2	0	1	1	2	39	56	**307**	8	7	3	0	0	19	9	17	7	37	53	**190**	28
Gamma	**343**	9	1	0	1	2	14	50	9	16	**356**	15	3	1	5	15	2	1	4	4	16	16	**198**	119
GPlus	**319**	8	1	0	0	0	9	83	6	10	**319**	12	5	2	15	51	2	1	1	4	14	12	**175**	151
McClain	**71**	3	1	1	5	12	57	270	0	0	**17**	4	4	13	87	295	0	0	0	0	0	9	**34**	317
Ptbiserial	**400**	11	6	0	3	0	0	0	72	120	**225**	3	0	0	0	0	31	62	79	65	55	39	**26**	3
Silhouette	**388**	3	1	4	4	5	8	7	14	37	**348**	7	0	0	8	6	6	0	3	1	12	46	**220**	72
Tau	**391**	16	9	0	4	0	0	0	68	144	**205**	3	0	0	0	0	33	82	119	68	40	14	**3**	1

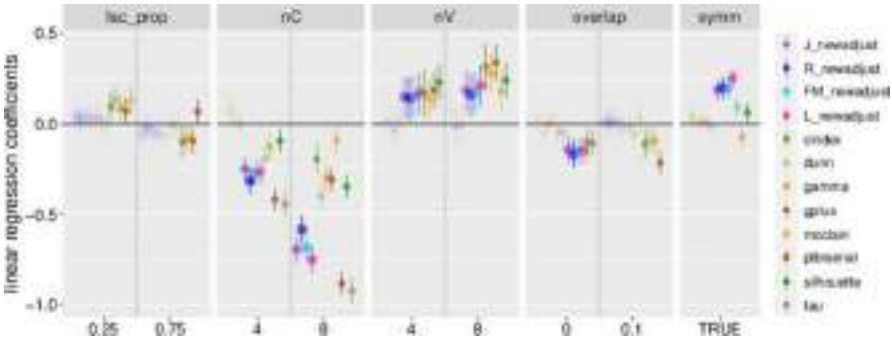


Fig. 3 Linear regression coefficients for the parameters of the five data set features, where coefficients whose confidence intervals contain 0 are displayed in transparent.

4 Conclusion

The aim of this study was to investigate the determination of the optimal number of clusters based on stability indices. Several variations of analysis methods of stability-based index values were presented and comparatively analyzed in a simulation study. The proposed adjustment of the index values with respect not only to their magnitude but also to the local slope was able to improve the standard stability indices, especially for a smaller number of clusters. The simulation study did not show any general superiority of stability-based approaches over internal validation indices.

In the future, the various methods of analyzing the stability-based index values should be examined in more detail, e.g., taking into account the Adjusted Rand Index. For this purpose, further research may address the characteristics of the evaluated curves more precisely, or further extend the approach of Ben-Hur et al. as a quantitative determination method, which has not been done yet.

References

1. Ahmad, A., Khan, S.: Survey of state-of-the-art mixed data clustering algorithms. IEEE Access, 31883–31902 (2019)
2. Huang, Z.: Extension to the k-Means algorithm for clustering large data sets with categorical values. Data Min. Knowl. Discov. **2**(6), 283–304 (1998)
3. Szepannek, G.: clustMixType: User-friendly clustering of mixed-type data in R. The R J. **10**(2), 200–208 (2018)
4. Szepannek, G., Aschenbruck, R.: clustMixType: k-prototypes clustering for mixed variable-type data. R package version 0.2-15 (2021) `https://CRAN.R-project.org/package=clusterMixType`
5. Thorndike, R. L.: Who belongs in the family. Psychometrika **18**(4), 267–276 (1953)
6. Hennig, C.: Clustering strategy and method selection. In: Hennig, C., Meila, M. , Murtagh, F., Rocci, R. (eds.) Handbook of Cluster Analysis, pp. 703–730. Chapman and Hall/CRC, New York (2015)
7. Halkidi, M., Vazirgiannia, M., Hennig, C.: Method-independent indices for cluster validation and estimating the number of clusters. In: Hennig, C., Meila, M. , Murtagh, F., Rocci, R. (eds.) Handbook of Cluster Analysis, pp. 595–618. Chapman and Hall/CRC, New York (2015)
8. Desgraupes, B.: clusterCrit: clustering indices. R package version 1.2.8 (2018) `https://CRAN.R-project.org/package=clusterCrit`
9. Aschenbruck, R., Szepannek, G.: Cluster validation for mixed-type data. Arch. Data Sci., Ser. A **6**(1), 1–12 (2020)
10. Lange, T., Roth, V., Braun, M. L., Buhmann, J. M.: Stability-based validation of clustering solutions. Neural. Comput. **16**(6), 1299–1323 (2004)
11. Dolnicar, S., Leisch, F.: Evaluation of structure and reproducibility of cluster solutions using bootstrap. Mark. Lett. **21**, 83–101 (2010)
12. Fang, Y., Wang, J.: Selection of the number of clusters via the bootstrap method. Comput. Stat. Data Anal. **56**(3), 468–477 (2012)
13. Mucha, H.-J., Bartel, H.-G.: Validation of k-means clustering: why is bootstrapping better than subsampling. Arch. Data Sci., Ser. A **2**(1), 1–14 (2017)
14. Ben-Hur, A., Elisseeff, A., Guyon, I.: A stability based method for discovering structure in clustered data. In: Pac. Symp. Biocomput. **2002**, 6–17 (2001)
15. Hennig, C.: Cluster-wise assessment of cluster stability. Comput. Stat. Data Anal. **52**(1), 258–271 (2007)
16. Rand, W. M.: Objective criteria for the evaluation of clustering methods. J. Am. Stat. Assoc. **66**(336) 846–850 (1971)
17. Fowlkes, E. B., Mallows, C. L.: A method for comparing two hierarchical clusterings. J. Am. Stat. Assoc. **78**(383) 553–569 (1983)
18. von Luxburg, U.: Clustering stability: an overview. Found. Trends® Mach. Learn. **2**(3), 235–274 (2010)
19. Dangl, R., Leisch, F.: Effects of resampling in determining the number of clusters in a data set. J. Classif. **37**(3), 558–583 (2020)
20. Jimeno, J., Roy, M., Tortora, C.: Clustering mixed-type data: a benchmark study on KAMILA and k-prototypes. In: Chadjipadelis, T., Lausen, B., Markos, A., Lee, T.R., Montanari, A., Nugent, R. (eds.) Data Analysis and Rationality in a Complex World, 83–91, Springer International Publishing, Cham (2021)
21. Leisch, F.: Resampling methods for exploring cluster stability. In: Hennig, C., Meila, M., Murtagh, F., Rocci, R. (eds.) Handbook of Cluster Analysis, pp. 637–652. Chapman and Hall/CRC, New York (2015)
22. Ilies, J., Wilhelm, A. F. X.: Projection-based partitioning for large, high-dimensional datasets. J. Comp. Graph. Stat. **19**(2), 474–492 (2010)

A Review on Official Survey Item Classification for Mixed-Mode Effects Adjustment

Afshin Ashofteh and Pedro Campos

Abstract The COVID-19 pandemic has had a direct impact on the development, production, and dissemination of official statistics. This situation led National Statistics Institutes (NSIs) to make methodological and practical choices for survey collection without the need for the direct contact of interviewing staff (i.e. remote survey data collection). Mixing telephone interviews (CATI) and computer-assisted web interviewing (CAWI) with direct contact of interviewing constitute a new way for data collection at the time COVID-19 crisis. This paper presents a literature review to summarize the role of statistical classification and design weights to control coverage errors and non-response bias in mixed-mode questionnaire design. We identified 289 research articles with a computerized search over two databases, Scopus and Web of Science. It was found that, although employing mixed-mode surveys could be considered as a substitution of traditional face-to-face interviews (CAPI), proper statistical classification of survey items and responders is important to control the nonresponse rates and coverage error risk.

Keywords: mixed-mode official surveys, item classification, weighting methods, clustering, measurement error

Afshin Ashofteh (✉)
Statistics Portugal (Instituto Nacional de Estatística, Departamento de Metodologia e Sistemas de Infomação) and NOVA Information Management School (NOVA IMS) and MagIC, Universidade Nova de Lisboa, Lisboa, Portugal, e-mail: afshin.ashofteh@ine.pt

Pedro Campos
Statistics Portugal (Instituto Nacional de Estatística, Departamento de Metodologia e Sistemas de Infomação) and Faculty of Economics, Universidade do Porto, and LIAAD INESC TEC, Portugal, e-mail: pedro.campos@ine.pt

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_7

1 Introduction

This paper provides a summary of a systematic literature review of the role of classification variables and weighting methods of mixed-mode surveys in minimizing the measurement error, coverage error, and nonresponse bias.

Before the COVID-19 pandemic, the statistical adjustment of mode-specific measurement effects was studied by many scholars. However, after the pandemic, survey methodologists made a strong effort to meet the challenges of new restrictions for collecting data with proper quality [1]. Data collection with mixing different modes by considering their contribution to the overall published statistics was considered as a solution by NSIs. The methodologists have been trying to use technology, data science, and mixed-device surveys to decrease the expected coverage error and nonresponse bias with new target populations at the time of pandemic rather than the traditional interviewer-assisted and paper survey modes [2]. This coverage error is caused by the changes of the target population from the general population to the general population accessible with technological devices. Te Braak et. al. [3] highlighted how the representativeness of self-administered online surveys is expected to be impacted by decreased response rates. Their research demonstrates that a huge group of respondents dropout selectively and that this selectivity varies depending on the dropout moment and demographic categorical information.

According to the studies in Statistics Portugal, using classification methods by categorical variables and applying the repeated weighting techniques seem to be fruitful to estimate and adjust for mode and device effects. Fortunately, many authors discussed the use of weights in statistical analysis [4]. It is important to improve inference in cases where mixed-mode effects are combined with measurement errors caused by primary data collection on categorical variables and socio-demographic information. On one side that the categorical variables are collected with the help of responders (primary data), the survey mode has a strong impact on answering behaviors and answering conditions. Respondents might evaluate some of the new categorical variables as sensitive information or privacy intrusive. They may not be willing to share these personal data by telephone or technological devices, which are necessary for statistical classification. Additionally, for NSIs, also the new data collection channels are costly and redesign of the survey estimation methodology is time consuming. On the other side, the categorical variables should be available in sampling frames (secondary data) and the coverage error is the main concern. For instance, in CATI surveys of Statistics Portugal after COVID-19, the population was considered as belonging to the following categories: (i) households with a listed landline telephone, (ii) households that do not have a telephone but use only a mobile telephone, and (iii) households that do not have a telephone at all (or whose number is unknown). We could expect these households with very different socioeconomic characteristics, and new methods of classification or clustering as helpful methods for measurement error adjustment at the time of the pandemic. However, if they are different in the important categorical variables of our survey, then a weighting solution could amplify a part of the sample, which does not represent the population. As a result, statistical classification would be another source of bias instead of

solving the problem. Therefore, we could expect two approaches. First, we could ignore classification, simply because we consider the groups are homogeneous and the weighting could be recommended to adjust for COVID-19 pandemic situation and non-observation errors. Second, the groups or responders are different and we need categorical variables. In this case, the non-observation errors of CATI and CAWI could not be covered by changing only the weights and we have to recommend CAPI to collect categorical information and apply both clustering and weighting together to have a reasonable coverage by mixed modes.

This study undergoes a systematic literature review on this topic guided by the following question. What is the best methodology or modified estimation strategy to mitigate the mode-effects problems based on design weighting and classification? To answer this question, we performed a systematic review analysis limited to the following databases: Web of Science, Scopus, and working papers from NSIs. We only considered papers written in English. This article is organized as follows: Section 2 presents the methodology of research that maps keyword identification search, databases, and bibliometric analysis. In Section 3, we present the results, identifying the PRISMA flow diagram, characteristics of the articles, author co-authorship analysis, as well as the Keywords occurrence over the years. In Section 4, we discuss the content analysis. Section 5 is about the main conclusions and finally, in Section 6, the main research gaps and future works are outlined.

2 Methods

To accomplish the research, the preferred reporting items for systematic reviews and meta-analysis methodology were adopted. The algorithm of the paper selection from databases (Scopus and WOS) was based on screening started by search keywords ((mixed-mode* OR "Mode effect*") AND (weighting OR weight* OR classification) AND ("Measurement error*" OR "Non-response bias" OR "Data quality" OR "response rate*") AND (capi OR "Computer Assisted Personal Interview*" OR cawi OR "Assisted Web Interview*" OR cati OR "Computer Assisted Telephone Interview*" OR "web survey*" OR "mail survey*" OR "telephone survey*")) and then the result was filtered by “official statistics”. The results of the two databases were merged, and then duplication was removed. For bibliometric analysis, the Mendeley open-source tool was used to extract metadata and eliminate duplicates. For network analysis, the VOSviewer open-source tool has been applied to visualize the extracted information from the data set and obtain the quantitative and qualitative outcomes. After assessing the eligibility, books and review papers were omitted from results and relevant articles picked up from databases. The final dataset was selected according to the visual abstract in Figure 2, which shows detailed information about this systematic literature review.

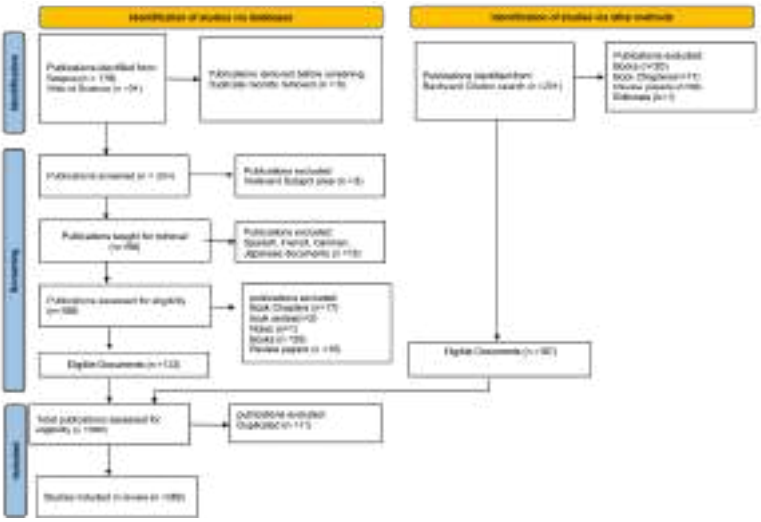


Fig. 1 Literature review flow diagram. (Source: Author's preparation).

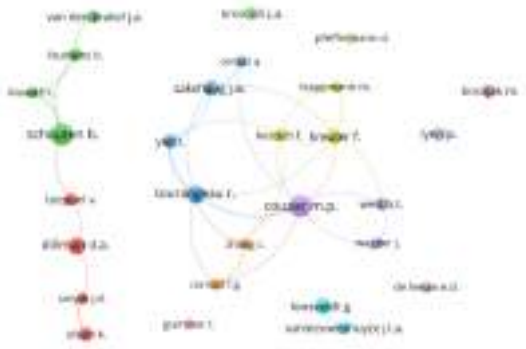


Fig. 2 Density visualization analysis of the 22 leader authors who have at least 3 papers.

3 Results

The 28 leader authors who had at least 4 papers are presented in Figure 2. Author occurrence analysis was performed by applying the VOSviewer research tool for network analysis. The top three leader authors were Mick P. Couper with 14 articles, Barry Schouten with 14 articles, and Roger Tourangeau with 11 articles. With the help of VOSviewer, keywords' analysis was accomplished. We analyzed the co-occurrence of author keywords with the full counting method. In the first step, we select one for the minimum occurrence of a keyword and the result was 711 keywords. We could see the application of keywords over years (Figure 3). Some of the keywords were not exactly the same, but their use and meaning were the same.

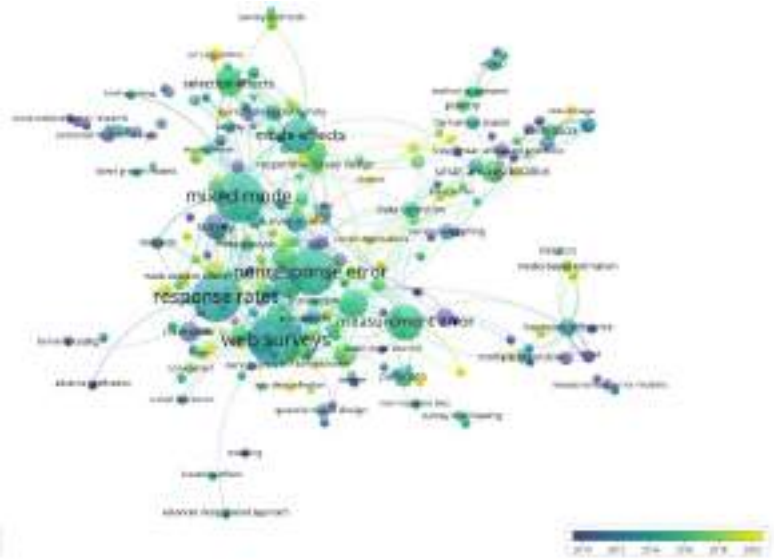


Fig. 3 Application of keywords over years.

We decided to match similar words to make the output clearer. Choosing the full counting method resulted in a total of 592 authors meeting the threshold.

4 Content Analysis

The studies emphasize the dramatic change in mixed-mode strategies in the last decades based on design-based and model-assisted survey sampling, time series methods, small area estimation [6], and high expectation to undergo further changes especially after the magnificent experience of NSIs, trying new modes after COVID-19 pandemic [7].

The problem is about mixed-mode effects and calibration, and briefly, we could follow several approaches such as design weighting to find sampling weights, nonresponse weighting adjustment, and calibration. The design weight of a unit may be interpreted as the number of units from population represented by a specific sample unit. Most surveys, if not all, suffer from nonresponse in item or unit. Auxiliary information could be used to improve the quality of design-weighted estimates. An auxiliary variable must have at least two characteristics to be considered in calibration: (i) It must be available for all sample units; and (ii) Its population total must be known.

The categorical variables from the demographic information of nonrespondents such as education level, age, income, location, language, and marital status could help the survey methodologists to categorize the target population and recognize

the best sequence of the modes [8]. Van Berkel et al. [9] considered nine strata in their classification tree by using age, ethnicity, urbanization, and income as explanatory variables. Re-interview design and inverse regression estimator (IREG) are among the best approaches to improve measurement bias by using related auxiliary information [10].

The focus of this approach is on the weights of estimators rather than the bias from the measurements. For an estimator, we could consider $y_{i,m}$ the measurement obtained from unit i through mode m. The $y_{i,m}$ consists of u_i as the observed value for respondent i, an additive mode-dependent measurement bias b_m, and a mode-dependent measurement variance $\varepsilon_{i,m}$ with an expected value equal to zero. Equation (1) shows the measurement error model.

$$y_{i,m} = u_i + b_m + \varepsilon_{i,m} \tag{1}$$

If we consider two different modes m and $\dot{m}$, then the differential measurement error between these two modes is given by

$$y_{i,m} - y_{i,\dot{m}} = (b_m - b_{\dot{m}}) + \left(\varepsilon_{i,m} - \varepsilon_{i,\dot{m}}\right) \tag{2}$$

The expected value of $(b_m - b_{\dot{m}})$ is the differential measurement bias. If we consider $\hat{t}_y$ as an estimation of the total of variable y according to its observations in different modes $y_{i,m}$, then

$$\hat{t}_y = \sum_{i=1}^{n} \omega_i y_{i,m} \tag{3}$$

where ω_i is a survey weight assigned to unit i with n the number of respondents. From a combination of equations (2) and (3), and taking the expectation over the measurement error model (1), we would have

$$E\left(\hat{t}_y\right) = E\left(\sum_{i=1}^{n} \omega_i y_{i,m}\right) = \sum_{i=1}^{n} \omega_i u_{i,m} + \sum_{i=1}^{n} b_m \omega_i \partial_{i,m} + \sum_{i=1}^{n} \omega_i \partial_{i,m} E\left(\varepsilon_{i,m}\right) \tag{4}$$

with $\partial_{i,m} = 1$ if unit i responded through mode m, and zero otherwise. Since $E\left(\varepsilon_{i,m}\right) = 0$

$$E\left(\hat{t}_y\right) = E\left(\sum_{i=1}^{n} \omega_i y_{i,m}\right) = \sum_{i=1}^{n} \omega_i u_{i,m} + \sum_{i=1}^{n} \omega_i \partial_{i,m} b_m \tag{5}$$

stating that the expected total of the survey estimate for Y consists of the estimated true total of U, plus true total of b_m from data collected through mode m. Since b_m is an unobserved mode-dependent measurement bias, $\sum_{i=1}^{n} \omega_i \partial_{i,m} b_m$ in equation (5) indicates the existence of an unknown mode-dependent bias for estimation of t_y. According to Equation (5), there is an unknown measurement bias in sequential mixed-mode designs that might be adjusted by different estimators. Data obtained

via a re-interview design or a sub-set of respondents to the first stage of a sequential mixed-mode survey provides necessary auxiliary information to adjust measurement bias in sequential mixed-mode surveys. Klausch et al [10] propose six different estimators and show that an inverse version of regression estimator (IREG) performs well under all considered scenarios. The idea of IREG is to use re-interview data to estimate the inverse slope of ordinary or generalized least squares linear regression of benchmark measurements y^{m_b} on y^{m_j} as follows [11]

$$y_i^{m_j} = \hat{\beta}_0 + \hat{\beta}_1 y_i^{m_b} \tag{6}$$

and estimate the measurement of target variable by applying the inverse of $\hat{\beta}_1$ in the following estimator, so-called inverse regression estimator

$$\hat{y}_{r_{mm}}^{ireg} = \frac{1}{(\hat{N}_{m_1} + \hat{N}_{m_2})} \left(\sum_{i=1}^{n_{m_b}} d_i y_i^{m_b} + \sum_{i=1}^{m_j} d_i \left(\hat{y}_{re}^{m_b} - \frac{1}{\hat{\beta}_1} \left(\hat{y}_{re}^{m_j} - y_i^{m_j} \right) \right) \right) b, j = 1, 2; b \neq j \tag{7}$$

where $\hat{y}_{re}^{m_j}$ and $\hat{y}_{re}^{m_b}$ are the respondents means of focal and benchmark mode outcome in the re-interview and d_i denotes the design weight of the sample design. For a detailed presentation and discussion of the methods see Chapter 8.5 in [12]. However, for longitudinal studies with different modes at different time points, the effect of time on the respondents would make it difficult to estimate the pure mixed-mode effect especially for volatile classification variables such as the address for immigrants. The solution could be conducting the survey on parallel or separate samples to evaluate the time effect and mode effect separately.

In practice, Statistics Portugal has been using the available information of a sampling frame as a part of FNA (the dwellings national register database) at the time of COVID-19. The situation was considered as telephone numbers are linked to a sample drawn from a population register in FNA for the samples for CATI rotation-scheme surveys such as Labor Force Survey. In 2020, the Labour Force Survey (LFS) in Portugal as a mandatory survey for the member states within the EU was adjusted for undercover of the percentage of households with a listed landline telephone. As a result, the comparison of these surveys after and before COVID-19 shows the usefulness of the discussed methodologies. In 2021, the successful CAWI mode census by Statistics Portugal shows respondents tend to favor the web-based questionnaire to avoid the risk of COVID-19 infection with a face-to-face interview. It shows the potential change in the mode tendency by responders.

5 Conclusions

COVID-19 crisis led to new solutions on item classification for mixed-mode effects adjustment, such as applying mode calibration to population subgroups by categorical variables such as gender, regions, age groups, etc. Studies offer sequential mixed-mode design started with CAWI as the cheapest mode supported by an initial

postal mail or telephone contact and possible cash incentive. With a lag, follow up the non-respondents with giving them a choice between CAPI and CATI according to their specific classification group and demographic information, such as education level, age, income, location, language, and marital status. It is fruitful to reduce the cost and increase the accuracy simultaneously.

This study showed that sample frames might need updates for necessary categorical information, which are based on choices made several years ago. Additionally, more research studies seem necessary for ethics concerns, privacy regulations, and standards for using categorical variables and classification information in social mixed-mode surveys and official statistics.

References

1. Ashofteh, A., Bravo, J. M.: A study on the quality of novel coronavirus (COVID-19) official datasets. Stat. J. IAOS, **36**(2), 291–301, (2020) doi: 10.3233/SJI-200674
2. Ashofteh, A., Bravo, J. M.: Data science training for official statistics: A new scientific paradigm of information and knowledge development in national statistical systems. Stat. J. IAOS, **37**(3), 771–789, (2021) doi: 10.3233/SJI-200674
3. Te Braak, P., Minnen, J., Glorieux, I.: The representativeness of online time use surveys. Effects of individual time use patterns and survey design on the timing of survey dropout. J. Off. Stat., **36**(4), 887–906, (2020)
4. Szymkowiak, M., Wilak, K.: Repeated weighting in mixed-mode censuses. Econ. Bus. Rev., **7**(1), 26–46, (2021)
5. Zax, M., Takahashi, S.: Cultural influences on response style: comparisons of Japanese and American college students. J. Soc. Psychol., **71**(1), 3–10, (1967)
6. Pfeffermann, D.: New important developments in small area estimation. Stat. Sci., 28(1), 40–68, (2013)
7. Toepoel, V., de Leeuw, E., Hox, J.: Single- and Mixed-Mode Survey Data Collection. SAGE Res. Methods Found, (2020) doi: 10.4135/9781526421036876933
8. Kim, S., Couper, M. P.: Feasibility and quality of a national RDD smartphone web survey: comparison with a cell phone CATI survey. Soc. Sci. Comput. Rev., **39**(6), 1218–1236, (2021)
9. Van Berkel, K., Van Der Doef, S., Schouten, B.: Implementing adaptive survey design with an application to the Dutch health survey. J. Off. Stat., **36**(3), 609–629, (2020) doi: 10.2478/jos-2020-0031
10. Klausch, T., Schouten, B., Buelens, B., van den Brakel, J.: Adjusting measurement bias in sequential mixed-mode surveys using re-interview data. J. Surv. Stat. Methodol., **5**(4), 409–432, (2017) doi: 10.1093/jssam/smx022
11. Särndal, C. E., Lundström, S.: Estimation in surveys with nonresponse. Estimation in surveys with nonresponse. John Wiley (2005) doi: 10.1002/0470011351
12. Schouten, B., Brakel, J. van den, Buelens, B., Giesen, D., Luiten, A., Meertens, V.: Mixed-Mode Official Surveys. Chapman and Hall/CRC (2021) doi: 10.1201/9780429461156

Clustering and Blockmodeling Temporal Networks – Two Indirect Approaches

Vladimir Batagelj

Abstract Two approaches to clustering and blockmodeling of temporal networks are presented: the first is based on an adaptation of the clustering of symbolic data described by modal values and the second is based on clustering with relational constraints. Different options for describing a temporal block model are discussed.

Keywords: social networks, network analysis, blockmodeling, symbolic data analysis, clustering with relational constraints

1 Temporal Networks

Temporal networks described by *temporal quantities* (TQs) were introduced in the paper [2]. We get a *temporal network* $\mathcal{N}_{\mathcal{T}} = (\mathcal{V}, \mathcal{L}, \mathcal{T}, \mathcal{P}, \mathcal{W})$ by attaching the *time* $\mathcal{T}$ to an ordinary network, where $\mathcal{V}$ is the set of nodes, $\mathcal{L}$ is the set of links, $\mathcal{P}$ is the set of node properties, $\mathcal{W}$ is the set of link weights, and $\mathcal{T} = [T_{min}, T_{max})$ is a linearly ordered set of time points $t \in \mathcal{T}$ which are usually integers or reals.

In a temporal network nodes/links activity/presence, nodes properties, and links weights can change through time. These changes are described with TQs. A TQ is described by a sequence $a = [(s_r, f_r, v_r) : r = 1, 2, \ldots, k]$ where $[s_r, f_r)$ determines a time interval and v_r is the value of the TQ a on this interval. The set $T_a = \bigcup_r [s_r, f_r)$ is called the *activity set* of a. For $t \notin T_a$ its value is *undefined*, $a(t) = \text{⌘}$.

Assuming that for every $x \in \mathbb{R} \cup \{\text{⌘}\} : x + \text{⌘} = \text{⌘} + x = x$ and $x \cdot \text{⌘} = \text{⌘} \cdot x = \text{⌘}$ we can extend the addition and multiplication to TQs

Vladimir Batagelj (✉)
IMFM, Jadranska 19, 1000 Ljubljana, Slovenia & IAM UP, Muzejski trg 2, 6000 Koper, Slovenia & HSE, 11 Pokrovsky Bulvar, 101000 Moscow, Russian Federation,
e-mail: vladimir.batagelj@fmf.uni-lj.si

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_8

$$(a+b)(t) = a(t) + b(t) \quad \text{and} \quad T_{a+b} = T_a \cup T_b$$
$$(a \cdot b)(t) = a(t) \cdot b(t) \quad \text{and} \quad T_{a \cdot b} = T_a \cap T_b$$

Let $T_V(v) \subseteq \mathcal{T}$, $T_V \in \mathcal{P}$, be the activity set for a node $v \in \mathcal{V}$ and $T_L(\ell) \subseteq \mathcal{T}$, $T_L \in \mathcal{W}$, the activity set for a link $\ell \in \mathcal{L}$. The following *consistency condition* must be fulfilled for activity sets: If a link $\ell(u,v)$ is active at the time point t then its end-nodes u and v should be active at the time point $t : T_L(\ell(u,v)) \subseteq T_V(u) \cap T_V(v)$.

In the following we will need

1. *Total*: $\text{total}(a) = \sum_i (f_i - s_i) \cdot v_i$
2. *Average*: $\text{average}(a) = \dfrac{\text{total}(a)}{|T_a|}$ where $|T_a| = \sum_i (f_i - s_i)$
3. *Maximum*: $\max(a) = \max_i v_i$

To support the computations with TQs we developed in Python the libraries TQ and Nets, see `https://github.com/bavla/TQ` .

2 Traditional (Generalized) Blockmodeling Scheme

A *blockmodel* (BM) [11] consists of structures obtained by identifying all units from the same cluster of the clustering / *partition* $\mathbf{C} = \{C_i\}$, $\pi(v) = i \Leftrightarrow v \in C_i$. Each pair of clusters (C_i, C_j) determines a block consisting of links linking C_i to C_j. For an exact definition of a blockmodel we have to be precise also about which blocks produce an arc in the *reduced graph* on classes and which do not, what is the *weight* of this arc, and in the case of generalized BM, of what *type*. The reduced graph can be represented by relational matrix, called also *image matrix*.

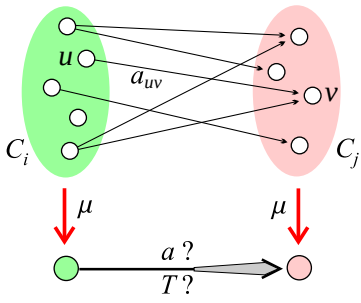


Fig. 1 Blockmodel.

To develop a BM method we specify a criterion function $P(\mu)$ measuring the "error" of the BM μ. We can introduce additional knowledge by constraining the partitions to a set Φ of feasible partitions. We are searching for a partition $\pi^* \in \Phi$ such that the corresponding BM μ^* minimizes the criterion function $P(\mu)$.

3 BM of Temporal Networks

For an early attempt of temporal network BM see [2, 5]. To the traditional BM scheme we add the time dimension. We assume that the network is described using temporal quantities [2] for nodes/links activity/presence, and some nodes properties and links weights. Then also the BM partition π is described for each node v with a

temporal quantity $\pi(v,t)$: $\pi(v,t) = i$ means that in time t node v belongs to cluster i. The structure and activity of clusters $C_i(t) = \{v : \pi(v,t) = i\}$ can change through time, but they preserve their identity.

For the BM μ the clusters are mapped into BM nodes $\mu : C_i \rightarrow [i]$. To determine the BM we still have to specify how the links from C_i to C_j are represented in the BM – in general, for the model arc $([i],[j])$, we have to specify two TQs: its *weight* a_{ij} and, in the case of generalized BM, its *type* τ_{ij}. The weight can be an object of a different type than the weights of the block links in the original temporal network.

We assume that in a temporal network $\mathcal{N} = (\mathcal{V}, \mathcal{L}, \mathcal{T}, \mathcal{P}, \mathcal{W})$ the links weight is described by a TQ $w \in \mathcal{W}$. In the following years we intend to develop BM methods case by case.

1. constant partition – nodes stay in the same cluster all the time:
   a. indirect approach based on clustering of TQs: $p(v) = \sum_{u \in N(v)} w(v,u)$, hierarchical clustering and leaders;
   b. indirect approach by conversion to the *clustering with relation constraint* (CRC);
   c. direct approach by (local) optimization of the criterion function P over Φ
2. dynamic partition – nodes can move between clusters through time. The details are still to be elaborated.

In this paper, we present approaches for cases 1.a and 1.b.

In the literature there exist other approaches to BM of temporal networks. A recent overview is available in the book [12].

3.1 Adapted Symbolic Clustering Methods

In [8] we adapted traditional leaders [13, 10] and agglomerative hierarchical [14, 1] clustering methods for clustering of modal-valued symbolic data. They can be almost directly applied for clustering units described by variables that have for their values temporal quantities.

For a unit X_i, each variable V_j is described with a size h_{ij} and a temporal quantity $\mathbf{x}_{ij}$, $X_{ij} = (h_{ij}, \mathbf{x}_{ij})$. In our algorithms we use *normalized* values of temporal variables $V' = (h, \mathbf{p})$ where

$$\mathbf{p} = [(s_r, f_r, p_r) : r = 1, 2, \ldots, k] \qquad \text{and} \qquad p_r = \frac{v_r}{h}$$

In the case, when $h = \text{total}(\mathbf{x})$, the normalized TQ $\mathbf{p}$ is essentially a probability distribution.

Both methods create cluster representatives that are represented in the same way.

3.2 Clustering of Temporal Network and CRC

To use the CRC in the construction of a nodes partition we have to define a dissimilarity measure $d(u,v)$ (or a similarity $s(u,v)$) between nodes. An obvious solution is $s(u,v) = f(w(u,v))$, for example

1. *Total activity*: $s_1(u,v) = \text{total}(w(u,v))$
2. *Average activity*: $s_2(u,v) = \text{average}(w(u,v))$
3. *Maximal activity*: $s_3(u,v) = \max(w(u,v))$

We can transform a similarity $s(u,v)$ into a dissimilarity by $d(u,v) = \frac{1}{s(u,v)}$ or $d(u,v) = S - s(u,v)$ where $S > \max_{u,v} s(u,v)$. In this way, we transformed the temporal network partitioning problem into a clustering with relational constraints problem [6, 360–369]. It can be efficiently solved also for large sparse networks.

3.3 Block Model

Having the partition π, to produce a BM we have to specify the values on its links. There are different options for model links weights $a(([i],[j]))$.

1. *Temporal quantities*: $a(([i],[j])) = \text{activity}(C_i, C_j) = \sum_{u \in C_i, v \in C_j} w(u,v)$, for $i \neq j$, and $a(([i],[i])) = \frac{1}{2}\text{activity}(C_i, C_i)$.
2. *Total intensities:* $a_t(([i],[j])) = \text{total}(a(([i],[j])))$.
3. *Geometric average intensities:* $a_g(([i],[j])) = \dfrac{a_t(([i],[j]))}{\sqrt{|C_i| \cdot |C_j|}}$.

4 Example: September 11th Reuters Terror News

The *Reuters Terror News* network was obtained from the CRA (Centering Resonance Analysis) networks produced by Steve Corman and Kevin Dooley at Arizona State University. The network is based on all the stories released during 66 consecutive days by the news agency Reuters concerning the September 11 attack on the U.S., beginning at 9:00 AM EST 9/11/01.

The nodes, $n = 13332$, of this network are important words (terms). For a given day, there is an edge between two words iff they appear in the same utterance (for details see the paper [9]). The network has $m = 243447$ edges. The weight of an edge is its daily frequency. There are no loops in the network. The network Terror News is undirected – so will be also its BM.

The Reuters Terror News network was used as a case network for the Viszards visualization session on the Sunbelt XXII International Sunbelt Social Network Conference, New Orleans, USA, 13-17. February 2002. It is available at `http://vlado.fmf.uni-lj.si/pub/networks/data/CRA/terror.htm` .

We transformed the Pajek version of the network into NetsJSON format used in libraries TQ and Nets. For a temporal description of each node/word for clustering we took its activity (sum of all TQs on edges adjacent to a given node v)

$$\mathrm{act}(v) = \sum_{u \in N(v)} w(v:u).$$

Our leaders' and hierarchical clustering methods are compatible – they are based on the same clustering error criterion function. Usually, the leaders' method is used to reduce a large clustering problem to up to some hundred units. With hierarchical clustering of the leaders of the obtained clusters, we afterward determine the "right" number of clusters and their representatives.

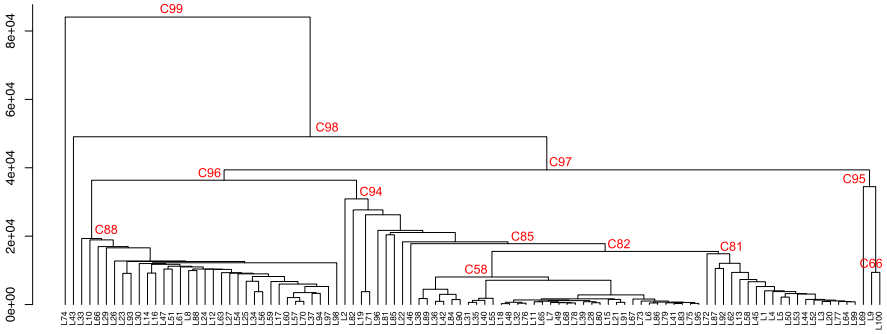


Fig. 2 Hierarchical clustering of 100 leaders in Terror News.

To cluster all 13332 words (nodes) in Terror News we used the adapted leaders' method searching for 100 clusters. We continued with the hierarchical clustering of the obtained 100 leaders. The result is presented in the dendrogram in Figure 2.

Fig. 3 Word clouds for clusters $C58$ and $C81$.

To get an insight into the content of a selected cluster we draw the corresponding word cloud based on the cluster's leader. In Figure 3 the word clouds for clusters $C58$ and $C81$ ($|C58| = 1396, |C81| = 2226$) are presented.

We can also compare the activities of pairs of clusters by considering the overlap of p-components (probability distributions) of their leaders. In Figure 4, we compare cluster $C58$ with cluster $C81$, and cluster $L96$ with cluster $C66$. In the right diagram some values are outside the display area: $L96[15] = 0.3524$, $C66[4] = 0.1961$, $C66[5] = 0.2917$.

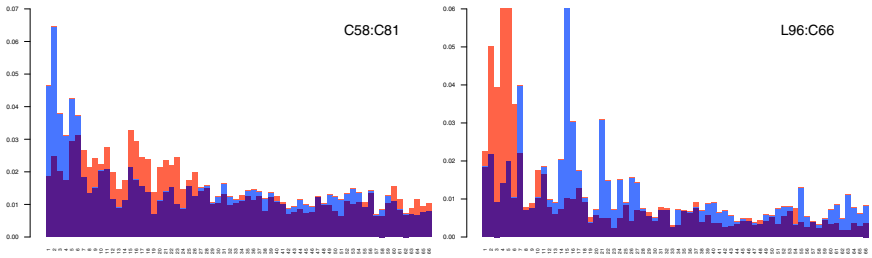
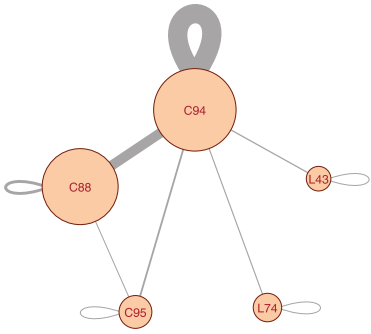


Fig. 4 Comparing activities of clusters (blue – first cluster, red – second cluster, violet – overlap).

We decided to consider in the BM the clustering of Terror News into 5 clusters $\mathbf{C} = \{C94, C88, C95, L43, L74\}$. The split of cluster C95 gives clusters of sizes 325 and 629 (for sizes, see the right side of Figure 5). Both clusters C94 and C88 have a chaining pattern at their top levels.

Because of large differences in the cluster sizes, it is difficult to interpret the total intensities image matrix. An overall insight into the BM structure we get from the geometric average intensities image matrix (right side) and the corresponding BM network (cut level 0.3), left side of Figure 5.



i	cluster	1	2	3	4	5
1	$C94$	23.85	12.23	2.26	1.57	1.42
2	$C88$		3.58	0.33	0.22	0.19
3	$C95$			0.56	0.07	0.07
4	$L43$				0.38	0.08
5	$L74$					0.39
	size	6018	5109	954	535	716

Fig. 5 Block model and image matrix.

A more detailed BM is presented by the activities (p-components) image matrix in Figure 6.

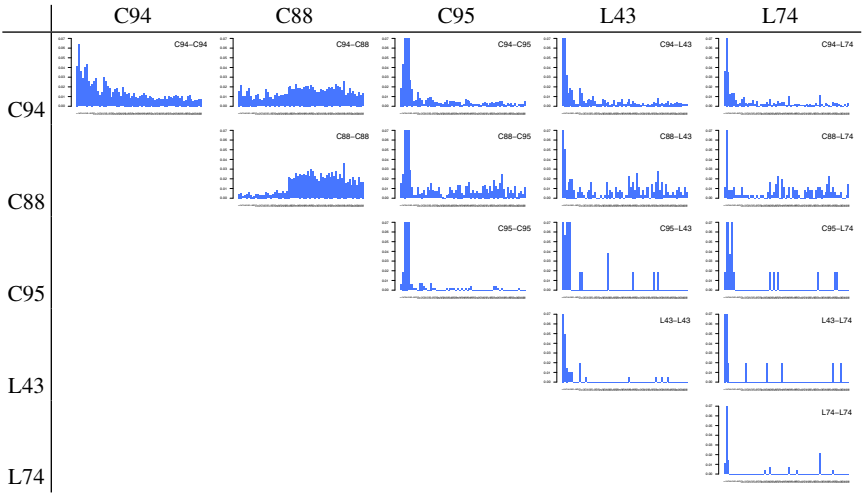


Fig. 6 BM represented as p-components of temporal activities of links between pairs of clusters.

A more compact representation of a temporal BM is a heatmap display of this matrix in Figure 7. Because of some relatively very large values, it turns out that the display of the matrix with logarithmic values provides much more information.

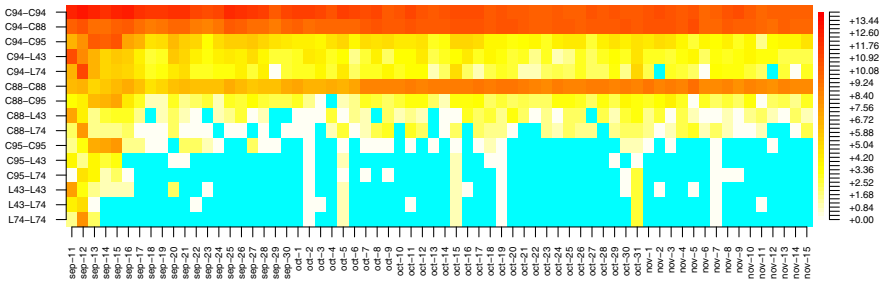


Fig. 7 BM heatmap with log2 values.

To the Terror News network, we applied also the clustering with relational constraints approach. Because of the limited space available for each paper, we can not present it here. A description of the analysis with the corresponding code is available at `https://github.com/bavla/TQ/wiki/BMRC` .

5 Conclusions

The presented research is a work in progress. It only deals with the two simplest cases of temporal blockmodeling. We provided some answers to the problem of normalization of model weights TQs when comparing them and some ways to present/display the temporal BMs.

We used different tools (R, Python, and Pajek) to obtain the results. We intend to provide the software support in a single tool – probably in Julia. We also intend to create a collection of interesting and well-documented temporal networks for testing and demonstrating the developed software.

Acknowledgements The paper contains an elaborated version of ideas presented in my talks at the XXXX Sunbelt Social Networks Conference (on Zoom), July 13-17, 2020 and at the EUSN 2021 – 5th European Conference on Social Networks, Naples (on Zoom), September 6-10, 2021.

This work is supported in part by the Slovenian Research Agency (research program P1-0294 and research projects J1-9187, J1-2481, and J5-2557), and prepared within the framework of the HSE University Basic Research Program.

References

1. Anderberg, M.R.: Cluster Analysis for Applications. Academic Press, New York (1973)
2. Batagelj, V., Praprotnik, S.: An algebraic approach to temporal network analysis based on temporal quantities. Soc. Netw. Anal. Min. **6**(1), 1–22 (2016)
3. Batagelj, V., Ferligoj, A.: Clustering relational data. In: Gaul, W., Opitz, O., Schader, M. (Eds.) Data Analysis / Scientific Modeling and Practical Application, pp. 3–15. Springer (2000)
4. Batagelj, V.: Generalized Ward and related clustering problems. In: Bock H.-H. (ed) Classification and Related Methods of Data Analysis, pp. 67–74. North-Holland, Amsterdam (1988)
5. Batagelj, V., Ferligoj, A., Doreian, P.: Indirect blockmodeling of 3-way networks. In: Brito, P., Bertrand, P., Cucumel, G., de Carvalho, F. (eds.) Selected Contributions in Data Analysis and Classification, pp. 151–159. Springer (2007)
6. Batagelj, V., Doreian, P., Ferligoj, A., Kejžar, N.: Understanding Large Temporal Networks and Spatial Networks: Exploration, Pattern Searching, Visualization and Network Evolution. Wiley (2014)
7. Batagelj, V., Kejžar, N.: Clamix – Clustering Symbolic Objects (2010) Program in R `https://r-forge.r-project.org/projects/clamix/`
8. Kejžar, N., Korenjak-Černe, S., Batagelj, V.: Clustering of modal-valued symbolic data. Adv. Data Anal. Classif. **15**, pp. 513—541 (2021)
9. Corman, S. R., Kuhn, T., McPhee, R. D., Dooley, K. J.: Studying complex discursive systems: Centering resonance analysis of communication. Hum. Commun. Res. **28**(2), 157–206 (2002)
10. Diday, E.: Optimisation en Classification Automatique. Tome 1.,2. INRIA, Rocquencourt (in French) (1979)
11. Doreian, P., Batagelj, V., Ferligoj, A.: Generalized Blockmodeling. Structural Analysis in the Social Sciences. Cambridge University Press (2005)
12. Doreian, P., Batagelj, V., Ferligoj, A. (Eds.) Advances in Network Clustering and Blockmodeling. Wiley (2020)
13. Hartigan, J. A.: Clustering Algorithms. Wiley-Interscience, New York (1975)
14. Ward, J. H.: Hierarchical grouping to optimize an objective function. J. Am. Stat. Asso.c **58**, 236–244 (1963)

Latent Block Regression Model

Rafika Boutalbi, Lazhar Labiod, and Mohamed Nadif

Abstract When dealing with high dimensional sparse data, such as in recommender systems, co-clustering turns out to be more beneficial than one-sided clustering, even if one is interested in clustering along one dimension only. Thereby, co-clusterwise is a natural extension of clusterwise. Unfortunately, all of the existing approaches do not consider covariates on both dimensions of a data matrix. In this paper, we propose a *Latent Block Regression Model* (LBRM) overcoming this limit. For inference, we propose an algorithm performing simultaneously co-clustering and regression where a linear regression model characterizes each block. Placing the estimate of the model parameters under the maximum likelihood approach, we derive a Variational Expectation-Maximization (`VEM`) algorithm for estimating the model's parameters. The finality of the proposed `VEM-LBRM` is illustrated through simulated datasets.

Keywords: co-clustering, clusterwise, tensor, data mining

1 Introduction

The *cluster-wise* linear regression algorithm CLR (or Latent Regression Model) is a finite mixture of regressions and one of the most commonly used methods for simultaneous learning and clustering [14, 5]. It aims to find clusters of entities to minimize the overall sum of squared errors from regressions performed over these clusters. Specifically, $\mathbf{X} = [x_{ij}] \in \mathbb{R}^{n \times v}$ is the covariate matrix and $\mathbf{Y} \in \mathbb{R}^{n \times 1}$ the response vector. The *cluster-wise* method aims to find g clusters $C_1, \ldots, C_g$ and regression coefficients $\boldsymbol{\beta}^{(k)} \in \mathbb{R}^{d \times 1}$ by minimizing the following objective function $\sum_{k=1}^{g} \sum_{i \in C_k} (y_i - \sum_{j=1}^{v} \beta_j^{(k)} x_{ij} + b_k)^2$ where:

- y_i is the value of the dependent variable for subject/observation i defined by $\mathbf{x}_i = (x_{i1}, \ldots, x_{id})$,
- x_{ij} is the value of the j-th independent variable for subject/observation i,
- $\beta_j^{(k)}$ is the j-th multiple regression coefficient and b_k is the *intercept*.

Rafika Boutalbi (✉)
Institute for Parallel and Distributed Systems, Analytic Computing, University of Stuttgart, Germany, e-mail: `rafika.boutalbi@ipvs.uni-stuttgart.de`

Lazhar Labiod · Mohamed Nadif
Centre Borelli UMR 9010, Université Paris Cité, France,
e-mail: `lazhar.labiod@u-paris.fr;mohamed.nadif@u-paris.fr`

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_9

Various adjustments have been made to this model to improve its performance in terms of clustering and prediction. In our contribution, we propose to embed the co-clustering in the model.

Co-clustering is a simultaneous clustering of both dimensions of a data matrix that has proven to be more beneficial than traditional one-sided clustering, especially when dealing with sparse data. When dealing with high dimensional data sparse or not, co-clustering turns out to be more valuable than one-sided clustering [1, 13], even if one is interested in clustering along one dimension only. In [4] the authors proposed the SCOAL approach (Simultaneous Co-clustering and Learning model), leading to co-clustering and prediction for binary data; they generalized the model to continuous data. However, this model does not take into account the sparsity of data in the sense that it does not lead to homogeneous blocks. The obtained results in terms of *Mean Square Error* (MSE) are good, but in terms of co-clustering (homogeneity of co-clusters), no analysis has been presented. This model is also related to the soft PDLF (Predictive Discrete Latent Factor) model [2], where the value of response y_{ij}'s in each co-cluster is modeled as a sum $\beta^T x_{ij} + \delta_{k\ell}$ where β is a global regression model. In contrast, $\delta_{k\ell}$ is a co-cluster specific offset. More recently, in [17] the authors proposed an algorithm taking into account only row covariates information to realize co-clustering and regression simultaneously. To this end, the authors are based on the latent block models [8]. In our contribution, we propose to rely also on this model but considering row and column covariates.

The proposed Latent Block Regression Model (LBRM) is an extension of finite mixtures of regression models where the co-clustering is embedded. It allows us to deal with co-clustering and regression simultaneously while taking into account covariates. To estimate the parameters we rely on a *Variational* Expectation-Maximization algorithm [7] referred to as `VEM-LBRM`.

2 From Clusterwise Regression to Co-clusterwise Regression

2.1 Latent Block Model (LBM)

Given an $n \times d$ data matrix $\mathbf{X} = (x_{ij}, i \in I = \{1, \ldots, n\}; j \in J = \{1, \ldots, d\})$. It is assumed that there exists a partition on I and a partition on J. A partition of $I \times J$ into $g \times m$ blocks will be represented by a pair of partitions $(\mathbf{z}, \mathbf{w})$. The k-th row cluster corresponds to the set of rows i such that $z_{ik} = 1$ and $z_{ik'} = 0\ \forall k' \neq k$. Thereby, the partition represented by $\mathbf{z}$ can be also represented by a matrix of elements in $\{0, 1\}^g$ satisfying $\sum_{k=1}^{g} z_{ik} = 1$. Similarly, the ℓ-th column cluster corresponds to the set of columns j and the partition $\mathbf{w}$ can be represented by a matrix of elements in $\{0, 1\}^m$ satisfying $\sum_{\ell=1}^{m} w_{j\ell} = 1$.

Considering the Latent Block Model (LBM) [6], it is assumed that each element x_{ij} of the $k\ell$th block is generated according to a parameterized probability density function (pdf) $f(x_{ij}; \alpha_{k\ell})$. Furthermore, in the LBM the univariate random variables x_{ij} are assumed to be conditionally independent given $(\mathbf{z}, \mathbf{w})$. Thereby, the conditional pdf of $\mathbf{X}$ can be expressed as $P(z_{ik} = 1, w_{j\ell} = 1|\mathbf{X}) =$

$P(z_{ik} = 1|\mathbf{X})P(w_{j\ell} = 1|\mathbf{X})$. From this hypothesis, we then consider the latent block model where the two sets I and J are considered as random samples and the row, and column labels become latent variables. Therefore, the parameter of the latent block model is $\boldsymbol{\Theta} = (\boldsymbol{\pi}, \boldsymbol{\rho}, \boldsymbol{\alpha})$, with $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_g)$ and $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_m)$ where $(\pi_k = P(z_{ik} = 1), k = 1, \ldots, g)$, $(\rho_\ell = P(w_{j\ell} = 1), \ell = 1, \ldots, m)$ are the mixing proportions and $\boldsymbol{\alpha} = (\alpha_{k\ell}; k = 1, \ldots g, \ell = 1, \ldots, m)$ where $\alpha_{k\ell}$ is the parameter of the distribution of block $k\ell$. Considering that the complete data are the vector $(\mathbf{X}, \mathbf{z}, \mathbf{w})$, i.e, we assume that the latent variable $\mathbf{z}$ and $\mathbf{w}$ are known, the resulting complete data log-likelihood of the latent block model $L_C(\mathbf{X}, \mathbf{z}, \mathbf{w}, \boldsymbol{\Theta}) = \log f(\mathbf{X}, \mathbf{z}, \mathbf{w}; \boldsymbol{\Theta})$ can be written as follows

$$\sum_{k=1}^{g} z_k \log \pi_k + \sum_{\ell=1}^{m} w_\ell \log \rho_\ell + \sum_{i=1}^{n}\sum_{j=1}^{d}\sum_{k=1}^{g}\sum_{\ell=1}^{m} z_{ik} w_{j\ell} \log \boldsymbol{\phi}_{k\ell}(x_{ij}; \boldsymbol{\alpha}_{k\ell}).$$

where the π_k's and ρ_ℓ's denote the proportions of row and columns clusters respectively; see for instance [8]. Note that the complete-data log-likelihood breaks into three terms: the first one depends on proportions of row clusters, the second on proportions of column clusters and the third on the pdf of each block or co-cluster. The objective is then to maximize the function $L_C(\mathbf{z}, \mathbf{w}, \boldsymbol{\Theta})$.

2.2 Latent Block Regression Model (LBRM)

For co-clustering of continuous data, the Gaussian latent block model can be used. For instance, note that it is easy to show that the minimization of the well-known criterion of $||\mathbf{X} - \mathbf{z}\boldsymbol{\mu}\mathbf{w}^T||^2 = \sum_{k=1}^{g}\sum_{\ell=1}^{m}\sum_{i|z_{ik}=1}\sum_{j|w_{j\ell}=1}(x_{ij} - \mu_{k\ell})^2$ where $\mathbf{z} \in \{0, 1\}^{n\times g}$, $\mathbf{w} \in \{0, 1\}^{d\times m}$ and $\boldsymbol{\mu} \in \mathbb{R}^{g\times m}$ is associated to Latent block Gaussian model whith $\boldsymbol{\alpha}_{k\ell} = (\mu_{k\ell}, \sigma^2_{k\ell})$, the proportions of row clusters and column clusters are equal and in addition the variances of blocks are identical [9]. Note that 1) the characteristic of the latent block model is that the rows and the columns are treated symmetrically 2) the estimation of the parameters requires a variational approximation [7, 17]. In the sequel, we see how can we integrate a regression model. Hereafter, we propose a novel Latent Block Regression model for co-clustering and learning simultaneously. The model considers the response matrix $\mathbf{Y} = [y_{ij}] \in \mathbb{R}^{n\times d}$ and the covariate tensor $\mathcal{X} = [1, \mathbf{x}_{ij}] \in \mathbb{R}^{n\times d\times v}$ where n is the number of rows, d the number of columns, and v the number of covariates. Figure 1 presents data structure for the proposed model LBRM.

In the following we propose the integration of mixture of regression [5] per block in the Latent Block model (LBM) considering the distribution $\Phi(y_{ij}|\mathbf{x}_{ij}; \lambda_{k\ell})$.We assume in the following the normality of Φ,

$$\Phi(y_{ij}|\mathbf{x}_{ij}; \lambda_{k\ell}) = p(y_{i,j}|\mathbf{x}_{ij}, \boldsymbol{\beta}_{k\ell}, \sigma_{k\ell}) = (2\pi\sigma^2_{k\ell})^{-0.5} \exp\left\{-\frac{1}{2\sigma^2_{k\ell}}(y_{ij} - \boldsymbol{\beta}^\top_{k\ell}\mathbf{x}_{ij})^2\right\}$$

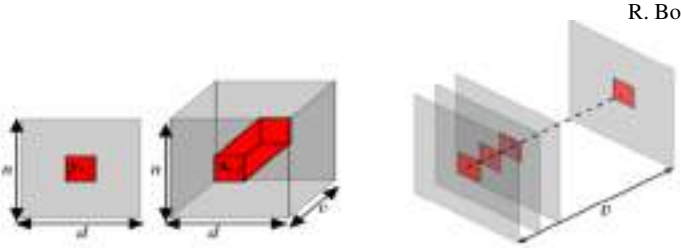


Fig. 1 Data representation for proposed model.

With the LBRM model, the parameter $\mathbf{\Omega}$ is composed of row and column proportions $\boldsymbol{\pi}$, $\boldsymbol{\rho}$ respectively, $\boldsymbol{\beta} = \{\boldsymbol{\beta}_{11}, \ldots, \beta_{gm}\}$ with $\boldsymbol{\beta}_{k\ell}^{\top} = (\beta_{k\ell}^{0}, \beta_{k\ell}^{1}, \ldots, \beta_{k\ell}^{v})$ where $\beta_{k\ell}^{0}$ represents the intercept of regression and $\boldsymbol{\sigma} = \{\sigma_{11}, \ldots, \sigma_{gm}\}$. The classification log-likelihood can be written:

$$\sum_{i,k} z_{ik} \log \pi_k + \sum_{j,\ell} w_{j\ell} \log \rho_\ell - \frac{1}{2} \sum_{k,\ell} z_{.k} w_{.\ell} \log(\sigma_{k\ell}^2) - \frac{1}{2\sigma_{k\ell}^2} \sum_{i,j,k,\ell} z_{ik} w_{j\ell} (y_{ij} - \boldsymbol{\beta}_{k\ell}^{\top} \mathbf{x}_{ij})^2$$

with $z_{.k} = \sum_i z_{ik}$ et $w_{.\ell} = \sum_j w_{j\ell}$.

3 Variational EM Algorithm

To estimate $\mathbf{\Omega}$, the EM algorithm [3] is a candidate for this task. It maximizes the log-likelihood $f(\mathcal{X}, \mathbf{\Omega})$ w.r. to $\mathbf{\Omega}$ iteratively by maximizing the conditional expectation of the complete data log-likelihood $L_C(\mathbf{z}, \mathbf{w}; \mathbf{\Omega})$ w.r. to $\mathbf{\Omega}$, given a previous current estimate $\mathbf{\Omega}^{(c)}$ and the observed data $\mathbf{x}$. Unfortunately, difficulties arise owing to the dependence structure among the variables x_{ij} of the model. To solve this problem an approximation using the [12] interpretation of the EM algorithm can be proposed; see, e.g., [7, 8]. Hence, the aim is to maximize the following lower bound of the log-likelihood criterion: $F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}; \mathbf{\Omega}) = L_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \mathbf{\Omega}) + H(\tilde{\mathbf{z}}) + H(\tilde{\mathbf{w}})$ where $H(\tilde{\mathbf{z}}) = -\sum_{i,k} \tilde{z}_{ik} \log \tilde{z}_{ik}$ with $\tilde{z}_{ik} = P(z_{ik} = 1|\mathcal{X})$, $H(\tilde{\mathbf{w}}) = -\sum_{j,\ell} \tilde{w}_{j\ell} \log \tilde{w}_{j\ell}$ with $\tilde{w}_{j\ell} = P(w_{j\ell} = 1|\mathcal{X})$, and $L_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}; \tilde{\mathbf{\Omega}})$ is the fuzzy complete data log-likelihood (up to a constant). $L_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}; \mathbf{\Omega})$ is given by

$$\begin{aligned} L_C(\mathbf{z}, \mathbf{w}, \mathbf{\Omega}) = & \sum_{i,k} \tilde{z}_{ik} \log \pi_k + \sum_{j,\ell} \tilde{w}_{j\ell} \log \rho_\ell - \frac{1}{2} \sum_{k,\ell} \tilde{z}_{.k} \tilde{w}_{.\ell} \log(\sigma_{k\ell}^2) \\ & - \frac{1}{2\sigma_{k\ell}^2} \sum_{i,j,k,\ell} \tilde{z}_{ik} \tilde{w}_{j\ell} (y_{ij} - \boldsymbol{\beta}_{k\ell}^{\top} \mathbf{x}_{ij})^2 \end{aligned}$$

The maximization of $F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \mathbf{\Omega})$ can be reached by realizing the three following optimization: update $\tilde{\mathbf{z}}$ by $\text{argmax}_{\tilde{\mathbf{z}}} F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \mathbf{\Omega})$, update $\tilde{\mathbf{w}}$ by $\text{argmax}_{\tilde{\mathbf{w}}} F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \mathbf{\Omega})$, and update $\mathbf{\Omega}$ by $\text{argmax}_{\mathbf{\Omega}} F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \mathbf{\Omega})$. In what follows, we detail the Expectation (E) and Maximization (M) step of the Variational EM algorithm for tensor data.

E-step. It consists in computing, for all i, k, j, ℓ the posterior probabilities $\tilde{z}_{ik}$ and $\tilde{w}_{j\ell}$ maximizing $F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \boldsymbol{\Omega})$ given the estimated parameters $\boldsymbol{\Omega}_{k\ell}$. It is easy to show that, the posterior probability $\tilde{z}_{ik}$ maximizing $F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \boldsymbol{\Omega})$ is given by: $\tilde{z}_{ik} \propto \pi_k \exp\left(\sum_{j,\ell} \tilde{w}_{j\ell} \log\left(p(y_{ij}|\mathbf{x}_{ij}, \boldsymbol{\beta}_{k\ell}, \sigma_{k\ell})\right)\right)$. In the same manner, the posterior probability $\tilde{w}_{j\ell}$ is given by: $\tilde{w}_{j\ell} \propto \rho_\ell \exp\left(\sum_{i,k} \tilde{z}_{ik} \log\left(p(y_{ij}|\mathbf{x}_{ij}, \boldsymbol{\beta}_{k\ell}, \sigma_{k\ell})\right)\right)$
M-step. Given the previously computed posterior probabilities $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{w}}$, the M-step consists in updating , $\forall k, \ell$, the parameters of the model π_k, ρ_ℓ, $\boldsymbol{\mu}_{k\ell}$ and $\boldsymbol{\lambda}_{k\ell}$ maximizing $F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \boldsymbol{\Omega})$. Using the computed quantities from step E, the maximization step (M-step) involves the following closed-form updates.

- Taking into account the constraints $\sum_k \pi_k = 1$ and $\sum_\ell \rho_\ell = 1$, it is easy to show that $\pi_k = \frac{\sum_i \tilde{z}_{ik}}{n} = \frac{\tilde{z}_{.k}}{n}$ and $\rho_\ell = \frac{\sum_j \tilde{w}_{j\ell}}{d} = \frac{\tilde{w}_{.\ell}}{d}$.
- The update of $\lambda_{k\ell}$ which is formed by $(\boldsymbol{\beta}_{k\ell}, \sigma_{k\ell})$ where can be given by simple derivates of $F_C(\tilde{z}, \tilde{w}, \boldsymbol{\Omega})$ with respect to $\boldsymbol{\beta}_{k\ell}$ and $\sigma_{k\ell}$ respectively. This leads to

$$\boldsymbol{\beta}_{k\ell} = \left(\sum_{i,j} \tilde{z}_{ik}\tilde{w}_{j\ell} y_{ij}\mathbf{x}_{ij}\right)\left(\sum_{i,j} \tilde{z}_{ik}\tilde{w}_{j\ell}\mathbf{x}_{ij}\mathbf{x}_{ij}^\top\right)^{-1}, \sigma^2_{k\ell} = \frac{\sum_{i,j} \tilde{z}_{ik}\tilde{w}_{j\ell}(y_{ij} - \boldsymbol{\beta}_{k\ell}^\top \mathbf{x}_{ij})^2}{\sum_{i,j} \tilde{z}_{ik}\tilde{w}_{j\ell}}.$$

The proposed algorithm for tensor data referred to as `VEM-LBRM` alternates the two previously described steps Expectation-Maximization. At the convergence, a hard co-clustering is deduced from the posterior probabilities.

4 Experimental Results

First, we evaluate the proposed `VEM-LBRM` on three synthetic datasets in terms of co-clustering and regression. We compare `VEM-LBRM` with some clustering and regression methods namely `Global model` which is a single multiple linear regression model performed on all observations, `K-means`, `Clusterwise`, `Co-clustering` and `SCOAL`. We retain two widely used measures to assess the quality of clustering, namely the Normalized Mutual Information (NMI) [16] and the Adjusted Rand Index (ARI) [15]. Intuitively, NMI quantifies how much the estimated clustering is informative about the true clustering. The ARI metric is related to the clustering accuracy and measures the degree of agreement between an estimated clustering and a reference clustering. Both NMI and ARI are equal to 1 if the resulting clustering is identical to the true one. On the other hand, we use RMSE (Root MSE) and MAE (Mean Absolute Error) metrics to evaluate the precision of prediction while RMSE is a loss function which is suitable for Gaussian noises when MAE uses the absolute value which is less sensitive to extreme values.

We generated tensor data $\mathbf{X}$ with size $200 \times 200 \times 2$ according to Gaussian model per block. In the simulation study, we considered three scenarios by varying the regression parameters — the examples have blocks with different regression collinearity and different co-clusters structure complexity. The parameters for each example are reported in Tables 1. In Figures 2 and 3 are depicted the true regression planes and the true simulated response matrix $\mathbf{Y}$.

Table 1 Parameters generation for examples.

Dataset	Example 1		Example 2		Example 3	
	$\boldsymbol{\pi} = [0.35, 0.35, 0.3]$, $\boldsymbol{\rho} = [0.55, 0.45]$					
σ	$\sigma = 5$		$\sigma = 7$		$\sigma = 7$	
$\boldsymbol{\Sigma}$	$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$		$\boldsymbol{\Sigma} = \begin{bmatrix} 2 & 0.3 \\ 0.3 & 2 \end{bmatrix}$		$\boldsymbol{\Sigma} = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}$	
Co-clusters	$\boldsymbol{\beta}_{k\ell}$	$\boldsymbol{\mu}_{k\ell}$	$\boldsymbol{\beta}_{k\ell}$	$\boldsymbol{\mu}_{k\ell}$	$\boldsymbol{\beta}_{k\ell}$	$\boldsymbol{\mu}_{k\ell}$
Cluster (1,1)	[1, -10, 1]	[5,20]	[1, -10, 1]	[5,20]	[1, -10, 1]	[5,20]
Cluster (1,2)	[10, 4, 13]	[5,10]	[1, -10, 1]	[5,10]	[1, -10, 1]	[5,10]
Cluster (2,1)	[3, 20, -2]	[10,20]	[1, -10, 1]	[10,20]	[1, -10, 1]	[5,30]
Cluster (2,2)	[-5, -2, -6]	[10,10]	[7, 5, -10]	[10,10]	[7, 5, -10]	[20,10]
Cluster (3,1)	[-10, 20, 10]	[20,20]	[7, 5, -10]	[20,20]	[7, 5, -10]	[20,20]
Cluster (3,2)	[7, 5, -10]	[20,10]	[7, 5, -10]	[20,10]	[7, 5, -10]	[20,30]

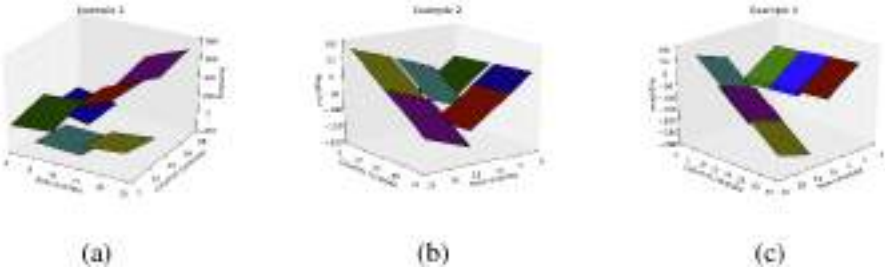


Fig. 2 Synthetic data: True regression plans according to the chosen parameters.

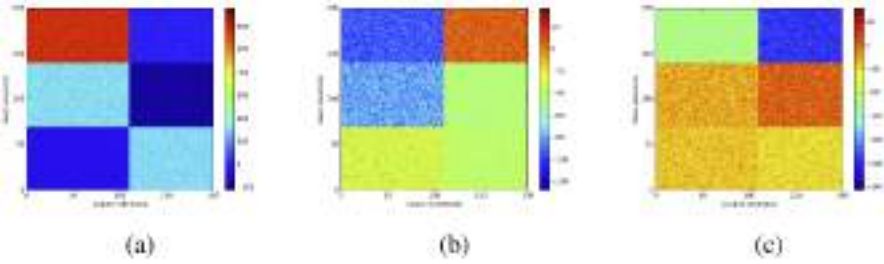


Fig. 3 Synthetic data: True co-clustering according to the chosen parameters.

In our illustrations, we consider co-clustering and regression challenges. All metrics concerning rows and columns are computed by averaging on ten random training, and testing data split using an 80% vs. 20% of training and validation data. Thereby, we compare `VEM-LBRM` with `Global model` (which is a multiple linear regression), `K-means`, `Clusterwise` by reshaping the tensor to matrix with size $N \times v$ where $N = n \times d$. On the other hand, the `VEM` algorithm for co-clustering is applied on response matrix $\mathbf{Y}$. Furthermore, for clustering algorithms, the RMSE, MAE, and R-squared are computed by applying linear regression on each obtained co-cluster. In Table, 2 are reported the performances for all algorithms. The missing values represent measures that cannot be computed by the corresponding models. From these comparisons, we observe that whether the block structure is easy to identify or not, the ability of `VEM-LBRM` to outperform other algorithms.

To go further, note that in [11], the authors reformulated the clusterwise and introduced the linear cluster-weighted model (CWM) in a statistical setting and showed that it is a general and flexible family of mixture models. They included in

Table 2 (co)-clustering and prediction: mean and sd in parentheses.

Examples	Algorithms	Regression					Clustering			
		RMSE		MAE		Rsquare	ARI		NMI	
		Training	Test	Training	Test	Avg.	Row	Col	Row	Col
Example1	Global model	164.38	164.05	145.29	145.05	0.46	-	-	-	-
		(0.03)	(0.49)	(0.08)	(0.71)	(0.0)	-	-	-	-
	K-means	49.62	49.51	34.86	34.91	0.8	0.61	-	0.49	-
		(60.2)	(67.48)	(33.56)	(35.79)	(0.02)	0.02	-	0.03	-
	Clusterwise	154.57	154.47	127.77	127.93	0.52	0.07	-	0.01	-
	($g = 3$)	(0.01)	(0.36)	(0.03)	(0.45)	(0.0)	0.0	-	0.0	-
	Co-clustering	10.86	10.83	7.29	7.29	0.88	0.84	1.0	0.71	1.0
	($g = 3$)	(14.76)	(14.36)	(4.67)	(4.59)	(0.0)	0.01	0.0	0.04	0.0
	SCOAL	14.99	14.92	10.45	10.41	0.99	0.91	1.0	0.84	1.0
	($g = 3, m = 2$)	(207.56)	(208.91)	(89.48)	(90.55)	(0.0)	0.01	0.0	0.04	0.0
	VEM-LBRM	**7.1**	**7.06**	**5.29**	**5.26**	**0.99**	**0.95**	**1.0**	**0.92**	**1.0**
	($g = 3, m = 2$)	**(17.71)**	**(16.86)**	**(6.8)**	**(6.32)**	**(0.0)**	**(0.01)**	**(0.0)**	**(0.03)**	**(0.0)**
Example2	Global model	29.15	29.21	24.64	24.68	0.34	-	-	-	-
		(0.04)	(0.15)	(0.04)	(0.12)	(0.0)	-	-	-	-
	K-means	10.43	10.49	7.73	7.77	0.71	0.56	-	0.45	-
		(0.25)	(0.24)	(0.17)	(0.16)	(0.01)	0.0	-	0.0	-
	Clusterwise	18.54	18.62	11.33	11.38	0.73	0.15	-	0.16	-
	($g = 3$)	(0.09)	(0.27)	(0.06)	(0.14)	(0.0)	0.0	-	0.0	-
	Co-clustering	7.5	7.49	5.89	5.9	0.8	0.95	1.0	0.94	1.0
	($g = 3$)	(1.35)	(1.38)	(0.82)	(0.86)	(0.07)	0.14	0.0	0.17	0.0
	SCOAL	12.63	12.69	8.75	8.81	0.81	0.97	1.0	0.94	1.0
	($g = 3, m = 2$)	(12.57)	(12.81)	(7.38)	(7.58)	(0.35)	0.1	0.0	0.17	0.0
	VEM-LBRM	**6.99**	**6.99**	**5.57**	**5.57**	**0.96**	**1.0**	**1.0**	**1.0**	**1.0**
	($g = 3, m = 2$)	**(0.01)**	**(0.04)**	**(0.01)**	**(0.02)**	**(0.0)**	**(0.0)**	**(0.0)**	**(0.0)**	**(0.0)**
Example3	Global model	45.38	45.24	38.33	38.21	0.49	-	-	-	-
		(0.06)	(0.24)	(0.07)	(0.26)	(0.0)	-	-	-	-
	K-means	10.47	10.41	7.44	7.42	0.83	0.54	-	0.45	-
		(1.73)	(1.74)	(1.08)	(1.08)	(0.08)	0.01	-	0.01	-
	Clusterwise	23.09	23.18	12.09	12.15	0.87	0.09	-	0.09	-
	($g = 3$)	(1.84)	(2.02)	(1.23)	(1.29)	(0.02)	0.0	-	0.0	-
	Co-clustering	9.48	9.39	6.98	6.93	0.73	0.74	1.0	0.7	1.0
	($g = 3$)	(0.16)	(0.22)	(0.01)	(0.02)	(0.02)	0.04	0.0	0.08	0.0
	SCOAL	27.32	27.14	16.82	16.73	0.57	0.98	1.0	0.96	1.0
	($g = 3, m = 2$)	(41.97)	(41.83)	(24.13)	(24.16)	(0.93)	0.07	0.0	0.12	0.0
	VEM-LBRM	**7.21**	**7.21**	**5.71**	**5.71**	**0.99**	**0.98**	**1.0**	**0.96**	**1.0**
	($g = 3, m = 2$)	**(0.68)**	**(0.7)**	**(0.42)**	**(0.42)**	**(0.0)**	**(0.07)**	**(0.0)**	**(0.12)**	**(0.0)**

the classical model of clusterwise the probability $\Phi'(\mathbf{x}_i|\mathbf{\Omega}_k)$ to model the covariates, whereas the classical cluster-wise model the output only using $\Phi(y_i|\mathbf{x}_i;\lambda_k)$. They prove that sufficient conditions for model identifiability are provided under a suitable assumption of Gaussian covariates [10]. We can include in LBRM a joint probability $\Phi'(\mathbf{x}_{ij}|\mathbf{\Omega}_{k\ell})$ where $\mathbf{\Omega}_{k\ell} = [\boldsymbol{\mu}_{k\ell}, \mathbf{\Sigma}_{k\ell}]$ to evaluate its impact in terms of clustering and regression. Figure 4 presents the graphical model of LBRM and its extension. The first experiments on real datasets give encouraging results.

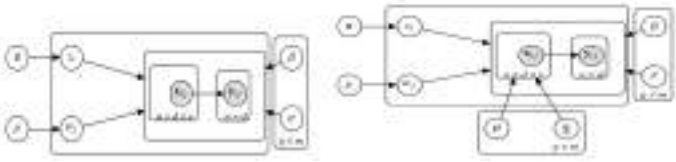


Fig. 4 Graphical model of LBRM (left) and its extension (right).

5 Conclusion

Inspired by the flexibility of the latent block model (LBM), we proposed extending it to tensor data aiming at both tasks: co-clustering and prediction. This model (LBRM) gives rise to a variational EM algorithm for co-clustering and prediction referred to as `VEM-LBRM`. This algorithm which can be viewed as the co-clusterwise algorithm can easily deal with sparse data. Empirical results on synthetic data showed that `VEM-LBRM` does give more encouraging results for clustering and regression than some algorithms that are devoted to one or both tasks simultaneously. For future work, we plan to develop the extension of LBRM and apply the proposed models for the recommender system task.

Acknowledgements Our work is funded by the German Federal Ministry of Education and Research under Grant Agreement Number 01IS19084F (XAPS).

References

1. Affeldt, S., Labiod, L., Nadif, M.: Regularized bi-directional co-clustering. Statistics and Computing, **31**(3), 1-17 (2021)
2. Agarwal, D., and Merugu, S.: Predictive discrete latent factor models for large scale dyadic data. In: SIGKDD, pp. 26–35 (2007)
3. Dempster, A. P., Laird, N. M., Rubin, D. B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society Series B (Methodological), **39**(1), 1–22 (1977)
4. Deodhar, M., Ghosh, J.: A framework for simultaneous co-clustering and learning from complex data. In: SIGKDD, pp. 250–259 (2007)
5. DeSarbo, W. S., and Cron, W. L.: A maximum likelihood methodology for clusterwise linear regression. Journal of Classification, **5**(2), 249–282 (1988)
6. Govaert, G., Nadif, M.: Clustering with block mixture models. Pattern Recognition, **36**, 463-473, (2003)
7. Govaert, G., Nadif, M.: An EM algorithm for the block mixture model. IEEE Transactions on Pattern Analysis and Machine Intelligence, **27**(4), 643–647 (2005)
8. Govaert, G., Nadif, M.: Block clustering with Bernoulli mixture models: Comparison of different approaches. Computational Statistics & Data Analysis, 3233–3245 (2008)
9. Govaert, G., Nadif, M.: Co-clustering: Models, Algorithms and Applications. John Wiley & Sons (2013)
10. Ingrassia, S., Minotti, S. C., Punzo, A.: Model-based clustering via linear cluster-weighted models. Computational Statistics & Data Analysis, **71**, 159–182 (2014)
11. Ingrassia, S., Minotti, S. C., Vittadini, G.: Local statistical modeling via a cluster-weighted approach with elliptical distributions. In: Journal of Classification, **29**(3), 363–401 (2012)
12. Neal, R. M., Hinton, G. E.: A view of the EM algorithm that justifies incremental, sparse, and other variants. In Learning in Graphical Models, pp. 355–368. Springer (1998)
13. Salah, A., Nadif, M.: Directional co-clustering. Advances in Data Analysis and Classification, **13**(3), 591-620 (2019)
14. Späth, H.: Algorithm 39 clusterwise linear regression. Computing, **22**(4), 367–373 (1979)
15. Steinley, D.: Properties of the Hubert–Arable Adjusted Rand Index. Psychological Methods, **9**(3), 386 (2004)
16. Strehl, A., Ghosh, J.: Cluster ensembles—a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research, **3**, 583–617 (2002)
17. Vu, D., Aitkin, M.: Variational algorithms for biclustering models. Computational Statistics & Data Analysis, **89**, 12–24 (2015)

Using Clustering and Machine Learning Methods to Provide Intelligent Grocery Shopping Recommendations

Nail Chabane, Mohamed Achraf Bouaoune, Reda Amir Sofiane Tighilt, Bogdan Mazoure, Nadia Tahiri, and Vladimir Makarenkov

Abstract Nowadays, grocery lists make part of shopping habits of many customers. With the popularity of e-commerce and plethora of products and promotions available on online stores, it can become increasingly difficult for customers to identify products that both satisfy their needs and represent the best deals overall. In this paper, we present a grocery recommender system based on the use of traditional machine learning methods aiming at assisting customers with creation of their grocery lists on the MyGroceryTour platform which displays weekly grocery deals in Canada. Our recommender system relies on the individual user purchase histories, as well as the available products' and stores' features, to constitute intelligent weekly grocery lists. The use of clustering prior to supervised machine learning methods allowed us to identify customers profiles and reduce the choice of potential products of interest for each customer, thus improving the prediction results. The highest average F-score of 0.499 for the considered dataset of 826 Canadian customers was obtained using the Random Forest prediction model which was compared to the Decision Tree, Gradient Boosting Tree, XGBoost, Logistic Regression, Catboost, Support Vector Machine and Naive Bayes models in our study.

Keywords: clustering, dimensionality reduction, grocery shopping recommendation, intelligent shopping list, machine learning, recommender systems

Nail Chabane · Mohamed Achraf Bouaoune · Reda Amir Sofiane Tighilt · Vladimir Makarenkov (✉)
Université du Québec à Montreal, 405 Rue Sainte-Catherine Est, Montreal, Canada,
e-mail: chabane.nail_amine@courrier.uqam.ca
e-mail: bouaoune.mohamed_achraf@courrier.uqam.ca
e-mail: tighilt.reda@courrier.uqam.ca;makarenkov.vladimir@uqam.ca

Bogdan Mazoure
McGill University and MILA - Quebec AI Institute, 845 Rue Sherbrooke O, Montreal, Canada,
e-mail: bogdan.mazoure@mail.mcgill.ca

Nadia Tahiri
University of Sherbrooke, 2500 Bd de l'Université, Sherbrooke, Canada,
e-mail: Nadia.Tahiri@USherbrooke.ca

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_10

1 Introduction

Grocery shopping is a common activity that involves different factors such as budget and impulse purchasing pressure [1]. Customers typically rely on a mental or digital list to facilitate their grocery trips. Many of them show a favorable interest towards tools and applications that help them manage their grocery lists, while keeping them updated with special offers, coupons and promotions [2, 3]. Major retailers throughout the world typically offer discounts on different products every week in order to improve sales and attract new customers. This very common practice leads to the fact that thousands of items go on special simultaneously across different retailers at a given week. The resulting information overload often makes it difficult for customers to quickly identify the deals that best suit their needs, which can become a source of frustration [4]. To address this problem, many grocery stores have taken advantage of the popularity of e-commerce to set up their own websites featuring various functionalities, including Recommender Systems, to assist customers during the shopping process.

Recommender Systems (RSs) [5] are tools and techniques that offer personalized suggestions to users based on several parameters (e.g. their past behavior). RSs have recently become a field of interest for researchers and retailers as many e-commerces, online book stores and streaming platforms have started to offer this service on their websites (e.g. Amazon, Netflix and Spotify). Here, we recall some recent works in this field. Faggioli et al. [6] used the popular Collaborative Filtering (CF) approach to predict the customer's next basket in a context of grocery shopping, taking into account the recency parameter. When comparing their model with the CF baseline models, Faggioli et al. observed a consistent improvement of their prediction results. Che et al. [7] used attention-based recurrent neural networks to capture both inter- and intra-basket relationships, thus modelling users' long-term preferences dynamic short-term decisions.

Content-based recommendation has also proven efficient in the literature, as demonstrated by Xia et al. [8] who proposed a tree-based model for coupons recommendation. By processing their data with undersampling methods, the authors were able to increase the estimated click rate from 1.20% to 7.80% as well as to improve significantly the F-score results using Random Forest Classifier and the recall results using XGBoost. Dou [9] presented a statistical model to predict whether a user will buy or not buy an item using Yandex's CatBoost method [10]. Dou relied on contextual and temporal features as well as on some session features, such as the time of visit of specific web pages, to demonstrate the efficiency of CatBoost in this context. Finally, Tahiri et al. [11] used recurrent and feedforward neural networks (RNNs and FFNs) in combination with non-negative matrix factorization and gradient boosting trees to create intelligent weekly grocery baskets to be recommended to the users of MyGroceryTour. Tahiri et al. considered different (from our study) features characterizing the users of MyGroceryTour to provide their predictions, with the best F-score results of 0.37 obtained from the augmented dataset.

2 Materials and Methods

2.1 Data Considered

In this section we describe the dataset obtained from MyGroceryTour website used in our research. MyGroceryTour [11] is a Canadian grocery shopping website and database available in both English and French languages. The main purpose of the website is to present weekly specials offered by the major grocery retailers in Canada. It allows users to display grocery products available in their living area, compare their products over different stores as well as to build their grocery shopping baskets based on the provided insights. MyGroceryTour users can easily archive and manage their grocery lists and access them at any given time.

In this study, we considered 826 MyGroceryTour users with varying numbers of grocery baskets (between 3 and 100 baskets were available per user). The grocery baskets contained different products added by users when they were creating their weekly shopping lists. In our recommender system (i.e current basket prediction experiment), we have considered the following features:

- *user_id* : unique user identifier (numerical)
- *basket_id* : unique basket identifier (numerical)
- *product_id* : unique product identifier (numerical)
- *category* : category of the product (categorical)
- *price* : price of the product (numerical)
- *special* : discount on the product (in %) compared to regular price (numerical)
- *distance_min* : minimal distance between user's home and the closest store where the product was available (numerical)
- *distance_mean* : mean distance between user's home and all stores where the product was available (numerical)
- *availability* : availability of the product at different stores (binary)

In addition, we engineered the *total_bought* feature which represents, for each product, the total number of times it has been bought over all users.

2.2 Data Normalization

Data normalization is an important data preprocessing step in both unsupervised and supervised machine learning [12] as well as in data mining [13]. Prior to feeding the data to our models we rescaled the available features using z-score standardization. Thus, all rescaled features had the mean of 0 and the standard deviation of 1:

$$z(x_f) = \frac{x_f - \mu_f}{\sigma_f}, \tag{1}$$

where x_f is the original value of the observation at feature f, μ_f is the mean and σ_f is the standard deviation of f.

2.3 Further Data Preprocessing Steps

In order to determine which weekly products could be recommended to a given user we propose to classify them using both clustering (unsupervised learning) and traditional supervised machine learning methods. The final recommendation is obtained based on the availability of the products, the data on the products' regular prices and available discounts, as well as on the user's shopping history. In our context, the baskets contain only the products bought by the users. The information about the other available products (not selected by the user at the moment he/she organized his/her shopping basket) is also available on MyGroceryTour. It has been used to create a large class of available items that were not bought by the user.

While we considered the items bought by a given user as positive feedback, we regarded the items that were available to this user at the time of the order, but not acquired by him/her, as a negative feedback. For an order of size P, if T is the total amount of items available to the user at the time of the order, the negative feedback N for that order is $N = T - P$. In this context, N usually represents thousands of products, while P is typically inferior to 50. This difference in size between positive and negative feedback can lead to a situation of imbalanced training data and could result in an important loss in performance. Similarly to Xia et al. [8], we applied an undersampling method to balance our data instead of considering all of the available disregarded items as the negative feedback.

To identify customer profiles and perform a preselection of products that are susceptible to be of interest to a given user, we first carried out the clustering of the normalized original dataset (the K-means [14] and DBSCAN [15] data partitioning algorithms were used). Then, we limited the choice of the items offered to a given user to the products purchased by the members of his/her cluster. By doing so, we managed to reduce the amount of products which could be recommended to the user and thus minimize eventual classifications mistakes. The clustering phase is detailed in the Subsection 2.4. Then traditional machine learning methods were used to provide the final weekly recommendation. The size S of the weekly basket recommended to a given user was equal to the mean size of his/her previous shopping baskets. As the number of items to be recommended by the machine learning methods was often greater than S, we retained as final recommendation the top S items, ranked according to the confidence score (i.e. the probability estimate for a given observation, computed using the *predict_proba* function from the *scikit-learn* [16] library).

2.4 Data Clustering

In this section, we present the steps we carried out to obtain the clusters of users. As explanatory features used to generate clusters, we considered the mean prices and mean specials of the products purchased by the user as well as a new feature, called here the fidelity ratio FR_u, which is meant to give insight on whether a given user u has a favorite store where he/she makes most of his/her grocery purchases. FR_u is defined as follows:

$$FR_u = \frac{X_{max,u} - \frac{1}{(n-1)} \sum_{i=2}^{n} X_{i,u}}{X_{total,u}}, \tag{2}$$

where $X_{max,u}$ is the total number of products bought by user u at the store where he/she made most of his/her purchases, n (n>1) is the total number of stores visited by user u, and $X_{total,u}$ ($X_{total,u} = X_{max,u} + \sum_{i=2}^{n} X_{i,u}$) is the total number of products purchased by user u over all stores he/she visited. A high fidelity ratio means that user u buys most of his/her products at the same store, whereas a low fidelity ratio indicates that user u buys his/her products at different stores. When user u purchases all of his/her products at the same store ($X_{max,u} = X_{total,u}$ and $n = 1$), the fidelity ratio equals 1. It equals 0 when he/she purchases the same number of products at different stores.

The K-means [14] and DBSCAN [15] algorithms were used to perform clustering. Here we present the results of DBSCAN, as the clusters provided by DBSACAN had less entity overlap than those provided by K-means. The main advantage of DBSCAN is that this density-based algorithm is able to capture clusters of any shape.

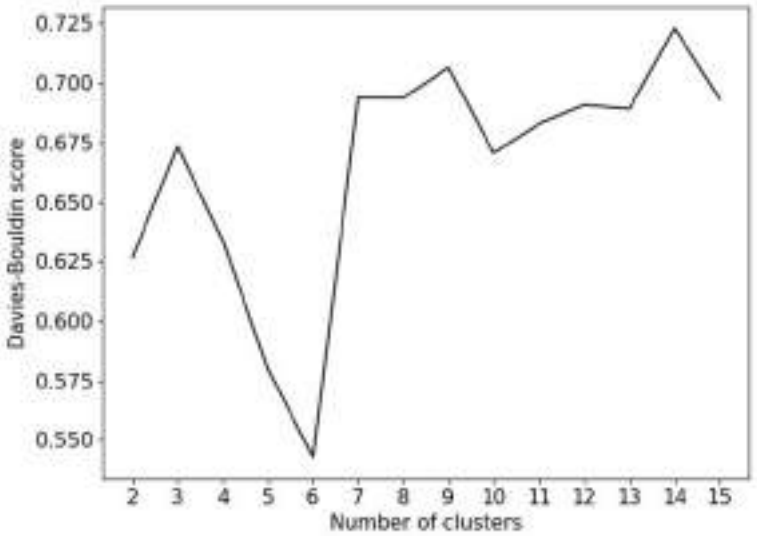


Fig. 1 Davies-Bouldin cluster validity index variation with respect to the number of clusters.

We used the Davies-Bouldin (DB) [17] cluster validity index to determine the number of clusters in our dataset. The Davies-Bouldin index is the average similarity between each cluster C_i for $i = 1, ..., k$ and its most similar counterpart C_j. It is calculated as follows:

$$DB = \frac{1}{k} \sum_{i=1}^{k} \underset{i \neq j}{max} R_{ij}, \tag{3}$$

where R_{ij} is the similarity measure between clusters calculated as $(d_i + d_j)/\delta_{ij}$, where d_i (d_j) is the the mean distance between objects of cluster C_i (C_j) and the cluster centroid and δ_{ij} is the distance between the centroids of clusters C_i and C_j.

Figure 1 illustrates the variation of the Davies-Bouldin cluster validity index whose lowest (i.e. best) value was reached for our dataset with 6 clusters. The resulting clusters are represented in Figure 2. After performing the data clustering, we applied the t-SNE [18] dimensionality reduction method for data visualisation purposes. Since t-SNE is known to preserve the local structure of the data but not the global one, we used the PCA initialization parameter to mitigate this issue.

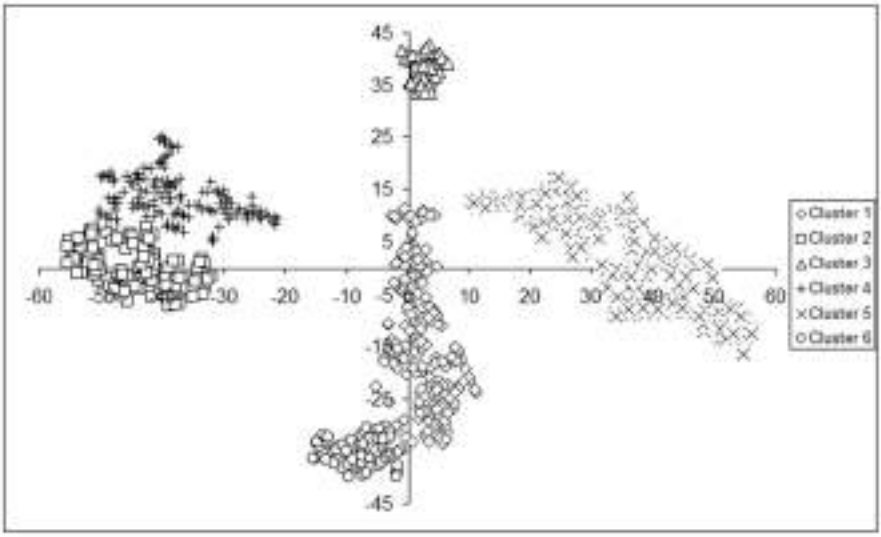


Fig. 2 Clustering results : Clustering obtained with DBSCAN with the best number of clusters according to the Davies-Bouldin index. Data reduction was performed using t-SNE. The 6 clusters of customers found by DBSCAN are represented by different symbols.

We have noticed that the users in Cluster 1 (see Fig. 2) are fairly sensitive to specials and have a high fidelity score, the users in Cluster 2 mostly purchase products on special in different stores, the users in Cluster 3 seem to be sensitive to the total price of their shopping baskets, Cluster 4 includes the users who are sensitive to specials but have a low fidelity score, Cluster 5 includes the users who are not very attracted by specials but are rather loyal to their favorite store(s), and the users in Cluster 6 tend to buy products on special and have high fidelity scores.

3 Application of Supervised Machine Learning Methods

To predict the products to be recommended for the current weekly basket, we used the following supervised machine learning methods: Decision Tree, Random Forest,

Gradient Boosting Tree, XGBoost, Logistic Regression, Catboost, Support Vector Machine and Naive Bayes. These methods were used through their *scikit-learn* implementations [16]. Due to the lack of large datasets we did not use deep learning models in our study. We decided to use these classical machine learning methods because they are usually recommended to work with smaller datasets contrary to their deep leaning counterparts. Also, deep leaning algorithms usually don't handle properly mixed types of features present in our data. Most of the methods we used are the ensemble methods, i.e. they use multiple replicates to reduce the variance. The F-score results provided by each method without (using all products available) and with clustering (using only the products purchased by the cluster members) are presented in Table 1.

As shown in Table 1, Random Forest outperformed the other competing methods without and with data clustering, providing the average F-scores of 0.494 and 0.499 (obtained over all users), respectively. Tree-based models relying on gradient boosting performed relatively well and could possibly give better results with a different data processing. We can also notice that all the methods, except CatBoost, benefited from the data clustering process.

Table 1 F-scores provided by ML methods without and with clustering of MyGroceryTour users.

Machine learning methods	Results without clustering	Results with clustering
CatBoost	0.438	0.438
Decision Tree	0.463	0.468
Gradient Boosting Tree	0.488	0.495
Logistic Regression	0.474	0.478
Naive Bayes	0.433	0. 436
Random Forest	**0.494**	**0.499**
SVM-RBF	0.392	0.397
XGBoost	0.476	0.481

4 Conclusion

In this paper, we presented a novel recommender system that is intended to predict the content of the customer's weekly basket depending on his/her purchase history. Our system is also able to predict the store(s) where the purchase(s) will take place. The clustering step allowed us to identify customer profiles and to improve the F-score result for every tested machine learning model, except CatBoost. Using our methodology and the new data available on MyGroceryTour, we were able to improve the F-score performance by the margin of 0.129, compared to the results obtained by Tahiri et al. [11]. Our model is able to predict products that will be purchased again or acquired for the first time by a given user, but it is not yet able to predict the optimal quantity for each product to be bought. Another important issue is how to provide plausible recommendations for customers without shopping history (i.e. the cold start problem). We will tackle these important issues in our future work.

References

1. Vincent-Wayne, M., Aylott, R.: An exploratory study of grocery shopping stressors. Int. J. Retail. Distrib. Manag. **26**, 362–373 (1998)
2. Newcomb, E., Pashley, T., Stasko, J.: Mobile computing in the retail arena. In: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 337-344. Association for Computing Machinery, New York (2003)
3. Sourav, B., Floréen, P., Forsblom, A., Hemminki, S., Myllymäki, P., Nurmi, P., Pulkkinen, T., Salovaara, A.: An Intelligent Mobile Grocery Assistant. In: 2012 Eighth International Conference on Intelligent Environments, pp. 165-172. IEEE, Guanajuato (2012)
4. Park, Y., Chang, K.: Individual and group behavior-based customer profile model for personalized product recommendation. Expert Systems with Applications **36**(2), 1932-1939 (2009)
5. Ricci, F., Rokach, L., Shapira, B.: Recommender Systems: Introduction and Challenges. In: Recommender Systems Handbook, pp. 1-34. Springer, Boston (2015)
6. Faggioli, G., Mirko P., Fabio A.: Recency aware collaborative filtering for next basket recommendation. In: Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization, pp. 80-87. Association for Computing Machinery, New York (2020)
7. Che et al.: Inter-basket and intra-basket adaptive attention network for next basket recommendation. IEEE Access **7**, 80644-80650 (2019)
8. Xia, Y. Giuseppe, D. F., Shikhar, V., Ankur, D.: A content-based recommender system for e-commerce offers and coupons. In: Proceedings of the SIGIR 2017 eCom workshop. eCOM@ SIGIR, Tokyo (2017)
9. Dou, X.: Online purchase behavior prediction and analysis using ensemble learning. In : 2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), pp. 532-536. IEEE (2020)
10. Prokhorenkova, L., Gleb G., Aleksandr V., Anna V. D., Andrey G.: CatBoost: unbiased boosting with categorical features (2017) Available via arXiv.
11. Tahiri, N., Mazoure, B. and Makarenkov, V.: An intelligent shopping list based on the application of partitioning and machine learning algorithms. In: Proceedings of the 18th Python in Science Conference (SCIPY 2019), pp. 85-92. Austin, Texas (2019)
12. Kotsiantis, S. B., Kanellopoulos, D., Pintelas, P. E.: Data preprocessing for supervised leaning. Int. J. Comput. Sci. **2**, 111-117 (2006)
13. García, S., Luengo, J., Herrera, F.: Data Preprocessing in Data Mining. Springer, Cham, Switzerland (2015)
14. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, vol. 1, no. 14, pp. 281-297 (1967)
15. Ester, M., Kriegel, H. P., Sander, J., Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining. AAAI Press, pp. 226–231. AAAI Press, Portland, Oregon (1996)
16. Pedregosa et al.: Scikit-learn: Machine Learning in Python. JMLR **12**, 2825-2830 (2011)
17. Davies, D. L., Bouldin, D. W.: A Cluster Separation Measure. In: IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-1, no. 2, pp. 224-227 (1979)
18. Van der Maaten, L. J. P., Hinton, G.: Visualizing High-Dimensional Data Using t-SNE. J. Mach. Learn. Res. **9**, 2579-2605 (2008)

COVID-19 Pandemic: a Methodological Model for the Analysis of Government's Preventing Measures and Health Data Records

Theodore Chadjipadelis and Sofia Magopoulou

Abstract The study aims to investigate the associations between the government's response measures during the COVID-19 pandemic and weekly incidence data (positivity rate, mortality rate and testing rate) in Greece. The study focuses on the period from the detection of the first case in the country (26th February 2020) to the first week of 2022 (08th January 2022). Data analysis was based on Correspondence Analysis on a fuzzy-coded contingency table, followed by Hierarchical Cluster Analysis (HCA) on the factor scores. Results revealed distinct time periods during which interesting interactions took place between control measures and incidence data.

Keywords: hierarchical cluster analysis, correspondence analysis, COVID-19, evidence-based policy making

1 Introduction

The present study focuses on the period of the COVID-19 pandemic in Greece, from the detection of the first case of COVID-19 to the first week of 2022. This period can be divided into five distinct phases. The first phase extends from the beginning of 2020 until the first lockdown, i.e., from the first case reported in Greece until the end of the first quarantine period in May 2020. The second phase concerns the interim period from June to October 2020, when the pandemic indices improved, and policies were loosened for the opening of tourism. The third phase concerns the second lockdown and the evolution of the pandemic in the country from November 2020 to April 2021, when the first vaccination period of the adult population took place. The fourth phase includes the interim period from May 2021 to October

Theodore Chadjipadelis (✉)
Aristotle University of Thessaloniki, Greece, e-mail: chadji@polsci.auth.gr

Sofia Magopoulou
Aristotle University of Thessaloniki, Greece, e-mail: sofimago@polsci.auth.gr

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_11

2021, where a general stabilization of the number of cases occurred, while the last period refers to a significant increase in the number of cases from November 2021 to January 2022.

Overall, from March 2020 to January 2022, a total of 1,79 million cases of COVID-19 were recorded in Greece (Figure 1) and a total of 22,635 deaths. Vaccination coverage is as of January 2022 over 65% of country's population, i.e., 7,241,468 fully vaccinated citizens.

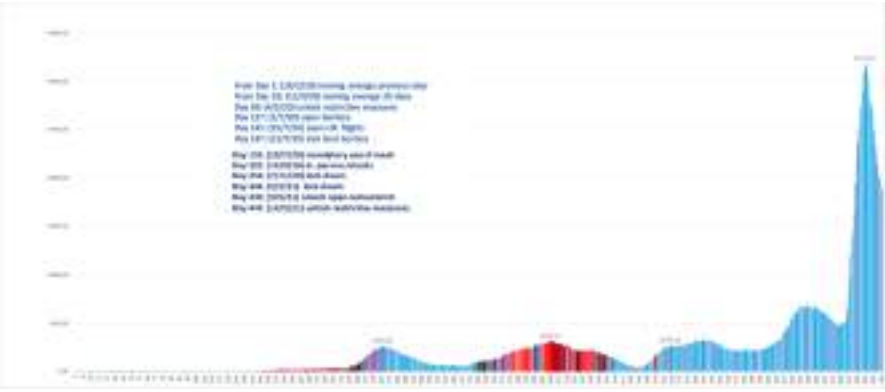


Fig. 1 Record of cases of COVID-19 in Greece (March 2020-January 2022).

In this study, a combination of multivariate data analysis methods was employed to analyze COVID-19-related data so as to assess the quality of decision-making outputs during the crisis and improve evidence-based decision-making processes. Section 2 presents the methodology and describes the data sources and the data analysis workflow. Section 3 presents the study results and Section 4 discusses the results and proposes methodological tools and presents the paper conclusions.

2 Methodology

2.1 Data

For the study purposes, data were obtained from the Oxford Covid-19 Government Response Tracker (OxCGRT) and were combined with self-collected Covid-19 data for Greece [3] daily updated in Greek. The Oxford Covid-19 Government Response Tracker (OxCGRT) collects publicly available information reflecting government response from 180 countries since 1 January 2020 [4]. The tracker is based on data for 23 indicators. In this study, two groups of indicators were considered: Containment & Closure and Health Systems in the case of Greece. The first group of indicators refers to "collective" level policies and measures, such as school closures and restriction in

mobility, while the second refers to "individual" level policies and measures, such as testing and vaccination. Specifically, the collective level indicators refer to policies taken by the governments' and reflect on a collective level on the society: school closing, workplace closing, cancelation of public events, restrictions on gathering, closure of public transport, stay at home requirements, internal movement restrictions and international travel controls. The health system policies primarily touch upon the individual level and specifically refer to: public information campaigns, testing, contact tracing, healthcare facilities, vaccines' investments, facial coverings, vaccination and protection of the elderly people. All collective-level indicators (C1 to C8) were summed to yield a total score (ranging from 0 to 16). Similarly, individual-level indicators (H1 to H3 and H6 to H8) were summed to compute a total score (ranging from to 12).

The self-collected data refer to positive cases, number of Covid-19-related deaths, number of tests and total number of vaccinations administered. These data have been recorded daily since March 2020 from public announcements by official and verified sources. A total of 94 time points were considered in the present study, corresponding to weekly data (Monday was used as a reference). Three quantitative indicators were derived, a positivity index (#cases / #tests), a mortality index (#deaths / #cases) and a testing index (#tests / #people). The number of vaccinations is not used in the present study because the vaccination process began in January 2021 and the administration of the booster dose began in September 2021. The final data set consisted of five indicators: two ordinal total scores, and three quantitative indices.

2.2 Data Analysis

A four-step data analysis strategy was adopted. In the first step, the three quantitative variables (positivity rate, mortality rate and testing rate) were transformed into ordinal variables, via a method used in [7] (see Step 1) transformation of continuous variables into ordinal categorical variables, with minimum information loss. Three ordinal variables were derived. In the second step, the five ordinal variables (i.e., the three recoded variables and the two ordinal total scores), were fuzzy-coded into three categories each, using the barycentric coding scheme proposed in [7]. This scheme has been recently evaluated in the context of hierarchical clustering in [7] and was applied with the DIAS Excel add-in [6]. Barycentric coding allows us to convert an m-point ordinal variable into an n-point fuzzy-coded variable [6, 7]. In other words, the transformation of the three quantitative variables into ordinal variables resulted in a generalized 0-1 matrix (fuzzy-coded matrix), where for each variable we obtain the estimated probability for each category. A drawback of the proposed approach is that the ordinal information in the 5 ordinal variables is lost.

The third step involved the application of Correspondence Analysis (CA) on the fuzzy-coded table with the 94 weeks as rows and the fifteen fuzzy categories as columns (see [1] for a similar approach). The number of significant axes was determined based on percentage of inertia explained and the significant points on each

axis were determined based on the values of two statistics that accompany standard CA output; quality of representation (COR) greater than 200 and contribution (CTR) greater than 1000/(n + 1), where n is the total number of categories (i.e., 15 in our case). In the final step, Hierarchical Cluster Analysis (HCA) using Benzecri's chi-squared distance and Ward's linkage criterion [2, 8] was employed to cluster the 94 points (weeks) on the CA axes obtained from the previous step. The number of clusters was determined upon the empirical criterion of the change in the ratio of between-cluster inertia to total inertia, when moving from a partition with r clusters to a partition with r – 1 cluster [8]. Lastly, we interpret the clusters after determining the contribution of each indicator to each cluster. All analyses were conducted with the M.A.D. [Méthodes de l'Analyse des Données] software [5].

3 Results

Correspondance Analysis resulted in four significant axes, which explain 74.91% of the total inertia (Figure 2). For each axis, we describe the main contrast between groups of categories based on their coordinates, COR and CTR values (Figure 3). "Low and moderate mortality rates" and "high factor testing rates" define a pole on the 1st axis, which is opposed to "average and high levels of "individual" measures". On the second axis, "low positivity rate" and "average levels of collective measures" define a pole, while "average and high positivity rate" and "high levels of collective measures" define the opposite pole. The third axis is characterized by "moderate and high mortality rate", "high levels of collective measures" and "average levels of individual measures" that are opposed to "average levels of collective measures". On the fourth axis, "average levels of collective measures" are opposed to "average testing rate" and "high levels of collective measures".

Total Inertia 0,62704				
Axis	Inertia	%Inertia	Cumulative	Histogram
1	0,1739028	27,73	27,73	**
2	0,1136495	18,12	45,86	***************************
3	0,1066425	17,01	62,87	************************
4	0,0755233	12,04	74,91	******************
5	0,0526040	8,39	83,30	************
6	0,0401367	6,40	89,70	**********
7	0,0307749	4,91	94,61	********
8	0,0163050	2,60	97,21	****
9	0,0113139	1,80	99,01	***
10	0,0061758	0,98	100,00	**
11	0,0000054	0,00	100,00	*
12	0,0000036	0,00	100,00	*

Fig. 2 Explained inertia by axis.

ID	Variables	Level	#G1	COR	CTR	#G2	COR	CTR	#G3	COR	CTR	#G4	COR	CTR
VAR01	[illegible]	low	-60	14	1	[illegible]	[illegible]	240	[illegible]	[illegible]	29	11	[illegible]	1
VAR02	[illegible]	average	-31	0	1	[illegible]	[illegible]	150	-173	[illegible]	61	[illegible]	[illegible]	[illegible]
VAR03	[illegible]	high	[illegible]	46	15	[illegible]	[illegible]	[illegible]	[illegible]	1	1	[illegible]	[illegible]	[illegible]
VAR11	[illegible]	low	113	288	12	9	1	1	-172	279	41	[illegible]	[illegible]	2
VAR12	[illegible]	average	[illegible]	[illegible]	45	-111	4	1	[illegible]	[illegible]	[illegible]	-130	[illegible]	8
VAR13	[illegible]	high	[illegible]	[illegible]	21	31	3	1	[illegible]	[illegible]	[illegible]	[illegible]	8	8
VAR21	[illegible]	low	[illegible]	[illegible]	[illegible]	-62	58	10	[illegible]	[illegible]	15	[illegible]	[illegible]	10
VAR22	[illegible]	average	[illegible]	147	59	[illegible]	[illegible]	28	[illegible]	11	6	[illegible]	[illegible]	[illegible]
VAR23	[illegible]	high	[illegible]	[illegible]	[illegible]	[illegible]	7	3	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
VAR31	containment & closure	low	[illegible]	11	1	130	[illegible]	3	-108	31	3	[illegible]	[illegible]	9
VAR32	containment & closure	average	[illegible]	[illegible]	28	[illegible]	[illegible]	122	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
VAR33	containment & closure	high	[illegible]	111	18	[illegible]	[illegible]	84	[illegible]	[illegible]	81	[illegible]	[illegible]	[illegible]
VAR41	health system	low	[illegible]	111	27	[illegible]	[illegible]	45	[illegible]	[illegible]	11	[illegible]	30	11
VAR42	health system	average	[illegible]	[illegible]	31	-29	3	6	[illegible]	[illegible]	[illegible]	63	1	3
VAR43	health system	high	[illegible]	[illegible]	26	[illegible]	41	4	-162	[illegible]	[illegible]	-40	[illegible]	4

Fig. 3 Category coordinates on the four CA axes (#G), quality of representation (COR) and contribution (CTR). COR values greater than 200 and CTR values greater than 1000 / 16 = 62.5 are shown in yellow. Positive coordinates are shown in green and negative in pink.

Hierarchical Cluster Analysis on the factor scores resulted in seven clusters using the empirical criterion for cluster determination (see Section 2.2). The corresponding dendrogram is shown in Figure 4. The seven nodes in the figure that correspond to the seven clusters are 182, 181, 175, 177, 171, 181, 133 and 179. Cluster content reflects the different periods (phases) presented in the introductory section.

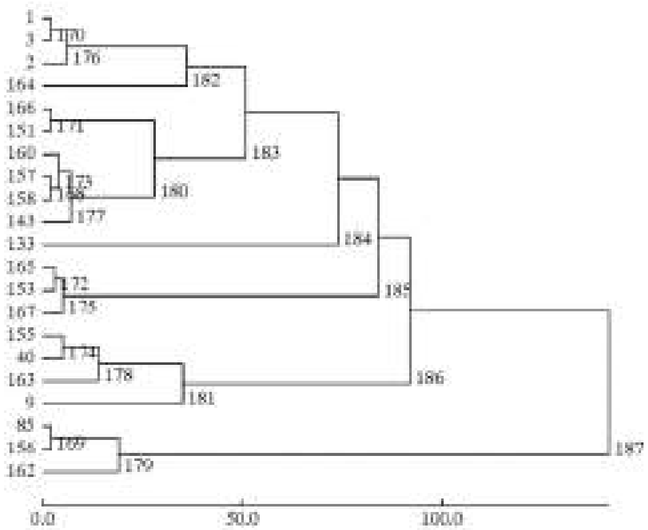


Fig. 4 Dendrogram of the HCA.

The first cluster (182) combines data points from March 2020, the onset of the pandemic with data points from a period following the summer of 2020 (October and November). This cluster is characterized by high positivity rate, low testing rate, high levels of "collective" measures (containment & closure) and low levels of "individual" measures (health system). The second cluster (181) contains data points from April and May 2020 and is characterized by low positivity rate, average to high mortality rate, low testing rate, high levels of "collective" measures (containment & closure) and average levels of "individual" measures (health system). The third cluster (175) combines summer months of 2020 and 2021. This cluster is characterized by low positivity rate, low testing rate and average levels of "collective" measures (containment & closure). The fourth cluster (177) marks the period of December 2020 and the period of spring of 2021, with average positivity rate and high levels of "collective" measures (containment & closure). The fifth cluster (171) refers to the period from December 2020 to February 2021, but also includes August 2021, with high levels of "collective" measures (containment & closure). The sixth cluster (133) refers to the period following the summer of 2021 (September and October 2021). In this cluster, average positivity rates were observed but also strict containment and closure measures.

Lastly, the seventh cluster (179) refers to November and December 2021, including also January 2022, with high positivity and high testing rates, while high levels of containment and closure and health system measures were observed. Figure 5 shows the contributions of each indicator in each cluster.

INT	Variable	Level	Cluster						
			133	171	175	177	179	181	182
VAR01	cases/tests	low			2.6888			3.6138	
VAR02	cases/tests	average	5.884			6.884			
VAR03	cases/tests	high					1.5892		13.556
VAR11	deaths/cases	low							
VAR12	deaths/cases	average						10.5462	
VAR13	deaths/cases	high						9.2727	
VAR21	tests/people	low			1.9732			2.6465	2.3519
VAR22	tests/people	average							
VAR23	tests/people	high					21.3527		
VAR31	containment & closure	low							
VAR32	containment & closure	average			6.8689				
VAR33	containment & closure	high	3.8875	2.2049		1.7047	2.5037	2.3608	1.971
VAR41	health system	low							10.3851
VAR42	health system	average						12.7955	
VAR43	health system	high					1.8392		

Fig. 5 Cluster description (contribution values of the indicators in each cluster - node).

4 Discussion

Based on the study results, we can argue that, when it comes to measures and real time data following a situation such as the pandemic, "the chicken and egg" dilemma arises. The question is whether "collective" and "individual" measures

affect daily incidence data or the inverse (i.e., that the daily data lead to measures). We conclude that in fact the two should be perceived as working in conjunction and not independently from one another. The analysis showed that lower positivity rate is accompanied by average levels of measures from the government at both the "individual" and the "collective" level. Furthermore, higher positivity rate is accompanied by higher levels of measures, as a response. With regard to mortality rate, we observed that higher mortality invokes higher levels of "collective" measures and average levels of "individual" measures, whereas average levels of "collective" measures are associated with higher mortality rate.

It is therefore evident that when it comes to decision making in crisis situations, a systematic collection, analysis and use of data is linked to more effective government response overall. Therefore, evidence-based policy making should be linked to crisis management. This paper presents a first attempt to capture an ongoing phenomenon and therefore it is crucial that the collection and analysis of data will be complemented until the end of the phenomenon.

References

1. Aşan, Z., Greenacre, M.: Biplots of fuzzy coded data. Fuzzy Sets and Systems, **183**(1), 57–71 (2011)
2. Benzècri, J. P.: L'Analyse des Données. 2. L'Analyse des Correspondances. Dunod, Paris (1973)
3. Chadjipadelis, T.: Facebook profile (2022). https://www.facebook.com/theodore.chadjipadelis
4. Hale, T., Petherick, A., Phillips, T., Webster, S.: Variation in government responses to COVID-19. Blavatnik School of Government Working Paper, 31, 2020-11 (2020)
5. Karapistolis, D.: Software Method of Data Analysis MAD. (2010) http://www.pylimad.gr/
6. Markos, A., Moschidis, O., Chadjipadelis, T.: Hierarchical clustering of mixed-type data based on barycentric coding (2022) https://arxiv.org/submit/4142768
7. Moschidis, O., Chadjipadelis, T.: A method for transforming ordinal variables. In: Palumbo, F., Montanari, A., Vichi, M. (eds) Data Science, pp. 285-294. Studies in Classification, Data Analysis, and Knowledge Organization. Springer, Cham. (2017) https://doi.org/10.1007/978-3-319-55723-6_22
8. Papadimitriou, G., Florou, G.: Contribution of the Euclidean and chi-square metrics to determining the most ideal clustering in ascending hierarchy (in Greek). In Annals in Honor of Professor I. Liakis, 546-581. University of Macedonia, Thessaloniki (1996)

pcTVI: Parallel MDP Solver Using a Decomposition into Independent Chains

Jaël Champagne Gareau, Éric Beaudry, and Vladimir Makarenkov

Abstract Markov Decision Processes (MDPs) are useful to solve real-world probabilistic planning problems. However, finding an optimal solution in an MDP can take an unreasonable amount of time when the number of states in the MDP is large. In this paper, we present a way to decompose an MDP into Strongly Connected Components (SCCs) and to find dependency chains for these SCCs. We then propose a variant of the Topological Value Iteration (TVI) algorithm, called *parallel chained TVI* (pcTVI), which is able to solve independent chains of SCCs in parallel leveraging modern multicore computer architectures. The performance of our algorithm was measured by comparing it to the baseline TVI algorithm on a new probabilistic planning domain introduced in this study. Our pcTVI algorithm led to a speedup factor of 20, compared to traditional TVI (on a computer having 32 cores).

Keywords: Markov decision process, automated planning, strongly connected components, dependancy chains, parallel computing

1 Introduction

Automated planning is a branch of Artificial Intelligence (AI) aiming at finding optimal plans to achieve goals. One example of problems studied in automated planning is the electric vehicle path-planning problem [1]. Planning problems with non-deterministic actions are known to be much harder to solve. Markov Decision

Jaël Champagne Gareau (✉)
Université du Québec à Montréal, Canada, e-mail: champagne_gareau.jael@uqam.ca

Éric Beaudry
Université du Québec à Montréal, Canada, e-mail: beaudry.eric@uqam.ca

Vladimir Makarenkov
Université du Québec à Montréal, Canada, e-mail: makarenkov.vladimir@uqam.ca

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_12

Processes (MDPs) are generally used to solve such problems leading to probabilistic models of applicable actions [2].

In probabilistic planning, a solution is generally a policy, i.e., a mapping specifying which action should be executed in each observed state to achieve an objective. Usually, dynamic programming algorithms such as Value Iteration (VI) are used to find an optimal policy [3]. Since VI is time-expensive, many improvements have been proposed to find an optimal policy faster, using for example the Topological Value Iteration (TVI) algorithm [4]. However, very large domains often remain out of reach. One unexplored way to reduce the computation time of TVI is by taking advantage of the parallel architecture of modern computers and by decomposing an MDP into independent parts which could be solved concurrently.

In this paper, we show that state-of-the-art MDP planners such as TVI can run an order of magnitude faster when considering task-level parallelism of modern computers. Our main contributions are as follows:

- An improved version of the TVI algorithm, *parallel-chained TVI* (pcTVI), which decomposes MDPs into independent chains of strongly connected components and solves them concurrently.
- A new parametric planning domain, *chained-MDP*, and an evaluation of pcTVI's performance on many instances of this domain compared to the VI, LRTDP [5] and TVI algorithms.

2 Related Work

Many MDP solvers are based on the Value Iteration (VI) algorithm [3], or more precisely on asynchronous variants of VI. In asynchronous VI, MDP states can be backed up in any order and do not need to be considered the same number of times. One way to take advantage of this is by assigning a priority to every state and by considering them in priority order.

Several state-of-the-art MDP algorithms have been proposed to increase the speed of computation. Many of them are able to focus on the most promising parts of MDP through heuristic search algorithms such as LRTDP [5] or LAO* [6]. Some other MDP algorithms use partitioning methods to decompose the state-space in smaller parts. For example, the P3VI (Partitioned, Prioritized, Parallel Value Iteration) algorithm partitions the state-space, uses a priority metric to order the partitions in an approximate best solving order, and solves them in parallel [7]. The biggest disadvantage of P3VI is that the partitioning is done on a case-by-case basis depending on the planning domain, i.e., P3VI does not include a general state-space decomposition method. The inter-process communication between the solving threads also incurs an overhead on the computation time. The more recent TVI (Topological Value Iteration) algorithm [4] also decomposes the state-space, but does it by considering the topological structure of the underlying graph of the MDP, making it more general than P3VI. Unfortunately, to the best of our knowledge, no parallel version of TVI has been proposed in the literature.

3 Problem Definition

There exist different types of MDP, including Finite-Horizon MDP, Infinite-Horizon MDP and Stochastic Shortest Path MDP (SSP-MDP) [2]. The first two of them can be viewed as special cases of SSP-MDP [8]. In this work, we focus on SSP-MDPs, which we describe formally in Definition 1 below.

Definition 1 A *Stochastic Shortest Path MDP* (SSP-MDP) is given by a tuple (S, A, T, C, G), where:

- S is a finite set of states;
- A is a finite set of actions;
- $T\colon S \times A \times S \to [0, 1]$ is a transition function, where $T(s, a, s')$ is the probability of reaching state s' when applying action a while in state s;
- $C\colon S \times A \to \mathbb{R}^+$ is a cost function, where $C(s, a)$ gives the cost of applying the action a while in state s;
- $G \subseteq S$ is the set of goal states (which can be assumed to be sink states).

We generally search for a policy $\pi\colon S \to A$ that tells us which action should be executed at each state, such that an execution following the actions given by π until a goal is reached has a minimal expected cost. This expected cost is given by a value function $V^\pi\colon S \to \mathbb{R}$. The Bellman Optimality Equations are a system of equations satisfied by any optimal policy.

Definition 2 The Bellman Optimality Equations are the following:

$$V(s) = \begin{cases} 0, & \text{if } s \in G, \\ \min\limits_{a \in A} \left[C(s, a) + \sum\limits_{s' \in S} T(s, a, s') V(s) \right], & \text{otherwise.} \end{cases}$$

The expression between square brackets is called the *Q-value* of a state-action pair:

$$Q(s, a) = C(s, a) + \sum_{s' \in S} T(s, a, s') V(s).$$

When an optimal value function $V^\star$ has been computed, an optimal policy $\pi^\star$ can be found greedily:

$$\pi^\star(s) = \operatorname{argmin}_{a \in A} Q^\star(s, a).$$

Most MDP solvers are based on dynamic programming algorithms like Value Iteration (VI), which update iteratively an arbitrarily initialized value function until convergence with a given precision ϵ. In the worst case, VI needs to do $|S|$ sweeps of the state space, where one sweep consists in updating the value estimate of every state using the Bellman Optimality Equations. Hence, the number of state updates (called a *backup*) is $O(|S|^2)$. When the MDP is acyclic, most of these backups are wasteful, since the MDP can in this situation be solved using only $|S|$ backups (ordered in reverse topological order), thus allowing one to find an optimal policy in $O(|S|)$ [8].

4 Parallel-chained TVI

In this section, we describe an improvement to the TVI algorithm, named pcTVI (Parallel-Chained Topological Value Iteration), which is able to solve an MDP in parallel (as P3VI). pcTVI uses the decomposition proposed by TVI, known to give good performance on many planning domains. We start by summarizing how the original TVI algorithm works.

First, TVI uses Kosaraju's graph algorithm on a given MDP to find the strongly connected components (SCCs) of its graphical structure (the graph corresponding to its all-outcomes determinization).The SCCs are found by Kosaraju's algorithm in reverse topological order, which means that for every $i < j$, there is no path from a state in the i^{th} SCC to a state in the j^{th} SCC. This property ensures that every SCC can be solved separately by VI sweeps if previous SCCs (according to the reverse topological order) have already been solved. The second step of TVI is thus to solve every SCC one by one in that order. Since TVI divides the MDP in multiple subparts, it maximizes the usefulness of every state backup by ensuring that only useful information (i.e., converged state values) is propagated through the state-space.

Unfortunately, TVI can only solve one SCC at a time. Since modern computers have many computing units (cores) which can work in parallel, we could theoretically solve many SCCs in parallel to greatly reduce computation time. Instead of choosing SCCs to solve in parallel arbitrarily or using a priority metric (as in P3VI), which incur a computational overhead to propagate the values between the threads, we want to consider their topological order (as in TVI) to minimize redundant or useless computations. One way to share the work between the processes is to find independent chains of SCCs which can be solved in parallel. The advantage of independent chains is that no coordination and communication is needed between the SCCs, which both removes some running-time overhead and simplifies the implementation.

The Parallel-Chained TVI algorithm we propose (Algorithm 1) works as follows. First, we find the graph G corresponding to the graphical structure of the MDP, decompose it into SCCs, and find the reverse topological order of the SCCs (as in TVI, but we use Tarjan's algorithm instead of Kosaraju's algorithm since it is about twice as fast). We then build the condensation of the graph G, i.e., the graph G_c whose vertices are SCCs of G, where an edge is present between two vertices scc_1 and scc_2 if there exists an edge in G between a state $s_1 \in scc_1$ and a state $s_2 \in scc_2$. We also store the reversed edges in G_c and a counter c_{scc} on every vertex scc which indicates how many incoming neighbors have not yet been computed. We use this (usually small) graph G_c to detect which SCCs are ready to be considered (the SCCs whose incoming neighbors have all been determined with precision ϵ, i.e., the SCCs whose associated counter c_{scc} is 0). When a new SCC is ready, it is inserted into a work queue from which the waiting threads acquire their next task.

Algorithm 1 Parallel-Chained Topological Value Iteration

```
1:  procedure pcTVI(M: MDP, t: Number of threads)
2:      ▷ Find the SCCs of M
3:      G ← Graph(M)                          ▷ G implicitly shares the same data structures as M
4:      SCCs ← Tarjan(G)                      ▷ SCCs are found in reverse topological order
5:
6:      ▷ Build the graph of SCCs of G
7:      G_c ← GraphCondensation(G, SCCs)
8:
9:      ▷ Solve in parallel independent SCCs
10:     Pool ← CreateThreadPool(t)                                   ▷ Create t threads
11:     V ← NewValueFunction()                ▷ Arbitrarily initialized; Shared by all threads
12:     Q ← CreateQueue()                                            ▷ Shared by all threads
13:     Insert(Q, Head(SCCs))                 ▷ The goal SCC is inserted in the queue
14:     while NotEmpty(Q) do                                         ▷ Only one thread runs this loop
15:         scc ← ExtractNextItem(Q)
16:         for all neighbor ∈ Neighbors(scc) do
17:             Decrement NumIncomingNeighbors(neighbor)
18:             if NumIncomingNeighbors(neighbor) = 0 then
19:                 AssignTaskToAvailableThread(Pool, PartialVI(M, V, scc))
20:                 Push(Q, scc)              ▷ Neighbors of scc are ready to be considered next
21:             end if
22:         end for
23:     end while
24:
25:     ▷ Compute and return an optimal policy using the computed value function
26:     Π ← GreedyPolicy(V)
27:     return Π
28: end procedure
```

5 Empirical Evaluation

In this section, we evaluate empirically the performance of pcTVI, comparing it to the three following algorithms: (1) VI – the standard dynamic programming algorithm (here we use its asynchronous round-robin variant), (2) LRTDP – a well-known heuristic search algorithm, and (3) TVI – the Topological Value Iteration algorithm described in Section 4. In the case of LRTDP, we carried out the admissible and domain-independent $h_{\min}$ heuristic, first described in the original paper introducing LRTDP [5]:

$$h_{\min}(s) = \begin{cases} 0, & \text{if } s \in G. \\ \min\limits_{a \in A_s} \left[C(s,a) + \min\limits_{s' \in succ_a(s)} V(s') \right], & \text{otherwise,} \end{cases}$$

where A_s denotes the set of applicable actions in state s and $succ_a(s)$ is the set of successors when applying action a at state s. The four competing algorithms (VI, TVI, LRTDP and pcTVI) were implemented in C++ by the authors of this paper and compiled using the GNU g++ compiler (version 11.2). All tests were performed on a

computer equipped with four Intel Xeon E5-2620V4 processors (each of them having 8 cores at 2.1 GHz, for a total of 32 cores). For every test domain, we measured the running time of the four compared algorithms carried out until convergence to an ϵ-optimal value function (we used $\epsilon = 10^{-6}$). Every domain was tested 15 times with randomly generated MDP instances. To minimize random factors, we report the median values obtained over these 15 MDP instances.

Since there is no standard MDP domain in the scientific literature suitable to benchmark a parallel MDP solver, we propose a new general parametric MDP domain that we use to evaluate the algorithms. This domain, which we call chained-MDP, uses 5 parameters: (1) k, the number of independent chains $\{c_1, c_2, \ldots, c_k\}$ in the MDP; (2) n_{scc}, the number of SCCs $\{scc_{i,1}, scc_{i,2}, \ldots, scc_{i,n_{scc}}\}$ in every chain c_i; (3) n_{sps}, the number of states per SCC; (4) n_a the number of applicable actions per state, and (5) n_e the number of probabilistic effects per action. The possible successors $succ(s)$ of a state s in $scc_{i,j}$ are states in $scc_{i,j}$ and either the states in $scc_{i,j+1}$ if it exists, or the goal state otherwise. When generating the transition function of a state-action pair (s, a), we sampled n_e states uniformly from $succ(s)$ with random probabilities. In each of our tests, we used $n_{scc} = 2$, $n_a = 5$ and $n_e = 5$. A representation of a Chained-MDP instance is shown in Figure 1.

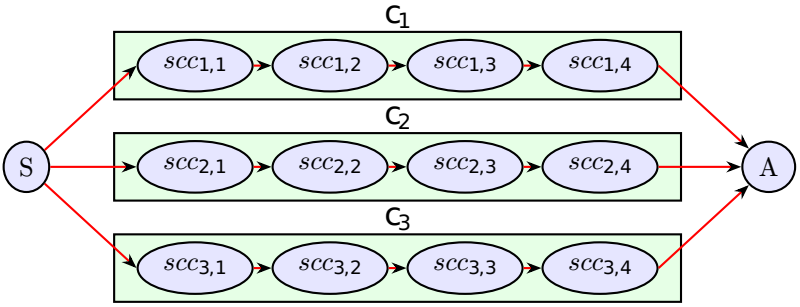


Fig. 1 A chained-MDP instance where $n_c = 3$ and $n_{scc} = 4$. Each ellipse represents a strongly connected component.

Figure 2 presents the obtained results for the Chained-MDP domain when varying the number of states and fixing the number of chains (32). We can observe that when the number of states is small, pcTVI does not provide an important advantage over the existing algorithms since the overhead of creating and managing the threads is taking most of the possible gains. However, as the number of states increases, the gap in the running time between pcTVI and the three other algorithms increases. This indicates that pcTVI is particularly useful on very large MDPs, which are usually needed when considering real-world domains.

Figure 3 presents the obtained results for the same Chained-MDP domain when varying the number of chains and fixing the number of states (1M). When the number of chains increases, the total number of SCCs implicitly increases (which also implies the number of states per SCC decreases). This explains why each tested

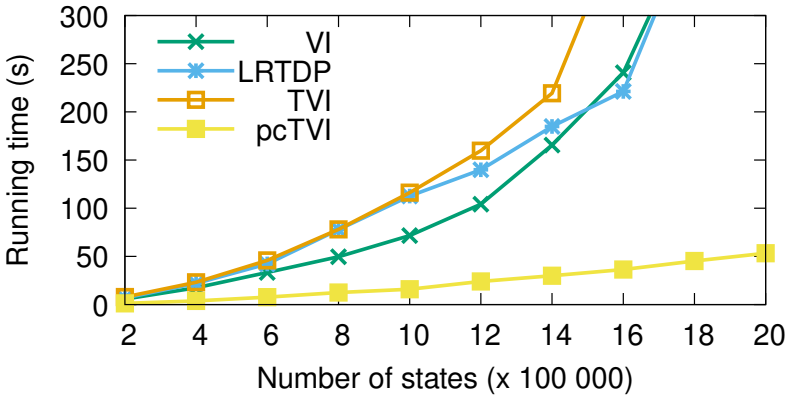


Fig. 2 Average running times (in s) for the Chained-MDP domain with varying number of states and fixed number of chains (32).

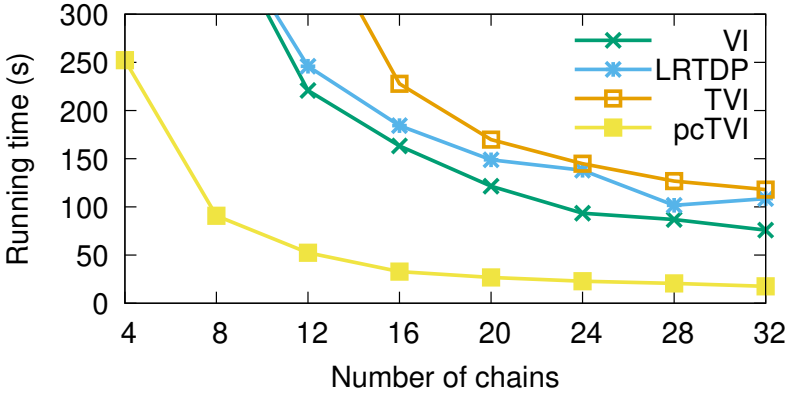


Fig. 3 Average running times (in s) for the Chained-MDP domain with varying number of chains and fixed number of states (1M).

algorithms becomes faster (TVI becomes faster by design, since it solves SCCs one-by-one without doing useless state backups, and VI and LRTDP become faster due to an increased locality of the considered states in memory, which improves cache performance). The performance of pcTVI increases as the number of chains increases (for the same reason as the others algorithms, but also due to increased parallelization opportunities). We can also observe that for domains with 4 chains only, pcTVI still clearly outperforms the other methods. This means that pcTVI does not need a highly parallel server CPU and can be used on standard 4-core computer.

6 Conclusion

The main contributions of this paper are two-fold. First, we presented a new algorithm, pcTVI, which is, to the best of our knowledge, the first MDP solver that takes into account both the topological structure of the MDP (as in TVI) and the parallel capacities of modern computers (as in P3VI). Second, we introduced a new parametric planning domain, Chained-MDP, which models any situation where different strategies (corresponding to a chain) can reach a goal, but where, once committed to a strategy, it is not possible to switch to a different one. This domain is ideal to evaluate the parallel performance of an MDP solver. Our experiments indicate that pcTVI outperforms the other competing methods (VI, LRTDP, and TVI) on every tested instance of the Chained-MDP domain. Moreover, pcTVI is particularly effective when the considered MDP has many SCC chains (for increased parallelization opportunities) of large size (for decreased overhead of assigning small tasks to the threads). As future work, we plan to investigate ways of pruning provably suboptimal actions, which would allow more SCCs to be found. While this paper focuses on the automated planning side of MDPs, the proposed optimization and parallel computing approaches could also be applied when using MDPs with Reinforcement Learning and other ML algorithms.

Acknowledgements This research has been supported by the *Natural Sciences and Engineering Research Council of Canada* (NSERC) and the *Fonds de Recherche du Québec — Nature et Technologies* (FRQNT).

References

1. Champagne Gareau, J., Beaudry E., Makarenkov, V.: A fast electric vehicle planner using clustering. In: Stud. in Classif., Data Anal., and Knowl. Organ., **5**, 17-25. Springer (2021)
2. Mausam, Kolobov, A.: Planning with Markov Decision Processes: An AI Perspective. Morgan & Claypool (2012)
3. Bellman, R.: Dynamic Programming. Prentice Hall (1957)
4. Dai, P., Mausam, Weld, D. S., Goldsmith, J.: Topological value iteration algorithms. J. Artif. Intell. Res., **42**, 181-209 (2011)
5. Bonet, B., Geffner, H.: Labeled RTDP: Improving the convergence of real-time dynamic programming. In: Proc. of ICAPS, pp. 12-21 (2013)
6. Hansen, E., Zilberstein, S.: LAO*: A heuristic search algorithm that finds solutions with loops. Artif. Intell., **129**(1-2), 35-62 (2001)
7. Wingate, D., Seppi, K.: P3VI: A partitioned, prioritized, parallel value iterator. In: Proc. of the Int. Conf. on Mach. Learn. (ICML), 863-870 (2004)
8. Bertsekas, D.: Dynamic Programming and Optimal Control, vol. 2. Athena scientific Belmont, MA (2001)

Three-way Spectral Clustering

Cinzia Di Nuzzo and Salvatore Ingrassia

Abstract In this paper, we present a spectral clustering approach for clustering *three-way data*. Three-way data concern data characterized by three modes: n units, p variables, and t different occasions. In other words, three-way data contain a $t \times p$ observed matrix for each statistical observation. The units generated by simultaneous observation of variables in different contexts are usually structured as three-way data, so each unit is basically represented as a matrix. In order to cluster the n units in K groups, the spectral clustering application to three-way data can be a powerful tool for unsupervised classification. Here, one example on real three-way data have been presented showing that spectral clustering method is a competitive method to cluster this type of data.

Keywords: spectral clustering, kernel function, three-way data

1 Introduction

Spectral clustering methods are based on the graph theory, where the units are represented by the vertices of an undirected graph and the edges are weighted by the pairwise similarities coming from a suitable kernel function, so the clustering problem is reformulated as a graph partition problem, see e.g. [16, 6]. The spectral clustering algorithm is a very powerful method for finding non-convex clusters of data, moreover, it is a handy approach for handling high-dimensional data since it works on a transformation of the raw data having a smaller dimension than the space of the original data.

Cinzia Di Nuzzo (✉)
Department of Statistics, University of Roma La Sapienza, Piazzale Aldo Moro, 5, 00185 Roma, Italy, e-mail: cinzia.dinuzzo@uniroma1.it

Salvatore Ingrassia
Department of Economics and Business, University of Catania, Piazza Università, 2, 95131 Catania, Italy, e-mail: s.ingrassia@unict.it

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_13

Three-way data derives from the observation of various attributes measured on a set of units in different situations; some examples are longitudinal data on multiple response variables and multivariate spatial data. Three-way data can also derive from temporal measurements of a feature vector, thus having the dataset composed of three modes: n units (matrices), p variables (columns), and t times (rows). Clustering of three-way data has attracted a growing interest in literature, see e.g. [14], [1]; model-based clustering of three-way data has been introduced by [15] in the framework of matrix-variate normal mixtures; recent papers include [9] handle on parsimonious models for modeling matrix data; [11] introduce two matrix-variate distributions, both the elliptical heavy-tailed generalization of the matrix-variate normal distribution; [12] deal with three-way data clustering using matrix-variate cluster-weighted models (MV-CWM); and, [13] consider an application to educational data via mixtures of parsimonious matrix-normal distribution.

In this paper, we present a spectral clustering approach for clustering *three-way data* and a suitable kernel function between matrices is introduced. As a matter of fact, the data matrices represent the vertices of the graph, consequently, the edges must be weighted by a single value.

The rest of the paper is organized as follows: in Section 2 the spectral clustering method is summarized; in Section 3 a method to select the parameters in the spectral clustering algorithm is described; in Section 4 the three-way spectral clustering with a new kernel function are introduced; in Section 5 an application based on real three-way data is presented. Finally, in Section 5 we provide concluding remarks.

2 Spectral Clustering

Spectral clustering algorithm for two-way data has been described in [8, 16, 6]. Here, we summarize the main step of this algorithm.

Let $V = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ be a set of points in $\mathcal{X} \subseteq \mathbb{R}^p$. In order to group the data V in K cluster, the first step concerns the definition of a symmetric and continuous function $\kappa : \mathcal{X} \times \mathcal{X} \to [0, \infty)$ called the *kernel function*. Afterwards, a *similarity matrix* $W = (w_{ij})$ can be assigned by setting $w_{ij} = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$, for $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}$. and finally the *normalized graph Laplacian* matrix $L_{\text{sym}} \in \mathbb{R}^{n \times n}$ is introduced

$$L_{\text{sym}} = I - D^{-1/2} W D^{-1/2}, \tag{1}$$

where $D = \text{diag}(d_1, d_2, \ldots, d_n)$ is the *degree matrix* and d_i is the *degree* of the vertex $\boldsymbol{x}_i$ defined as $d_i = \sum_{j \neq i} w_{ij}$ and I denotes the $n \times n$ identity matrix. The Laplacian matrix L_{sym} is positive semi-definite with n non-negative eigenvalues. For a fixed $K \ll n$, let $\{\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_K\}$ be the eigenvectors corresponding to the smallest K eigenvalues of L_{sym}. Then, the *normalized Laplacian embedding in the K principal subspace* is defined as the map $\boldsymbol{\Phi}_{\boldsymbol{\Gamma}} : \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\} \to \mathbb{R}^K$ given by

$$\boldsymbol{\Phi}_{\boldsymbol{\Gamma}}(\boldsymbol{x}_i) = (\gamma_{1i}, \ldots, \gamma_{Ki}), \quad i = 1, \ldots, n,$$

where $\gamma_{1i}, \ldots, \gamma_{Ki}$ are the i-th components of $\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_K$, respectively. In other words, the function $\Phi_{\boldsymbol{\Gamma}}(\cdot)$ maps the data from the input space $\mathcal{X}$ to a feature space defined by the K principal subspace of L_{sym}. Afterwards, let $\boldsymbol{Y} = (\boldsymbol{y}_1', \ldots, \boldsymbol{y}_n')$ be the $n \times K$ matrix given by the embedded data in the feature space, where $\boldsymbol{y}_i = \Phi_{\boldsymbol{\Gamma}}(\boldsymbol{x}_i)$ for $i = 1, \ldots, n$. Finally, the embedded data $\boldsymbol{Y}$ are clustered according to some clustering procedure; usually, the k-means algorithm is taken into account in literature. However, to this end Gaussian mixtures have been proposed because they yield elliptical cluster shapes, i.e. more flexible cluster shapes with respect to the k-means, see [2]. Finally, we point out that the performances of other mixture models based on non-Gaussian component densities have been analyzed, but Gaussian mixture models can be considered as a good trade-off between model simplicity and effectiveness, see [3] for details.

3 A Graphical Approach for Parameter Selection

According to spectral clustering algorithm introduced in Section 2, the spectral approach requires to set: *i*) the number of clusters K, *ii*) the kernel function κ (with the corresponding parameter). In order to select these quantities, in the following we summarize the method proposed in [4].

To begin with, we point out that the choice of the kernel function affects the entire data structure in the graph, and consequently, the structure of the Laplacian matrix and its eigenvectors. An optimal kernel function should lead to a similarity matrix W having (as much as possible) diagonal blocks: in this case, we get well-separated groups and we are also able to understand the number of groups in that data set by counting the number of blocks. For the sake of simplicity, we consider here the self-tuning kernel introduced by [17]

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{\epsilon_i \epsilon_j}\right) \tag{2}$$

with $\epsilon_i = \|\boldsymbol{x}_i - \boldsymbol{x}_h\|$, where $\boldsymbol{x}_h$ is the h-th neighbor of point $\boldsymbol{x}_i$ (similarly for ϵ_j). This function allow to get a similarity matrix that does not depend on any parameter so that the algorithm of spectral clustering will be based on the pairwise proximity between units. On the contrary, we need to select the h-th neighbor of the unit in (2).

The main novelty of the joint-graphical approach concerns the analysis of some graphic features of the Laplacian matrix including the shape of the embedded space. Indeed, the embedded data provide useful information for the clustering, in particular the main results in [10] and [5] allow to deduce that if the embedded data assume a cones structure, then the number of clusters is equal to the number of the cones/spikes in the feature space; furthermore, a clearer clustering structure emerges when the spikes are narrower and well separated.

The idea behind the graphical approach is to select the number K of groups and the parameter h in the kernel function from a joint analysis of three main characteristics: the plot of the Laplacian matrix; the maxima values of the eigengaps between two

consecutive eigenvalues; the scatter plot of the mapped data in the feature space and in particular the number of spikes counted in the embedded data space.

We remark that we cannot analyze all possible values of $h \in \{1, 2, \ldots, n-1\}$ and hence we choose a suitable subset $\mathcal{H} \subset \{1, 2, \ldots, n-1\}$, in particular we choose $\mathcal{H} = \{1\%, 2\%, 5\%, 10\%, 15\%, 20\%\} \times n \subset \{1, 2, \ldots, n-1\}$, and select $h \in \mathcal{H}$, see the following procedure for details.

Parameter selection (K and h)

Input: data set V, kernel function κ, $\mathcal{H}$.

1. For each h in $\mathcal{H}$, compute the matrix M_s and analyze the block structure in the greyscale plot of M_s.
2. For each h in $\mathcal{H}$, plot the embedded data in the feature space and analyze the shape of the cone structure.
3. If the number of blocks in Step 1 is equal to the number of spikes in Step 2, then set K equal to the number of blocks. Go to Step 5.
4. Otherwise, analyze the eigengap plot.
   a. If this plot shows a unique maximum eigengap for each $h \in \mathcal{H}$, then set K according to this maximum. Go to Step 5.
   b. If this plot shows multiple maxima for different $h \in \mathcal{H}$, select the number of clusters K not to be smaller than the number of tight spikes in the corresponding plot of the embedded data.
5. Select $h \in \mathcal{H}$ such that the clearest orthogonal data structure emerges from the plot of the embedded data.
6. Stop.

Output: K, h.

4 Three-way Spectral Clustering

In this section, we propose a spectral approach for clustering three-way data. Three-way data consists of a data set referring to the same sets of units and variables, observed in different situations, i.e., a set of multivariate matrices, that can be organized in three modes: n units, p variables, and t situations. Therefore, given n matrices that represent the vertices of the graph, each matrix is composed by p columns that represent our variables and t rows that represent the time or another feature. So we have a tensor of dimension $n \times t \times p$, thus the dataset is a tensor $\{\boldsymbol{X}\}_{isk}$ for $i = 1, \ldots, n$, $s = 1, \ldots, t$, $k = 1, \ldots, p$.

We define a distance function δ_M between two matrices $A, B \in \mathbb{R}^{p \times t}$ such that $\delta_M : R^{t \times p} \times R^{t \times p} \to [0, +\infty)$ is defined as

$$\delta_M(A,B) := \|A-B\|_F = \sqrt{\sum_{s=1}^{t}\sum_{k=1}^{p}|a_{sk}-b_{sk}|^2} \tag{3}$$

where $\|\cdot\|_F$ is Frobenius norm[1]. Thus the distance between two units in the matrix data $\boldsymbol{X}$ is equal to

$$\delta_M(X_{i_1sk}, X_{i_2sk}) = \sqrt{\sum_{s=1}^{t}\sum_{k=1}^{p}|X_{i_1sk}-X_{i_2sk}|^2}, \qquad \text{for } i_1, i_2 = 1,\ldots,n. \tag{4}$$

For simplicity, in the following, we denote $\delta_M(X_{i_1sk}, X_{i_2sk})$ by $\delta_M(i_1,i_2)$. Moreover, we define the three-way self-tuning kernel function as

$$\kappa_S : \boldsymbol{X}\times\boldsymbol{X} \to [0,+\infty), \qquad \kappa_S(i_1,i_2) = \exp\left(-\frac{\delta_M(i_1,i_2)}{\epsilon_{i_1}\epsilon_{i_2}}\right) \tag{5}$$

where ϵ_{i_1} and ϵ_{i_2} need to be selected like in the kernel defined in (2).

Afterwards, we compute the similarity matrix W given by $w_{i_1i_2} = \kappa(i_1,i_2)$, so that we can apply the spectral clustering algorithm.

Finally, we point out that, differently from approaches based on mixtures of matrix-variate data, the number of variables of the data set is not a critical issue because the spectral clustering algorithm is based on distance measures.

5 A Real Data Application

We apply the three-way spectral clustering to the analysis of the Insurance data set, available in the `splm` R package. This dataset was initially introduced by [7] and has recently been analyzed by [12]. The goal is to study the consumption of non-life insurance during the years 1998-2002 in the 103 Italian provinces, so $t = 5$ and $n = 103$. As regards the number of variables, we consider all the variables contained in the data set, so $p = 11$. Thus, we have 103 matrices of dimensions 5×11.

The 103 Italian provinces are divided into north-west (24 provinces), north-east (22 provinces), center (21 provinces), south (23 provinces), and islands (13 provinces).

As regard the choice of K and h, we consider the graphical approach introduced in Section 3. In Figure 1 the geometric features of spectral clustering are plotted as h varies. From the number of blocks of the Laplacian matrix (Figure 1-a)), the first maximum eigengap (Figure 1-b)) and the number of spikes in the feature space (Figure 1-c)), we deduce that the number of clusters is $K = 2$. For the selection of

[1] In general, given a matrix $A \in \mathbb{R}^{n\times m}$, with $A = (a_{ij})$ for $i = 1,\ldots,n$ and $j = 1,\ldots,m$. The Frobenius norm is defined by

$$\|A\|_F := \sqrt{\sum_{j=1}^{m}\sum_{i=1}^{n}|a_{ij}|^2}.$$

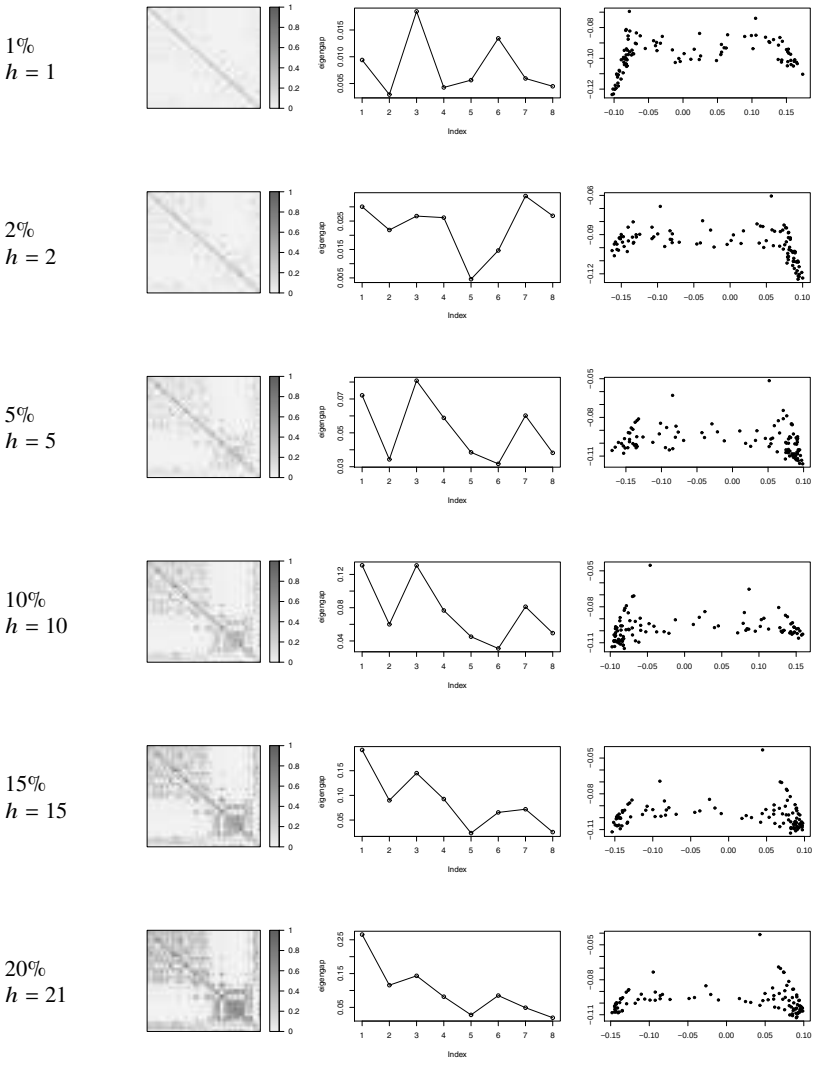


Fig. 1 *Insurance data.* Spectral clustering features: *a*) plot of Laplacian matrix in greyscale; *b*) plot of the first eight eigengap values; *c*) scatterplot of the embedded data along with directions (γ_1, γ_2).

Table 1 *Insurance data.* Table of spectral clustering result.

Cluster 1	NORTHWEST (24 provinces) NORTH EAST (22 provinces) CENTRE (15 provinces)
Cluster 2	CENTRE (6 provinces) SOUTH (23 provinces) ISLANDS (13 provinces)

h we choose indifferently $h = 15$ and $h = 21$ because in these cases the maximum eigengap highlights the maximum values corresponding to $K = 2$. In Table 1 the clustering results are presented. This table shows that only 6 center provinces are classified together with the southern provinces. But to be sure that these provinces are neighboring the south provinces, let us analyze spectral clustering results on the map of Italy. Figure 2-a) illustrates the partition deriving from spectral clustering in the political map of Italy, where Italian regions are described by the yellow lines, while the provinces are by the black lines. The result shows a clear separation between center-north Italy and south-insular Italy, in fact, the center-north has a level of insurance penetration close to the European averages, while the South is less developed economically. However, the Massa-Carrara province should belong to the centre-north group. Moreover, we remark that the Rome province, being the capital of Italy, has one socio-economic development comparable to that of north Italy justifying belonging to the centre-north group.

Furthermore, in Figure 2-b) we also represented the partition produced by MN-CWM proposed in [12], we note that the two clustering results are very similar to each other and differ only for one province of central Italy (precisely for the province of Terni). It should also be emphasized that the dataset analyzed by [12] is different from the one analyzed here, since, to avoid excessive parameterization of the models, the authors select only $p = 5$ variables in the data set.

a) b)

Fig. 2 *Insurance data.* a) Three-way spectral clustering; b) Method proposed by [12].

6 Conclusion

In this paper, a spectral approach to cluster three-way data has been proposed. So the data are organized in a tensor and the vertices in the graph are represented by the matrices of dimension $t \times p$. In order to weigh the matrices in the graph, a kernel function based on the Frobenius norm between the matrix difference has been introduced. The performance of the spectral clustering algorithm has been shown in one real three-way data set. Our method is competitive with respect to other clustering methods proposed in the literature to perform matrix-data clustering. Finally, in order to provide suggestions for future research, other kernel functions can be introduced considering different distances with respect to the Frobenius norm.

Acknowledgements This work was supported by the University of Catania grant PIACERI/CRASI (2020).

References

1. Bocci, L., Vicari, D.: ROOTCLUS: Searching for "ROOT CLUSters" in Three-Way Proximity Data. Psychometrika. **84**, 941–985 (2019)
2. Di Nuzzo, C., Ingrassia, S.: A mixture model approach to spectral clustering and application to textual data. Stat. Meth. Appl. Forthcoming (2022)
3. Di Nuzzo, C.: Model selection and mixture approaches in the spectral clustering algorithm. Ph.D. thesis, Economics, Management and Statistics, University of Messina (2021)
4. Di Nuzzo, C., Ingrassia, S.: A joint graphical approach for model selection in the spectral clustering algorithm. Tech. Rep. (2022)
5. Garcia Trillos, N., Hoffman, F., Hosseini, B.: Geometric structure of graph Laplacian embeddings. arXiv preprint arXiv:1901.10651. (2019)
6. Meila, M.: Spectral clustering. In Hennig, C., Meila, M., Murtagh, F., Rocci, R. (eds.). Handbook of Cluster Analysis. Chapman and Hall/CRC (2015)
7. Millo, G., Carmeci, G.: Non-life insurance consumption in Italy: A sub-regional panel data analysis. J. Geogr. Syst. **12**, 1–26 (2011)
8. Ng, A., Jordan, M., Weiss, Y.: On spectral clustering: Analysis and an algorithm. Adv. Neural Inf. Process. Syst. **14** (2002)
9. Sarkar, S., Zhu, X., Melnykov, V., Ingrassia, S.: On parsimonious models for modeling matrix data. Comput. Stat. Data Anal. **142**, 106822 (2020)
10. Schiebinger, G., Wainwright, M. J., Yu, B.: The geometry of kernelized spectral clustering. Ann. Stat. **43**(2), 819–846 (2015a)
11. Tomarchio, S. D., Punzo, A., Bagnato, L.: Two new matrix-variate distributions with application in model-based clustering. Comput. Stat. Data Anal. **152**, 107050 (2020)
12. Tomarchio, S. D., McNicholas, P., Punzo, A.: Matrix normal cluster-weighted models. J. Classif. **38**, 556-575 (2021)
13. Tomarchio, S. D., Ingrassia, S., Melnykov, V.: Modeling students' career indicators via mixtures of parsimonious matrix-normal distributions. Aust. New Zeal. J. Stat. Forthcoming (2022)
14. Vichi, M., Rocci, R., Kiers, H. A. L.: Simultaneous component and clustering models for three-way data: Within and between approaches. J. Classif. **24**, 71–98 (2007)
15. Viroli, C.: Finite mixtures of matrix normal distributions for classifying three-way data. Stat. Comput. **21**, 511–522 (2011)
16. von Luxburg, U.: A tutorial on spectral clustering. Stat. Comput. **17**(4), 395–416 (2007)
17. Zelnik-Manor, L., Perona, P.: Self-tuning spectral clustering. Adv. Neural Inf. Process. Syst. **17** (2004)

Improving Classification of Documents by Semi-supervised Clustering in a Semantic Space

Jasminka Dobša and Henk A. L. Kiers

Abstract In the paper we propose a method for representation of documents in a semantic lower-dimensional space based on the modified Reduced k-means method which penalizes clusterings that are distant from classification of training documents given by experts. Reduced k-means (RKM) enables simultaneously clustering of documents and extraction of factors. By projection of documents represented in the vector space model on extracted factors, documents are clustered in the semantic space in a semi-supervised way (using penalization) because clustering is guided by classification given by experts, which enables improvement of classification performance of test documents.

Classification performance is tested for classification by logistic regression and support vector machines (SVMs) for classes of Reuters-21578 data set. It is shown that representation of documents by the RKM method with penalization improves the average precision of classification by SVMs for the 25 largest classes of Reuters collection for about 5,5% with the same level of average recall in comparison to the basic representation in the vector space model. In the case of classification by logistic regression, representation by the RKM with penalization improves average recall for about 1% in comparison to the basic representation.

Keywords: classification of textual documents, LSA, reduced k-means

Jasminka Dobša (✉)
Faculty of Organization and Informatics, University of Zagreb, Pavlinska 2, 40000 Varaždin, Croatia, e-mail: jasminka.dobsa@foi.hr

Henk A. L. Kiers
Department of Psychology, University of Groningan, Grote Kruisstraat 2/1, 9712 TS Groningen, The Netherlands, e-mail: h.a.l.kiers@rug.nl

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_14

1 Introduction

There are two main families of methods that deal with representation of documents and words that index them: global matrix factorization methods such as Latent Semantic Analysis (LSA) [2] and local context window methods such as the continuous bag of words (CBOW) model and the continuous skip-gram model [8]. The latter use neural networks for learning of representations of words and are intensively explored lately in the scientific community since the development of fast processors has enabled processing of huge amounts of data which resulted in improvements in performance of wide spectra of text mining and natural language tasks. However, representation of words solely by context window methods has a drawback due to the neglect of information about global corpus statistics [9].

In this paper we propose a method for representation of documents by application of a penalized version of the RKM method [4] on a term-document matrix. The corpus of textual documents is represented by a sparse term-document matrix in which entry (i, j) is equal to the weight of the i-th index term for the j-th document. Weights of terms are given by the TfIdf weighting which utilizes local information about the frequency of the i-th term in the j-th document and global information about usage of the i-th term in the entire collection. A benchmark method that utilizes global matrix factorization on term-document matrices is LSA [2] which uses truncated singular value decomposition (SVD) for representation of terms and documents in lower-dimensional semantic space. SVD does not capture the clustering structure of data which motivates application of the RKM.

The rest of the paper is organized as follows: the second section describes related work on representation of documents and words and methods of dimensionality reduction related to RKM. The third section describes the modified RKM method with penalization, while the fourth section describes an experiment on Reuters-21578 data set. In the last section conclusions and directions for further work are given.

2 Related Work

2.1 Representation by Matrix Factorization Methods

A benchmark method among methods that utilize matrix factorization for representation of textual documents is the method of LSA introduced in 1994 [2]. By LSA a sparse term-document matrix is transformed via SVD into a dense matrix of the same term-document type with representations of words (index terms) and documents in a lower-dimensional space. The idea is to map similar documents, or those that describe the same topics, closer to each other regardless of the terms that are used in them. A very efficient application of LSA is in cross-lingual information retrieval where relevant documents for a query in one language are retrieved from a set of documents in another language [7]. According to our knowledge application

of methods that simultaneously cluster objects and extract factors in the field of text mining is very limited. In [6] a method is proposed for cross-lingual information retrieval based on the RKM method.

2.2 Neural Network Word Embeddings

Another approach is to learn representations of words, or so called embeddings, by using local context windows. In 2003 Bengio and coauthors [1] proposed a neural probabilistic language model that uses simple neural network architecture to learn distributed representations for each word as well as probability functions for word sequences, expressed in terms of these representations. Mikolov and coautors [8] proposed in 2013 two models based on single-layer neural network architectures: the skip gram-model that predicts context words given the current word and the continuous bag of words model which predicts current words based on the context. In 2014 the GloVe model [9] was proposed, based on the critique that neural network models suffer from the disadvantage that they do not utilize co-occurrence statistics of the entire corpus, but scan only context windows of words ignoring vast amounts of repetition in the data. That model exploits the advantages of global matrix factorization methods by utilization of term-term co-occurrence matrices and local context window methods.

Word embedding can be classified as static such as word2vec [8] and GloVe [9], and contextual, such as ELMo [10] and BERT [5]. Contextual representation is introduced in [10] in order to model characteristics of word use (syntax and semantics) on one side and variation in word representation due to the context in which words are appearing.

2.3 Methods for Simultaneous Clustering and Factor Extraction

A standard procedure for clustering of objects in a lower-dimensional space is tandem analysis which includes projection of data by principal components and clustering of data in a lower-dimensional space. Such an approach was criticized in [3] and [4] since principal components may extract dimensions which do not necessarily significantly contribute to the identification of a clustering structure in the data. As a response, De Soete and Carroll proposed the method of RKM [4] which simultaneously clusters data and extracts the factors of variables by reconstructing the original data with only centroids of clusters in a lower-dimensional space. The algorithm of Factorial k-means (FKM) proposed by Vichi and Kiers [13] has the same aim of simultaneous reduction of objects and variables and it reconstructs the data in a lower-dimensional space by its centroids in the same space. The application of the latter method is limited in text mining since the method is limited to cases in which the number of variables is less than the number of cases. In [11] the RKM

and FKM methods are compared using simulations and theoretically in order to identify cases for their application. Timmerman and associates also propose method of Subspace k-means [12] which gives an insight into cluster characteristics in terms of relative positions of clusters given by centroids and the shape of the clusters given by within cluster residuals.

3 Reduced k-Means with Penalization

Let $\mathbf{X}$ be $m \times n$ term-document matrix. We use the following notation:

- $\mathbf{A}$ is an $m \times k$ columnwise orthonormal matrix of extracted factors;
- $\mathbf{M}$ is an $n \times c$ membership matrix, where c is a predefined number of clusters; $m_{ic} = 1$ if object (document) i belongs to cluster c and 0 otherwise;
- $\mathbf{Y}$ is a $c \times k$ matrix which gives centroids of clusters in the lower-dimensional space.

By definition, we suppose that every document in the collection belongs to exactly one cluster. The RKM method minimizes the loss function

$$\mathbf{F(M,A)} = \|\mathbf{X} - \mathbf{A}\mathbf{Y}^T\mathbf{M}^T\|^2 \tag{1}$$

in the least squares sense. The dimension of the lower-dimensional space must be less or equal to the number of clusters. Modified RKM with penalization minimizes the loss function

$$\mathbf{F(M,A)} = \|\mathbf{X} - \mathbf{A}\mathbf{Y}^T\mathbf{M}^T\|^2 + \lambda\|\mathbf{M} - \mathbf{G}\|^2 \tag{2}$$

where $\mathbf{G}$ is $n \times c$ membership matrix based on expert judgements. If c is number of classes then $g_{ic} = 0$ if object (document) i belongs to class c, and 0 otherwise. By the second summand in the loss function we penalize clusterings that are distant from the classes by expert judgements using parameter λ that regularizes the importance of that penalization. We use the alternating least squares (ALS) algorithm analogous to the one in [4] which alternates between corrections of the loading matrix $\mathbf{A}$ in one step and of the membership matrix $\mathbf{M}$ in another. As each of the steps in the ALS algorithm improves the loss function, the algorithm converges to at least a local minimum. By starting the procedure from a large number of random initial estimates and choosing the best solution, the chances of obtaining the global minimum are increased.

4 Experiment

4.1 Design of Experiment

Experiments are conducted for classification on the Reuters-21578 data set, specifically using the ModApte Split which assigns Reuters reports from April 7, 1987 and before to the training set, and after, until end of 1987, to the test set. It consists of 9603 training and 3299 test documents. The collection has 90 classes which contain at least one training and test document. Documents are represented by a bag of words representation. A list of index terms is formed based on terms that appear in at least four documents of the collection, which resulted in a list of 9867 index terms.

Classification is conducted by logistic regression (LR) and SVM algorithm. The basic model is the bag of words representation (full representation), while representations in the lower-dimensional space are obtained by SVD (Latent Sematic Analysis), RKM and RKM with penalization ($\lambda = 0.1, 0.2, 0.4, 0.6$). For RKM and RKM with penalization representations are obtained by applying matrix factorization on the term-document matrix of the training documents, and by projection of test documents on factors given by matrix A in the factorization. RKM is computed for 90 clusters (which corresponds to the number of classes in the collection) using as dimension of the lower-dimensional space $k = 85$, and truncated SVD is computed for $k = 85$ as well. The RKM and RKM with penalization algorithms are run 10 times (with different starting estimates), and the representation and factorization with the minimal loss function is chosen. The optimal cost parameter for LR and SVM is chosen by grid search technique from the set of values 0.1, 0.5, 1, 10, 100 and 1000. For the classification methods, the LiblineaR library in R is used, while RKM and RKM with penalization algorithm are implemented in Matlab.

4.2 Results

Results are given in terms of precision, recall, and F_1 measure of the classification. Recall is proportion of correctly classified samples among all positive samples (i.e., samples actually belonging to the class, according to the expert), while precision is proportion of correctly classified samples among all samples classified as positive by the model. In the Figures 1 and 2, are shown results of average F_1 measures of classification for 5 classes sorted in descending order by their size, i.e. number of train documents (which is 2877 to 389 for classes 1-5, 369 to 181 for classes 6-10, 140 to 111 for classes 11-15, 101 to 75 for classes 16-20, 75 to 55 for classes 21-25, 50 to 41 for classes 26-30, 40 to 37 for classes 31-35, 35 to 24 for classes 36-40, 23 to 19 for classes 41-45, 18 to 16 for classes 46-50, 16 to 13 for classes 51-55, and 13-10 for classes 56-60). Figure 1 shows the results for classification by LR, while Figure 2 for classification by SVM. Only the 60 largest classes are observed since smaller classes (less than 10 training documents) are not interesting for the

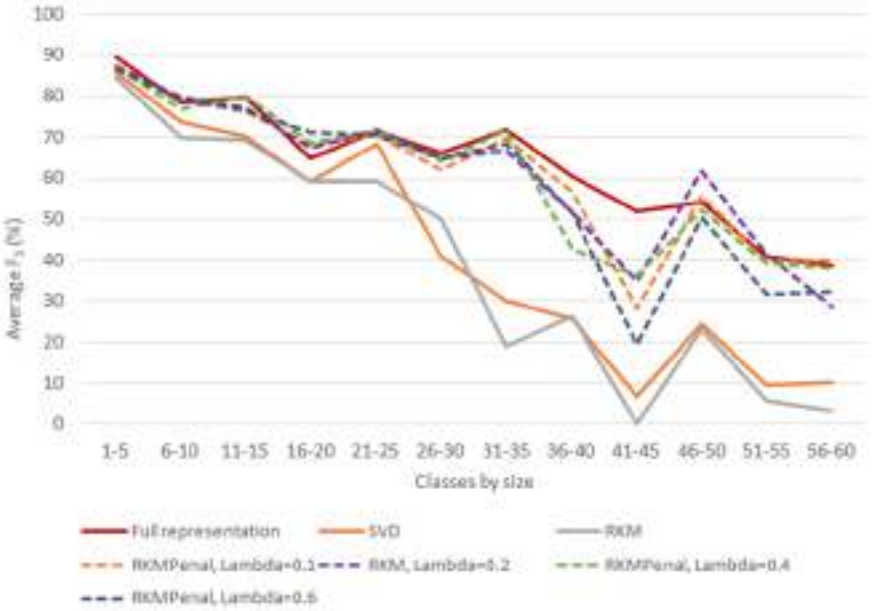


Fig. 1 Average F_1 measure of classification by LR for 5 classes sorted by their size.

research, because for those classes recall is low and it can be expected that full bag of words representation will result in better recognition since classes can possibly be recognized by key words, but not by transformed representations. It can be seen that F_1 measures are comparable for the full representation and various representations by RKM with penalization for both classification algorithms for the biggest 25 classes. For smaller classes results for representation by RKM with penalization are unstable, although for some classes they were better than the basic representation (in the case of LR). Classification for representations obtained by SVM and RKM without penalization resulted in lower F_1 measures for all class sizes.

In Table 1 are shown average precision, recall and F_1 measures for the 25 largest classes for both classification algorithms and all observed representations. In the case of classification by LR the average recall is improved for representation by RKM with penalization (for $\lambda = 0.4$) approximately 1% compared to basic full representation. For classification by SVM average precision is improved for representation by RKM with penalization (for $\lambda = 0.6$) for almost 6% and F_1 measure is improved for representation by RKM with penalization ($\lambda = 0.4$) for 2% in comparison to the basic full representation. The best results are obtained for classification by the SVM algorithm and representation with RKM with penalization with $\lambda = 0.2$ for which precision is improved for 5% with the similar level of recall as in the basic representation.

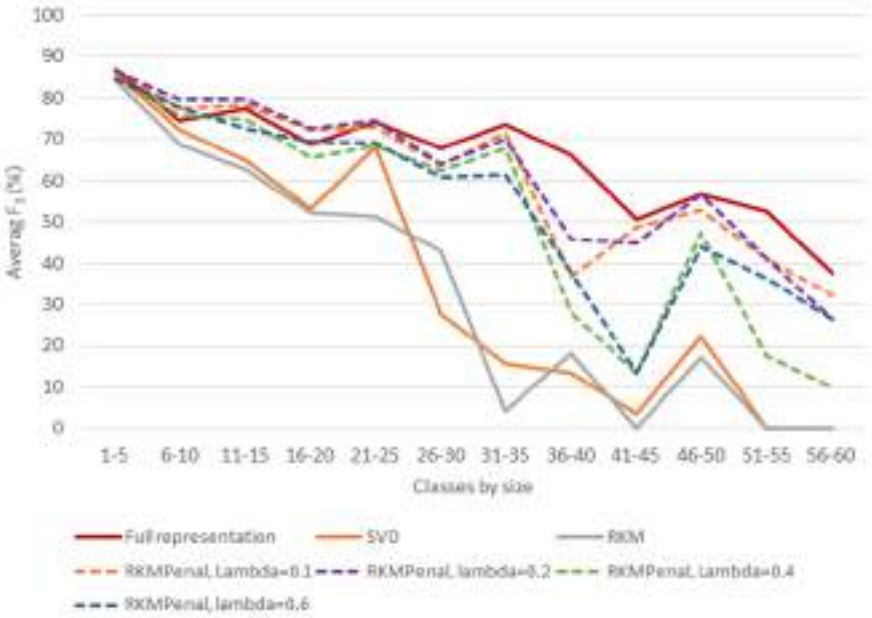


Fig. 2 Average F_1 measure of classification by SVM for 5 classes sorted by their size.

Table 1 Average precision, recall, and F_1 measure of classification for the 25 largest classes.

Class. algorithm	Logistic regression			SVM		
Representation	Precision	Recall	F_1	Precision	Recall	F_1
Full	**86.31**	70.24	76.84	82.76	71.72	76.47
SVD	82.80	64.84	71.42	85.24	61.61	68.99
RKM	80.80	61.10	68.44	82.93	55.66	63.83
RKMPenal, $\lambda = 0.1$	84.24	70.71	76.27	87.24	71.01	77.62
RKMPenal, $\lambda = 0.2$	84.68	71.23	76.72	87.78	**72.16**	**78.57**
RKMPenal, $\lambda = 0.4$	84.72	**71.38**	**76.88**	87.86	64.93	73.87
RKMPenal, $\lambda = 0.6$	85.89	70.40	76.80	**88.40**	66.11	74.75

5 Conclusions and Further Work

In this paper we propose a modification of the RKM method that simultaneously clusters documents and extracts factors on one side, and penalizes clusterings that are distant from the classification of the training documents given by experts on the other side. We show that such a modification enables representation of textual documents in a semantic lower-dimensional space that improves performance of classification. The method is tested for classes of Reuters-21758 data set and compared to the full bag of words representation and the method of LSA. It is also shown that the

original RKM method without proposed modification does not have the same effect on classification performance; it has a similar effect as the LSA method.

The proposed representation method can improve precision and recall of classification for sufficiently large classes, i.e. those that have enough training documents to enable capturing of semantic relations and characteristics of classes. A more important effect can be observed in the improvement of precision.

In the future we plan to investigate hybrid models using representation of words by neural language models and application in different domains, such as classification of images.

References

1. Bengio, J., Ducharme, R., Vincet, P., Jauvin, C.: A Neural probabilistic language model. Journal of Machine Learning Research **3**, 1137-1155 (1997)
2. Deerwester, S., Dumas, S. T., Furnas, G.W., Landauer, T. K., Harshman, R. A.: Indexing by latent semantic analysis. Journal of the American Society for Information Science **41(6)**, 381-407 (1990)
3. De Sarbo, W. S., Jedidi, K., Cool, K., Schendel, D.: Simultaneous multidimensional unfolding and cluster analysis: an investigation of strategic groups. Marketing Letters, **2**, 129-146 (1990)
4. De Soete, G., Carroll, J. D.: K-means clustering in a low-dimensional Euclidean space. In: Diday, E., Lechevallier, Y., Schader, M., Bertrand, P., Burtschy, B. (eds.) New Approaches in Classification and Data Analysis. Studies in Classification, Data Analysis, and Knowledge Organization, pp. 212-219. Springer, Heidelberg (1994)
5. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of Annual Conference of the North American Chapter of the Association for Computation Linguistic, pp. 4171-4186, Association for Computational Linguistic (2019)
6. Dobša, J., Mladenić, D., Rupnik, J., Radošević, D., Magdalenić, I.: Cross-language information retrieval by Reduced k-means, International Journal of Computer Information Systems and Industrial Management Applications, **10**, 314-322 (2018)
7. Dumas, S., Letche, T., Littman, M., Landauer, T.: Automatic cross-language retrieval using latent semantic indexing. In: Proceedings of the AAAI spring symposium on cross-language text and speech retrieval, pp. 15-21. American Association for Artificial Intelligence (1997)
8. Mikolov, T., Chen, K., Corrado, G.S., Dean, J.: Efficient estimation of word representations in vector space (2013) Available via arXiv.org `https://arxiv.org/abs/1301.3781.Cited21Jan2022`
9. Pennington, J., Socher, R., Manning, C. D.: GloVe: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532-1543, Association for Computational Linguistics, (2014)
10. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Tettlemoyer, L.: Deep contextualized word representations. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 1:2227-2237 (2018)
11. Timmerman, M. E. Ceulemans, E., Kiers, H. A. L., Vichi, M: Factorial and Reduced k-means reconsidered. Computational Statistics & Data Analyisis, **54**, 1856-1871 (2010)
12. Timmerman, M. E., Ceulemans, E., De Rover, K., Van Leeuwen, K.: Subspace$k$-means clustering, Behavioural Research, **45**, 1011-1023 (2013)
13. Vichi, M., Kiers, H. A. L.: Factorial k-means analysis for two-way data, Computational Statistics & Data Analysis, **37**, 49-64 (2001)

Trends in Data Stream Mining

João Gama

Abstract Learning from data streams is a hot topic in machine learning and data mining. This article presents our recent work on the topic of learning from data streams. We focus on emerging topics, including fraud detection and hyper-parameter tuning for streaming data. The first study is a case study on interconnected by-pass fraud. This is a real-world problem from high-speed telecommunications data that clearly illustrates the need for online data stream processing. In the second study, we present an optimization algorithm for online hyper-parameter tuning from non-stationary data streams.

Keywords: fraud detection, hyperparameter tuning, learning from data streams

1 Introduction

The developments of information and communication technologies dramatically change the data collection and processing methods. What distinguishes current data sets from earlier ones are automatic data feeds. We do not just have people entering information into a computer. We have computers entering data into each other. In most challenging applications, data are modeled best not as persistent tables, but rather as transient data streams.

This article presents our recent work on the topic of learning from data streams. It is organized into main sections. The first one is a real-world application of data stream techniques to a telecommunications fraud detection problem. It is based on the work presented in [5]. The second topic discusses the problem of hyperparameter tuning in the context of data stream mining. It is based on the work presented in [4].

João Gama (✉)
FEP-University of Porto and INESC TEC
R. Dr. Roberto Frias, Porto, Portugal, e-mail: jgama@fep.up.pt

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_15

2 Fraud Detection: a Case Study

The high asymmetry of international termination rates with regard to domestic ones, where international calls have higher charges applied by the operator where the call terminates, is fertile ground for the appearance of fraud in Telecommunications. There are several types of fraud that exploit this type of differential, being the Interconnect Bypass Fraud one of the most expressive [1, 3].

In this type of fraud, one of several intermediaries responsible for delivering the calls forwards the traffic over a low-cost IP connection, reintroducing the call in the destination network already as a local call, using VOIP Gateways. This way, the entity that sent the traffic is charged the amount corresponding to the delivery of international traffic. However, once it is illegally delivered as national traffic, it will not have to pay the international termination fee, appropriating this amount.

Traditionally, the telecom operators analyze the calls of these Gateways to detect the fraud patterns and, once identified, have their SIM cards blocked. The constant evolution in terms of technology adopted on these gateways allows them to work like real SIM farms capable of manipulating identifiers, simulating standard call patterns similar to the ones of regular users, and even being mounted on vehicles to complicate the detection using location information.

The interconnect bypass fraud detection algorithms typically consume a stream S of events, where S contains information about the origin number $A-Number$, the destination number $B-Number$, the associated timestamp, and the status of the call (accomplished or not). The expected output of this type of algorithm is a set of potential fraudulent $A-Numbers$ that require validation by the telecom operator. This process is not fully automated to avoid blocking legit $A-Numbers$ and getting penalties. In the interconnect bypass fraud, we can observe three different types of abnormal behaviors:

1. the burst of calls, which are $A-Numbers$ that produce enormous quantities of $\#calls$ (above the $\overline{\#calls}$ of all $A-Numbers$) during a specific time window W. The size of this time window is typically small;
2. the repetitions, which are the repetition of some pattern ($\#calls$) produced by a $A-Number$ during consecutive time windows W;
3. the mirror behaviors, which are two distinct $A-Numbers$ (typically these $A-Numbers$ are from the same country) that produces the same pattern of calls ($\#calls$) during a time window W.

Algorithm 2 The `Lossy Counting` Algorithm.

```
1: procedure LossyCounting(S: A Sequence of Examples; ϵ: Error margin; α: fast forgetting parameter)
2:     n ← 0; Δ ← 0; T ← 0;
3:     for example e ∈ S do
4:         n ← n + 1
5:         if e is monitored then
6:             Increment Count_e
7:         else
8:             T ← T ∪ {e, 1 + Δ}
9:         end if
10:        if ⌈n/ϵ⌉ ≠ Δ then
11:            Δ ← n/ϵ
12:        end if
13:        for all j ∈ T do
14:            if Count_j < δ then
15:                T ← T \ {j}
16:            end if
17:        end for
18:    end for
19: end procedure
```

Algorithm 3 The `Lossy Counting` with Fast Forgetting Algorithm.

```
1: procedure LossyCounting(S: A Sequence of Examples; ϵ: Error margin; α: fast forgetting parameter)
2:     n ← 0; Δ ← 0; T ← 0;
3:     for example e ∈ S do
4:         n ← n + 1
5:         if e is monitored then
6:             Increment Count_e
7:         else
8:             T ← T ∪ {e, 1 + Δ}
9:         end if
10:        if ⌈n/ϵ⌉ ≠ Δ then
11:            Δ ← n/ϵ
12:        end if
13:        for all j ∈ T do
14:            Count_j ← α * Count_j
15:            if Count_j < δ then
16:                T ← T \ {j}
17:            end if
18:        end for
19:    end for
20: end procedure
```

Figures 1 and 2 present the evolving top-10 most active phone numbers. The first Figure 1 presents the top-10 cumulative counts, while the Figure 2 presents the top-10 counts with forget.

3 Learning to Learn Hyperparameters

A hyperparameter is a parameter whose value is used to control the learning process. Hyperparameter optimization (or tuning) is the problem of choosing a set of optimal hyper-parameters for a learning algorithm. For this propose we adapt the Nelder-Mead algorithm [4] for the streaming context. This algorithm is a simplex search algorithm for multidimensional unconstrained optimization without derivatives. The vertexes of the simplex, which define a convex hull shape, are iteratively updated in order to sequentially discard the vertex associated with the largest cost function value.

The Nelder-Mead algorithm relies on four simple operations: *reflection*, *shrinkage*, *contraction* and *expansion*. Figure 3 illustrates the four corresponding Nelder-Mead operators R, S, C and E. Each vertex represents a model containing a set of hyper-parameters. The vertexes (models under optimisation) are ordered and named according to the root mean square error (RMSE) value: best (B), good (G), which is

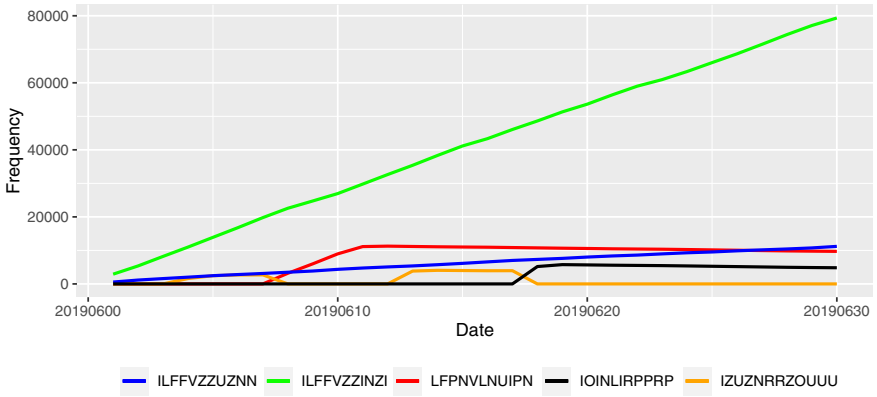


Fig. 1 Approximate Counts with Lossy Counting.

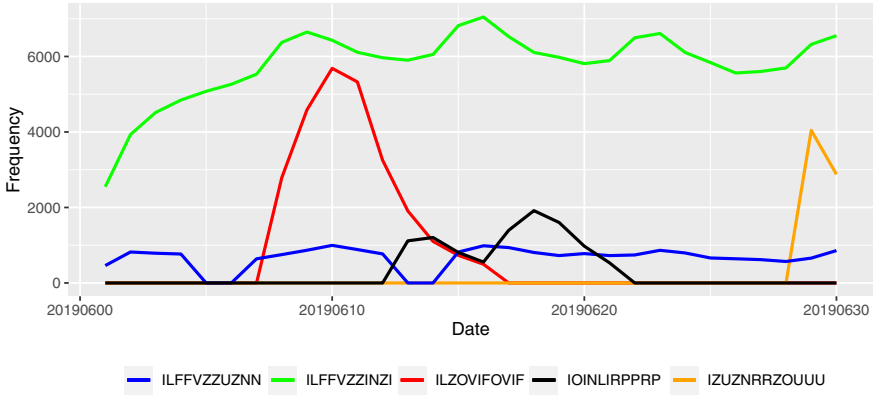


Fig. 2 Approximate Counts with Lossy Counting and Fast Forgetting.

the closest to the best vertex, and worst (W). M is a mid vertex (auxiliary model). The bottom panel in Figure 3 describe the four operations: `Contraction`, `Reflexion`, `Expansion`, and `Shrink`.

For each Nelder-Mead operation, it is necessary to compute an additional set of vertexes (midpoint M, reflection R, expansion E, contraction C and shrinkage S) and verify if the calculated vertexes belong to the search space. First, the algorithm computes the midpoint (M) of the best face of the shape as well as the reflection point (R). After this initial step, it determines whether to reflect or expand based on the set of heuristics.

The dynamic sample size, which is based on the RMSE metric, attempts to identify significant changes in the streamed data. Whenever such a change is detected, the Nelder-Mead compares the performance of the $n+1$ models under analysis to choose the most promising model. The sample size S_{size} is given by Equation 1 where σ

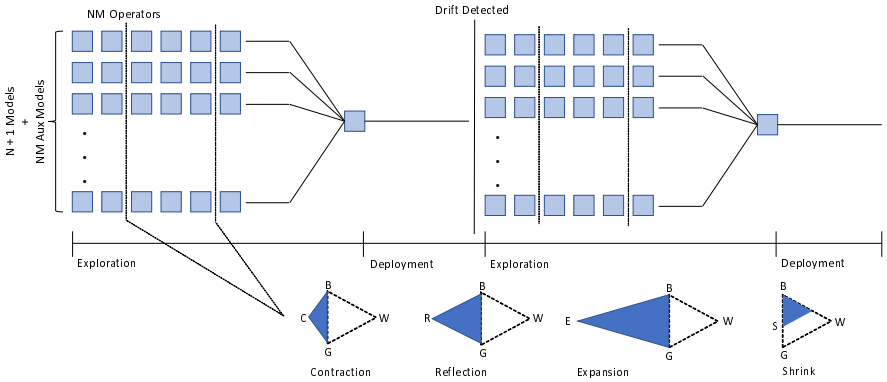


Fig. 3 SPT working modes: Exploration and Deployment. Bottom panel illustrates the Nelder & Mead operators.

represents the standard deviation of the RMSE and M the desired error margin. We use $M = 95\%$.

$$S_{size} = \frac{4\sigma^2}{M^2} \tag{1}$$

However, to avoid using small samples, that imply error estimations with large variance, we defined a lower bound of 30 samples. The adaptation of the Nelder-Mead algorithm to on-line scenarios relies extensively on parallel processing. The main thread launches the $n+1$ model threads and starts a continuous event processing loop. This loop dispatches the incoming events to the model threads and, whenever it reaches the sample size interval, assesses the running models, and calculates the new sample size. The model assessment involves the ordering of the $n + 1$ models by RMSE value and the application of the Nelder-Mead algorithm to substitute the worst model. The Nelder-Mead parallel implementation creates a dedicated thread per Nelder-Mead operator, totaling seven threads. Each Nelder-Mead operator thread generates a new model and calculates the incremental RMSE using the instances of the last sample size interval. The worst model is substituted by the Nelder-Mead operator thread model with the lowest RMSE.

Figure 4 presents the critical difference diagram [2] of three hyper-parameter tuning algorithms: SPT, Grid search, default parameter values on four benchmark classification datasets. The diagram clearly illustrates the good performance of SPT.

4 Conclusions

This paper reviews our recent work in learning from data streams. The two works present different approaches to dealing with high-speed and time-evolving data: from applied research in fraud detection to fundamental research on hyperparameter

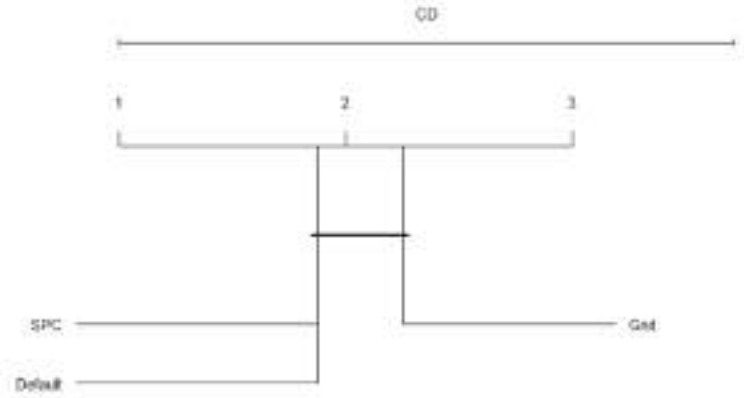


Fig. 4 Critical Difference Diagram comparing Self hyperparameter tuning, Grid hyperparameter tuning, and default parameters in 4 classification problems.

optimization for streaming algorithms. The first work identifies burst on the activity in phone calls, using approximate counting with forgetting. The last work presents a streaming optimization method to find the minimum of a function and its application in finding the hyper-parameter values that minimize the error. We believe that the two works reported here will have an impact on the work of other researchers.

Acknowledgements I would like to thank my collaborators Bruno Veloso and Rita P. Ribeiro that contribute to this work.

References

1. Ali, M. A., Azad, M. A., Centeno, M. P., Hao, F., van Moorsel, A.: Consumer-facing technology fraud: Economics, attack methods and potential solutions. Future Generation Computer Systems, **100**, 408–427 (2019)
2. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research, **7**(Jan), 1–30 (2006)
3. Laleh, N., Azgomi, M. A.: A taxonomy of frauds and fraud detection techniques. In International Conference on Information Systems, Technology and Management, pp. 256–267. Springer (2009)
4. Veloso, B., Gama, Malheiro, J. B., Vinagre, J.: Hyperparameter self-tuning for data streams. Information Fusion, **76**, 75–86 (2021)
5. Veloso, B., Tabassum, S., Martins, C., Espanha, R., Azevedo, R., Gama, J.: Interconnect bypass fraud detection: a case study. Annals des Télécommunications, **75**(9), 583–596 (2020)

Old and New Constraints in Model Based Clustering

Luis A. García-Escudero, Agustín Mayo-Iscar, Gianluca Morelli, and Marco Riani

Abstract Model-based approaches to cluster analysis and mixture modeling often involve maximizing classification and mixture likelihoods. Without appropriate constrains on the scatter matrices of the components, these maximizations result in ill-posed problems. Moreover, without constrains, non-interesting or "spurious" clusters are often detected by the EM and CEM algorithms traditionally used for the maximization of the likelihood criteria. A useful approach to avoid spurious solutions is to restrict relative components scatter by a prespecified tuning constant. Recently new methodologies for constrained parsimonious model-based clustering have been introduced which include the 14 parsimonious models that are often applied in model-based clustering when assuming normal components as limit cases. In this paper we initially review the traditional approaches and illustrate through an example the benefits of the adoption of the new constraints.

Keywords: model based clustering, mixture modelling, constraints

L. A. García-Escudero
Department of Statistics and Operational Research and IMUVA, University of Valladolid, Spain, e-mail: lagarcia@eio.uva.es

A. Mayo-Iscar
Department of Statistics and Operational Research and IMUVA, University of Valladolid, Spain, e-mail: agustinm@eio.uva.es

G. Morelli
Department of Economics and Management and Interdepartmental Centre of Robust Statistics, University of Parma, Italy, e-mail: gianluca.morelli@unipr.it

M. Riani (✉)
Department of Economics and Management and Interdepartmental Centre of Robust Statistics, University of Parma, Italy, e-mail: mriani@unipr.it

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_16

1 Introduction

Given a sample of observations $\{x_1, ..., x_n\}$ in $\mathbb{R}^p$, a widely used method in unsupervised learning is to assume multivariate normal components and to adopt a maximum likelihood approach for clustering purposes. With this idea in mind, well-known classification and mixture likelihood approaches can be followed.

In this work, we use $\phi(\cdot; \mu, \Sigma)$ to denote the probability density function of a p-variate normal distribution with mean μ and covariance matrix Σ.

In the *classification likelihood* approach we search for a partition $\{H_1, ..., H_k\}$ of the indices $\{1, \cdots, n\}$, centres $\mu_1, \cdots, \mu_k$ in $\mathbb{R}^p$, symmetric positive semidefinite $p \times p$ scatter matrices $\Sigma_1, \cdots, \Sigma_k$ and positive weights $\pi_1, \cdots, \pi_k$ with $\sum_{j=1}^k \pi_j = 1$, which maximize

$$\sum_{j=1}^{k} \sum_{i \in H_j} \log \left(\pi_j \phi(x_i; \mu_j, \Sigma_j) \right). \tag{1}$$

On the other hand, in the *mixture likelihood* approach, we seek the maximization of

$$\sum_{i=1}^{n} \log \left(\sum_{j=1}^{k} \pi_j \phi(x_i; \mu_j, \Sigma_j) \right), \tag{2}$$

with similar notation and conditions on the parameters as above. In this second approach, a partition into k groups can be also obtained, from the fitted mixture model, by assigning each observation to the cluster-component with the highest posterior probability.

Unfortunately, it is well-known that the maximization of "log-likelihoods" like (1) and (2) without constraints on the Σ_j matrices is a mathematically ill-posed problem [1, 2]. To see this unboundedness issue, we can just take $\mu_1 = x_1$, $\pi_1 > 0$ and $|\Sigma_1| \to 0$ making (2) to diverge to infinity or (1) also to diverge with $H_1 = \{1\}$.

This lack of boundedness can be solved by just focusing on local maxima of the likelihood target functions. However, many local maxima are often found and it is difficult to know which are the most interesting ones. See [3] for a detailed discussion of this issue. In fact, non-interesting local maxima denoted as "spurious" solutions, which consist of a few, almost collinear, observations, are often detected by the Classification EM algorithm (CEM), traditionally applied when maximizing (1), and by the EM algorithm, traditionally applied when maximizing (2). A recent review of approaches for dealing with this lack of boundedness and for reducing the detection of spurious solutions can be found in [4].

It is also common to enforce constraints on the Σ_j scatter matrices when maximizing (1) or (2). Among them, the use of "parsimonious" models [5, 6] is one of the most popular and widely applied approaches in practice. These parsimonious models follow from a decomposition of the Σ_j scatter matrices as

$$\Sigma_j = \lambda_j \Omega_j \Gamma_j \Omega_j', \tag{3}$$

with $\lambda_j = |\Sigma_j|^{1/p}$ (volume parameters),

$$\Gamma_j = \mathsf{diag}(\gamma_{j1}, ..., \gamma_{jl}, ..., \gamma_{jp}) \text{ with } \det(\Gamma_j) = \prod_{l=1}^{p} \gamma_{jl} = 1$$

(shape matrices), and Ω_j (rotation matrices) with $\Omega_j \Omega_j' = \mathsf{I}_p$. Different constraints on the λ_j, Ω_j and Γ_j elements are considered across components to get 14 parsimonious models (which are coded with a combination of three letters). These models reduce notably the number of free parameters to be estimated, so improving efficiency and model interpretability. Moreover, many of them turn the constrained maximization of the likelihoods into well-defined problems and help to avoid spurious solutions. Unfortunately, the problems remain for models with unconstrained λ_j volume parameters, which are coded with the first letter as a `V` (`V**` models). Aside from relying on good initializations, it is common to consider the early stopping of iterations when approaching scatter matrices with very small eigenvalues or when detecting components accounting for a reduced number of observations. A not fully iterated solution (or no solution at all) is then returned in these cases. The idea is known, for instance, to be problematic when dealing with (well-separated) components made up of a few observations.

Starting from a seminal paper by [7], an alternative approach is to constrain the Σ_j scatter matrices by specifying some tuning constants that control the strength of the constraints. In this direction, the ratio between the largest and the smallest of the $k \times p$ eigenvalues of the Σ_j matrices was forced to be smaller than a given fixed constant $c^* \geq 1$ [8, 9, 10, 11, 12]. This means that the maximization of (1) and (2) is done under the (more simple) constraint:

$$\max_{jl} \lambda_l(\Sigma_j) / \min_{jl} \lambda_l(\Sigma_j) \leq c^*, \tag{4}$$

where $\{\lambda_l(\Sigma_j)\}_{l=1}^{p}$ are the set of eigenvalues of the Σ_j matrix, $j = 1, ..., k$.

With this eigenvalue-ratio approach, we need a very high c^* value to be close to affine equivariance. Unfortunately, such a high c^* value does not always successfully prevent us from incurring into spurious solutions.

2 The New Constraints

García-Escudero *et al.* [13] have recently introduced three different types of constraints on the Σ_j matrices which depend on three constants c_{det}, c_{shw} and c_{shb} all of them being greater than or equal to 1.

The first type of constraint serves to control the maximal ratio among determinants and, consequently, the maximum allowed difference between component volumes:

$$\text{“deter”:} \qquad \frac{\max_{j=1,...,k} |\Sigma_j|}{\min_{j=1,...,k} |\Sigma_j|} = \frac{\max_{j=1,...,k} \lambda_j^p}{\min_{j=1,...,k} \lambda_j^p} \leq c_{\mathsf{det}}. \tag{5}$$

The second type of constraint controls departures from sphericity "within" each component:

$$\text{shape-"within":} \qquad \frac{\max_{l=1,...,p} \gamma_{jl}}{\min_{l=1,...,p} \gamma_{jl}} \leq c_{\text{shw}} \text{ for } j = 1, ..., k. \tag{6}$$

This provides a set of k constraints that in the most constrained case, $c_{\text{shw}} = 1$, imposes $\Gamma_1 = ... = \Gamma_p = \mathsf{I}_p$, where I_p is the identity matrix of size p, i.e., sphericity of components.

Note that the new determinant-and-shape constraints (based on $c_{\text{det}} > 1$ and $c_{\text{shw}} = 1$) in (4) allow us to deal with spherical "heteroscedastic" cases, whereas the eigenvalue ratio constraint with $c^* = 1$ can only handle the spherical "homoscedastic" case. Constraints (5) and (6) were the basis for the "deter-and-shape" constraints in [14]. These two constraints alone resulted in mathematically well-defined constrained maximizations of the likelihoods in (1) and (2). However, although highly operative in many cases, they do not include, as limit cases, all the already mentioned 14 parsimonious models. For instance, we may be interested in the same (or not very different) Γ_j or Σ_j matrices for all the mixture components and these cannot be obtained as limit cases from the "deter-and-shape" constraints.

The third constraint serves to control the maximum allowed difference between shape elements "between" components:

$$\text{shape-"between":} \qquad \frac{\max_{j=1,...,k} \gamma_{jl}}{\min_{j=1,...,k} \gamma_{jl}} \leq c_{\text{shb}} \text{ for } l = 1, ..., p. \tag{7}$$

This new type of constraint allows us to impose "similar" shape matrices for the components and, consequently, enforce $\Gamma_1 = ... = \Gamma_k$ in the most constrained $c_{\text{shb}} = 1$ case.

3 An Illustration Example of the New Constraints

Figure 1 shows an example based on three groups. The data have been generated imposing equal determinants $c_{\text{det}} = 1$, a sensible departure from sphericity "within" each component $c_{\text{shw}} = 40$ and a very moderate difference "between" shape elements components, $c_{\text{shb}} = 1.3$. No constraint has been imposed on the rotation matrices. Finally an average overlap of 0.10 has been imposed. The generation of these data sets has been done through the `MixSim` method of [15], as extended by [16] and incorporated into the `FSDA` Matlab toolbox [17]. The overlap is defined as a sum of pairwise misclassification probabilities. See more details in [16].

The application of traditional tclust approach with maximum ratio between eigenvalues (c^*) respectively equal to 128 and 10^{10} produces the classifications shown in the left panels of Figure 2. In fact, it could be seen that the results in the top left panel would be exactly the same one for any choice of c^* within the interval $[16, 128]$. This means that a higher value of c^* would be apparently needed to detect

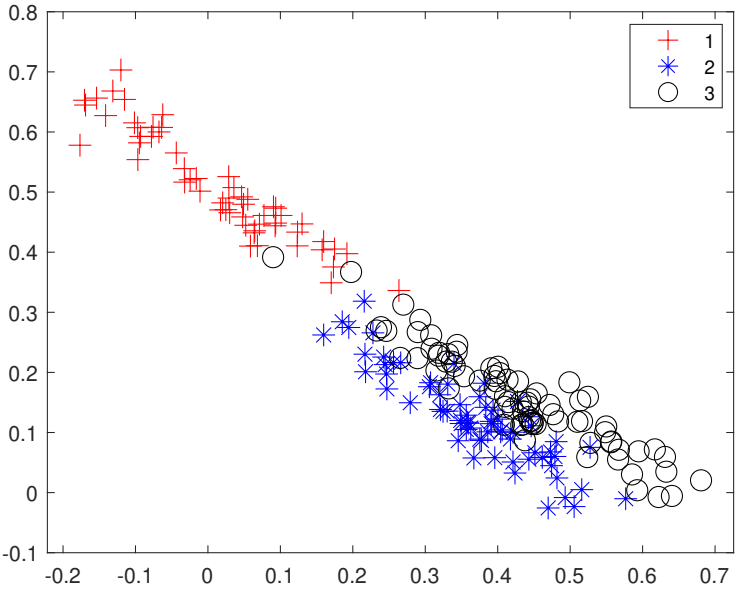


Fig. 1 An example with simulated data with 3 clusters in two dimensions. The average overlap is 0.10. The data have been generated using equal determinants, moderate difference between shape elements "between" components and sensible departure from sphericity "within" each component.

those two almost parallel clusters that were shown in Figure 1. However, choosing a value greater for c^* may destroy the desired protection against spurious solutions provided by the constraints. For example, we see in the lower left panel how the choice $c^* = 10^{10}$ results in the detection of a spurious group consisting of a single observation.

The panels on the right, on the other hand, show the partitions resulting from the 3 new constraints imposed on the components covariance matrices. The top right panel shows the result of applying the 3 new restrictions with values of the tuning constants very close to the real values used to generate the dataset. We can see that, in this case, it is possible to recover the real structure of the data generating process. Moreover, the real cluster structure is also recovered in the low right panel by choosing larger values of these tuning constants, but not too large just to avoid detection of spurious solutions. Some guidelines about how to choose these tuning constants can be found in [13].

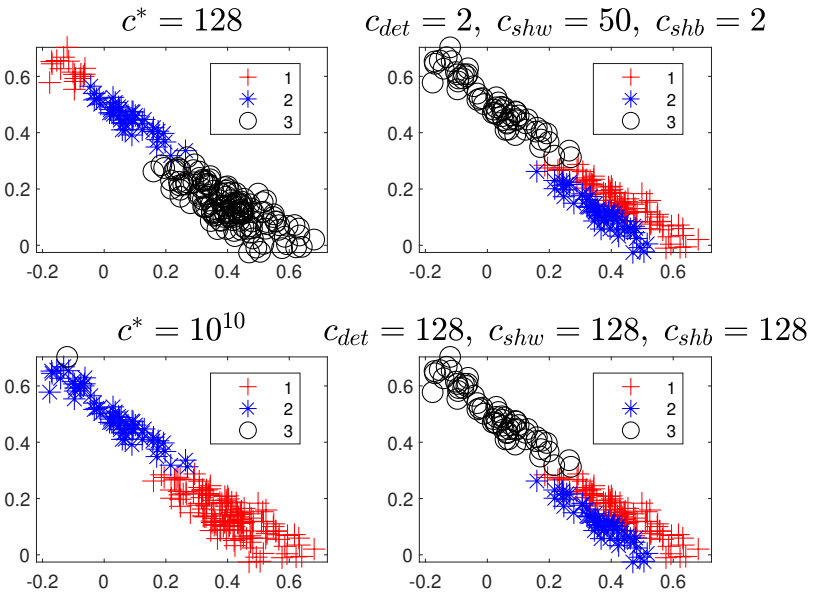


Fig. 2 Comparison between the traditional (left panels) and new tclust procedure (right panels).

References

1. Kiefer, J., Wolfowitz, J.: Consistency of the maximum likelihood estimator in the presence of infinitely many incidental parameters. Ann. Math. Stat. **27**, 887-906 (1956)
2. Day, N. E.: Estimating the components of a mixture of normal distributions. Biometrika, **56**, 463-474 (1969)
3. McLachlan, G., Peel, D. A.: Finite Mixture Models. Wiley Series in Probability and Statistics, New York (2000)
4. García-Escudero, L. A., Gordaliza, A., Greselin, F., Ingrassia, S., Mayo-Iscar, A.: Eigenvalues and constraints in mixture modeling: geometric and computational issues. Adv. Data Anal. Classif. **12**, 203-233 (2018)
5. Celeux, G., Govaert, G.: Gaussian parsimonious clustering models. Pattern Recogn. **28**, 781-793 (1995)
6. Banfield, J. D., Raftery, A. E.: Model-based Gaussian and non-Gaussian clustering. Biometrics **49**, 803-821 (1993)
7. Hathaway, R. J.: A constrained formulation of maximum likelihood estimation for normal mixture distributions. Ann. Stat. **13**, 795-800 (1985)
8. Ingrassia, S., Rocci, R.: Constrained monotone EM algorithms for finite mixture of multivariate Gaussians. Comput. Stat. Data Anal. **51**, 5339-5351 (2007)
9. García-Escudero, L. A., Gordaliza, A., Matrán, C., Mayo-Iscar, A.: A general trimming approach to robust cluster analysis. Ann. Stat. **36**, 1324-1345 (2008)
10. García-Escudero, L. A., Gordaliza, A., Matrán, C., Mayo-Iscar, A.: Exploring the number of groups in robust model-based clustering. Stat. Comput. **21**, 585-599 (2011)
11. García-Escudero, L. A., Gordaliza, A., Mayo-Iscar, A.: A constrained robust proposal for mixture modeling avoiding spurious solutions. Adv. Data Anal. Classif. **8**, 27-43 (2014)

12. García-Escudero, L. A., Gordaliza, A., Matrán, C., Mayo-Iscar, A.: Avoiding spurious local maximizers in mixture modeling. Stat. Comput. **25**, 619-633 (2015)
13. García-Escudero, L. A., Mayo-Iscar, A., Riani, M.: Constrained parsimonious model-based clustering. Stat. Comput. **32** (2022)
14. García-Escudero, L. A., Mayo-Iscar, A., Riani, M.: Model-based clustering with determinant-and-shape constraint. Stat. Comput. **25**, 1-18 (2020)
15. Maitra, R., Melnykov, V.: Simulating data to study performance of finite mixture modeling and clustering algorithms. J. Comput. Graph. Stat. **19**, 354-376 (2010)
16. Riani, M., Cerioli, A., Perrotta, D., Torti, F.: Simulating mixtures of multivariate data with fixed cluster overlap in FSDA library. Adv. Data Anal. Classif. **9**, 461-481 (2015)
17. Riani, M., Perrotta, D., Torti, F.: FSDA: a Matlab toolbox for robust analysis and interactive data exploration. Chemometr. Intell. Lab. Syst. **116**, 17-32 (2012)

Clustering Student Mobility Data in 3-way Networks

Vincenzo Giuseppe Genova, Giuseppe Giordano, Giancarlo Ragozini, and Maria Prosperina Vitale

Abstract The present contribution aims at introducing a network data reduction method for the analysis of 3-way networks in which classes of nodes of different types are linked. The proposed approach enables simplifying a 3-way network into a weighted two-mode network by considering the statistical concept of joint dependence in a multiway contingency table. Starting from a real application on student mobility data in Italian universities, a 3-way network is defined, where provinces of residence, universities and educational programmes are considered as the three sets of nodes, and occurrences of student exchanges represent the set of links between them. The Infomap community detection algorithm is then chosen for partitioning two-mode networks of students' cohorts to discover different network patterns.

Keywords: 3-way network, complex network, community detection, mobility data, tertiary education

Vincenzo Giuseppe Genova
Department of Economics, Business, and Statistics, University of Palermo, Italy,
e-mail: vincenzogiuseppe.genova@unipa.it

Giuseppe Giordano
Department of Political and Social Studies, University of Salerno, Italy,
e-mail: ggiordano@unisa.it

Giancarlo Ragozini
Department of Political Science, Federico II University of Naples, Italy,
e-mail: giragoz@unina.it

Maria Prosperina Vitale (✉)
Department of Political and Social Studies, University of Salerno, Italy,
e-mail: mvitale@unisa.it

P. Brito et al. (eds.), *Classification and Data Science in the Digital Age*,
Studies in Classification, Data Analysis, and Knowledge Organization,
https://doi.org/10.1007/978-3-031-09034-9_17

1 Introduction

Many complex relational data structures can be described as multimode or multiway networks in which nodes belonging to different modes are linked. The most common multimode network in social networks is represented by the affiliation network, where two-mode data, actors and events, form a bipartite graph divided into two groups [6]. In the case of tripartite networks, we deal with three types of nodes, and different graph structures can be defined.

Although only a few papers deal with methods for these networks, in recent years, a growing number of works have appeared –especially in bipartite and tripartite cases– to disentangle the inherent complexity of such kinds of data structures. Looking at clustering and community detection algorithms proposed to partition a network into groups, we can identify some strands, all deriving from generalizations of methods suited for one-mode [19] and two-mode networks [2]. A classical approach consists of applying the usual community detection algorithms on a unique supra-adjacency matrix defined by combining all the possible two-mode networks in a block matrix [11, 15]. Alternative methods rely on projecting each two-mode networks and on applying separately the usual community detection algorithms on these matrices [10]. In addition, there are methods adopting both an optimization procedure for 3-way networks [16, 17, 14] by extending the idea of bipartite modularity [2], and an indirect blockmodeling approach by deriving a dissimilarity measure based on structural equivalence concept [3].

In our opinion, approaches based on the analysis of the k-modes examined considering the collection of the $k(k-1)/2$ two-mode networks [10] cannot take into account statistical associations among all modes at same time. Hence, the aim of the contribution is to present a network data reduction method based on the concept of joint dependence in a multiway contingency table [1].

Starting from real applications on the Italian student mobility phenomenon in higher education [12, 21, 7, 8, 13, 22], a 3-way network is defined, where provinces of residence, universities and educational programmes are considered as the three modes. Student mobility flows, measured in terms of occurrences, represent the set of links between them. Assuming that the statistical dependency between the set of nodes provinces of residence and the other two sets of nodes can be captured by the joined pair of nodes (universities and educational programmes), the tripartite network is transformed into a bipartite network, where the two modes are given by Italian provinces of residence (first mode) and the set of nodes given by all possible pairs of universities and educational programmes (second mode). Thus, taking advantage of this approach of network simplification, network indexes and clustering techniques for bipartite networks are available. Hence, the Infomap community detection algorithm is adopted [9, 4] to partition the derived network.

The remainder of the paper is organized as follows. Section 2 presents the details of the proposed strategy of analysis, and the main results are reported from the analysis of student mobility data of Italian universities. Section 3 provides final remarks.