Università degli Studi di Palermo

AREA QUALITÀ, PROGRAMMAZIONE E SUPPORTO STRATEGICO
SETTORE STRATEGIA PER LA RICERCA
U. O. DOTTORATI

Dottorato in Scienze Economiche e Statistiche
Dipartimento di Scienze Economiche, Aziendali e Statistiche
SECS-S/01 - Statistica

Distance-based and ranking methods for preference rankings, preference-approvals and textual analysis

IL DOTTORE
Alessandro Albano

IL COORDINATORE
Andrea Consiglio

IL TUTOR
Antonella Plaia

CO TUTOR
Mariangela Sciandra

CICLO XXXV
ANNO CONSEGUIMENTO TITOLO 2022

Abstract

This thesis offers an original contribution to knowledge on distance-based methods for preference rankings and preference-approvals and the definition of a new ranking method for textual analysis. The essay starts with a short review of rankings, focusing on leading distance and correlation measures along with ranking aggregation and prediction tasks. Then, we present the definition of a new element weighted rank correlation coefficient and its corresponding element weighted ranking distance for linear, weak, and incomplete orderings. The proposed measures, encoding the individual importance of alternatives and their similarity structure, allow us to build two algorithms to perform weighted aggregation and prediction of rankings.

The focus then shifts to preference-approvals, an extension of the traditional preference model obtained by combining preference rankings and approval voting. In this framework, one of the most pressing issues is to develop clustering methods to tackle the complexity of the preference-approval space. To this aim, a new preference-approval metric is first proposed to identify optimal clusters of votes. The alternatives are then clustered using a new pseudometric to reduce the complexity of the preference-approval space.

The last chapter presents a new ranking method for topic modelling, one of the most famous machine learning models for textual analysis. The ranking method proposed is based on a new topic-coherence measure employing Statistically Validated Networks. To prove the effectiveness of our ranking method, we collect the judgements of PhD students from the University of Palermo, Italy, and construct a benchmark dataset. These judgments are taken as ground truth, showing that the proposed measure reproduces human judgment rankings more precisely than the state-of-the-art measures.

The right understanding of any matter,
and a misunderstanding of the same matter,
do not wholly exclude each other.
- Franz Kafka

Acknowledgements

The doctorate has been a time of tremendous growth for me. I consider myself privileged to have met so many great and valuable people on my path.
First and foremost, I would like to convey my heartfelt gratitude to Prof. Antonella Plaia and Prof. Mariangela Sciandra, who have been invaluable guides throughout my journey. In addition to having outstanding scientific backgrounds, strong statistical expertise, and a passion for teaching, these two academics also have exceptional attitudes. Professor Plaia proved to be an excellent supervisor; I immediately appreciated her kindness, fairness, pragmatism, and the order and cleanliness of her mind. Professor Sciandra guided me with her unwavering enthusiasm. Her inventiveness, active attitude, and thousand ideas were crucial to my development.
I wish to express my gratitude to Prof. José Luis García-Lapresta of the University of Valladolid, where I was a visiting student for three months in a wonderful city. Our collaboration was very fruitful, and I found a prepared, caring, helpful and affectionate professor in him. I always felt at home during the three months there, which allowed me to give my best.
Mention must be made to the department's statistics professors who helped me build the foundation of my scientific knowledge. In particular, Professor Massimo Attanasio, who was the supervisor of my bachelor's and master's thesis, from whom I learnt so much and who encouraged me to undertake a doctorate.
I want to thank my PhD colleagues with whom I've enjoyed many laughs and inside jokes (do you feel more like X or Y?). Andrea P., Nicoletta, Salvo, Furio, and Andrea S., in particular, were the XXXV cycle colleagues with whom I shared most of my academic life and to whom I am most attached (all-you-can-eat ribs?).
I am grateful to my parents and sister for filling me with their love; you are and will always be a shield against life's hardships. I want to convey my deepest gratitude to my girlfriend, Francesca, for being my happy island, always being by my side, and for her unconditional love and support.

Finally, I thank my best friends Gabriele, Gianluca, Andrea and Federico; I would not be the person I am without you.

Contents

1 Introduction **1**

1.1 Outline of the thesis 2

2 Preliminaries on rankings **6**

2.1 Notation 6

2.2 Distances and correlation for rankings 7

2.2.1 Kendall's correlation coefficient τ_b 8

2.2.2 Emond and Mason's correlation coefficient τ_x 9

2.2.3 Spearman's distance d_s 10

2.2.4 Kemeny distance d_K 10

2.3 Aggregation of rankings: the consensus problem 12

2.4 Prediction of rankings: the Label Ranking task 13

3 Element weighted Kemeny distance for ranking data **17**

3.1 Introduction 17

3.2 Item weighted distances and correlation coefficient 18

3.2.1 Introducing element weights in the Kemeny distance 19

3.2.2 A new weighted rank correlation coefficient 22

3.2.3 The case of 0-weight items 24

3.2.4 A variant of element weights: the element similarities 25

3.3 The choice of weights 28

3.3.1 Frequency-based weights 28

3.4 Reaching the weigthed consensus ranking 30

3.5 Experimental evaluation 32

3.5.1 Simulation under model I 32

3.5.2 Simulation under model II 36

3.5.3 A real data application: the ISTAT dataset 38

3.5.4 A real data application: the Quiz dataset . . . 40
3.6 Concluding remarks . . . 41

4 A weighted distance-based approach with boosted decision trees for Label Ranking 42
4.1 Introduction . . . 42
4.2 Decision trees and boosting methods . . . 44
4.3 Building an item-weighted tree ensemble for label ranking . . . 45
4.3.1 Weighted distance-based trees: splitting and labelling criteria . . . 46
4.3.2 Item-weighted boosting algorithm . . . 48
4.4 Experimental evaluation . . . 55
4.4.1 Simulation study . . . 56
4.4.2 Real Data applications . . . 65
4.5 Concluding remarks . . . 70

5 A family of distances for preference-approvals 71
5.1 Introduction . . . 71
5.2 Preference-approval . . . 73
5.2.1 Codifications . . . 75
5.3 The proposal . . . 76
5.4 Clustering tasks . . . 83
5.4.1 Universe of preference-approvals . . . 83
5.4.2 A real data application . . . 90
5.5 Concluding remarks . . . 97

6 A new pseudometric for clustering alternatives in preference-approvals 100
6.1 Introduction . . . 100
6.1.1 A pseudometric on preferences . . . 102
6.1.2 A pseudometric on approvals . . . 102
6.2 The proposal . . . 103
6.2.1 Preference discordances . . . 103
6.2.2 Approval discordances . . . 105
6.2.3 Global discordances . . . 106
6.2.4 Clustering procedure and visualization . . . 108
6.3 Case studies . . . 109
6.3.1 Eurobarometer dataset . . . 109
6.3.2 Pew Research Center dataset . . . 116
6.4 Concluding remarks . . . 121

7 Ranking coherence in Topic Models using Statistically Validated Networks 122
7.1 Introduction . . . 122
7.2 Background and related works . . . 124
7.2.1 Literature review . . . 125
7.2.2 Qualitative methods . . . 126
7.2.3 Quantitative methods . . . 127
7.3 Methods . . . 131
7.3.1 Statistically Validated Networks . . . 132
7.3.2 Coherence based on SVNs . . . 134
7.4 Experimental evaluation . . . 138
7.4.1 Dataset and pre-processing . . . 138
7.4.2 Coherence-based topic annotations . . . 140
7.4.3 Data analysis and results . . . 143
7.4.4 Interpretation of the resulting topics . . . 145
7.4.5 Summary of main findings . . . 148
7.5 Concluding remarks . . . 148

8 Conclusions 150

A Additional material Chapter 2 169
A.1 A comparison of the weighted and unweighted QuickCons algorithms' computation times . . . 169

B Additional material Chapter 3 170
B.1 Variable importance in the boosting procedure, datasets: German2005, German2009 and Top7Movies . . . 170

C Proofs of formulas in Chapter 4 172
C.1 Proof of Proposition 1 . . . 172
C.2 Proof of Proposition 2 . . . 175

D Additional material Chapter 6 177

List of Figures

2.1 Permutation polytope for full and partial rankings of four objects (Heiser and D'Ambrosio, 2013). 12

3.1 Distribution of $\tau_{x,e}$ vs weighting vectors 35
3.2 Distribution of $\tau_{x,e}$ vs weighting vectors 37

4.1 Exponential multiplier M_1 and binary logistic multiplier M_2, vs the ranking correlation coefficient $\tau_{x,e}$, with $e_b = 0.1$. 52
4.2 Theoretical partition of the predictor space (X_1, X_2) with 4 items . . . 57
4.3 AdaBoost.R.M1, AdaBoost.R.M2, AdaBoost.R.M3 and BoostLR (Dery and Shmueli, 2020) for all the simulated scenarios with 50 trees. Different levels of homogeneity among the rankings, $\theta = (0.4, 0.7, 2)$, two sample sizes, $n = (200, 500)$ and weights $\mathbf{w}_1 = (1, 1, 1, 1)$, Model I. . 60
4.4 AdaBoost.R.M1, AdaBoost.R.M2 and AdaBoost.R.M3 for all the simulated scenarios with 50 trees. Different levels of homogeneity among the rankings, $\theta = (0.4, 0.7, 2)$, two sample sizes, $n = (200, 500)$ and weights $\mathbf{w}_2 = (5, 2, 2, 5)$, Model I. 61
4.5 AdaBoost.R.M1, AdaBoost.R.M2 and AdaBoost.R.M3 for all the simulated scenarios with 50 trees. Different levels of homogeneity among the rankings, $\theta = (0.4, 0.7, 2)$, two sample sizes, $n = (200, 500)$ and weights $\mathbf{w}_3 = (10, 1, 1, 10)$, Model I. 62
4.6 AdaBoost.R.M1, AdaBoost.R.M2 and AdaBoost.R.M3 for all the simulated scenarios with 50 trees. Different levels of homogeneity among the rankings, $\theta = (0.4, 0.7, 2)$, two sample sizes, $n = (200, 500)$ and a penalization matrix $\mathbf{P}$, Model II. 64
4.7 Parties similarity 65
4.8 Item-weighted boosting applied to German Elections dataset 2005: $err(b)$. 67

4.9 Item-weighted boosting applied to German Elections dataset 2009: $err(b)$. 67
4.10 Item-weighted boosting applied to Top7movies dataset: $err(b)$. 69

5.1 Preference-approval plane. 80
5.2 Distances between preference-approvals for 2 alternatives, $r = 1$ and $\lambda = 0.5$. 84
5.3 Distances between preference-approvals for 2 alternatives, $r = 1$ and $\lambda = 0.75$. 85
5.4 Hierarchical clustering dendrogram for 2 alternatives, $r = 1$, $\lambda = 0.5$ (left) and $\lambda = 0.75$ (right). 85
5.5 Distance between preference-approvals for 2 alternatives, $r = 2$, $\lambda = 0.5$ 86
5.6 Hierarchical clustering dendrogram for 2 alternatives, $r = 1$ (left), $r = 2$ (right) and $\lambda = 0.5$. 87
5.7 Average cluster-wise stability over r. 94
5.8 EU cluster dendrogram. 95
5.9 Map of EU voters with clusters. 95

6.1 Individual preference-discordance of two adjacent alternatives by m. . 104
6.2 Heatmaps δ_λ. 107
6.3 Preference-approval plane, Eurobarometer. 112
6.4 Graphical representation of RKM clusters. 114
6.5 Preference-approval plane, Pew Research Center. 118
6.6 Graphical representation of RKM clusters. 120

7.1 Bipartite network where S is the set of corpus sentences and W is the set of topic words. 132
7.2 Venn Diagram showing the overlap of two words 133
7.3 Diagram describing the 5 steps of the algorithm. 134
7.4 Statistically Validated Network of an artificial topic. 135
7.5 Annotators' coherence evaluations 143
7.6 SVN representation of Topic z_2 and Topic z_6 146
7.7 SVN representation of Topic z_3 and Topic z_{28} 147

A.1 Computation times comparison: item-weighted QuickCons vs unweighted QuickCons. 169

B.1 Variable Importance at final step for 2005-2009 German Elections dataset 170

List of Tables

3.1 Weighting vector and data matrix 20
3.2 Weighted Kemeny distances 20
3.3 Relative weights ${}_a r_{ij}$ of each inversion with arithmetic average 21
3.4 Relative weights ${}_p r_{ij}$ of each inversion with product 21
3.5 Weighting vector and original data matrix 25
3.6 Weighting vector and modified data matrix 25
3.7 Data matrix 26
3.8 Penalization matrix 27
3.9 Relative weights r_{ij} of each inversion with item similarities 27
3.10 Unweighted and weighted Kemeny distances 27
3.11 Ordering data matrix 29
3.12 Weighting vector 29
3.13 Item weighted Kemeny distances $d_{K,e}$ 29
3.14 Relative weights r_{ij} of each generic inversion when using w_1 33
3.15 Relative weights r_{ij} of each generic inversion when using w_2 33
3.16 Relative weights r_{ij} of each generic inversion when using w_3 33
3.17 Distribution of consensus ranking vs weighting vector 34
3.18 Distribution of consensus ranking vs weighting vector 36
3.19 Relative weight of each inversion 39
3.20 Consensus ranking for each weighting vectors 39
3.21 Relative weights r_{ij} of each generic inversion 40

4.1 Tagging strategy applied to the universe of permutations (with ties) of 4 items 46
4.2 Predictor matrix structure 51
4.3 Model weights and relative model weights of a demonstrative example. 54
4.4 Simulated data penalization matrix $\mathbf{P}$. 63

4.5 Characteristics of the real-world datasets. 65
4.6 Political parties penalization matrix **P**. 66
4.7 Top7 movies penalization matrix **P**. 69

5.1 Number of approvals, linear orders, weak orders and preference-approvals. 74
5.2 Quotients between preference-approvals and approvals. 75
5.3 Quotients between preference-approvals and weak orders. 75
5.4 Values of h for $\lambda = 0.5$. 78
5.5 Values of h for $\lambda = 0.75$. 78
5.6 Cophenetic dendrogram correlations for $m = 2$, $r = 1, 1.5, 2, 5, 10$ and $\lambda = 0.5$. 87
5.7 Cophenetic dendrogram correlations for $m = 3$, $r = 1, 1.5, 2, 5, 10$ and $\lambda = 0.5$. 87
5.8 Cophenetic dendrogram correlations for $m = 4$, $r = 1, 1.5, 2, 5, 10$ and $\lambda = 0.5$. 88
5.9 Cophenetic dendrogram correlations for $m = 5$, $r = 1, 1.5, 2, 5, 10$ and $\lambda = 0.5$. 88
5.10 Cluster central permutations. 89
5.11 Multinomial probability vectors. 90
5.12 Average adjusted Rand index over r and θ. 90
5.13 Votes in France. 92
5.14 Votes in the EU. 93
5.15 Distance between countries and representative cluster preference-approvals. 98

6.1 Values in the EU. 110
6.2 Votes in Italy. 110
6.3 ExpectedRank and RelativeApproval. 113
6.4 Distances $\delta_{0.5}$. 113
6.5 Votes in the EU. 116
6.6 Lines of action. 117
6.7 Linguistic terms. 117
6.8 Pew Research Center example. 117

7.1 Relationship between giving neutral answers and failing at least one control topic evaluation 141
7.2 Control topics' scores assigned by annotators, reliable annotators are highlighted in red. 142
7.3 Coherence scores: the **S** matrix 144

7.4 Ranking coherence scores: the **R** matrix 144
7.5 Emond and Mason τ_x rank correlation coefficient with human judgments for metrics. 145

B.1 Variable Importance at final step for Top7Movies dataset 171

D.1 Spearman rank correlation coefficient and Pearson correlation coefficient with human judgments for metrics without noise 178
D.2 Coherence scores 179
D.3 Ranking coherence scores 180

Chapter 1

Introduction

A widespread data collection practice for gathering preference data is to ask a group of people to provide their opinion on a finite selection of choices. Preference rankings arise when voters are asked to order the alternatives from best to worst. The main issues concerning the analysis of preference rankings are the aggregation and the prediction of preferences. The aggregation of preferences consists of identifying a compromise or a "consensus"[1]. The prediction of preferences, also called Label Ranking, consists of building preference models that learn to predict the ranking responses of new instances based on a set of predictor characteristics. One of the most effective ways to tackle these two tasks is to employ distance-based methods. The approaches in the literature are based on the classical unweighted rank distance measures. Thus, they are not sensitive to the individual importance of alternatives. Nevertheless, in many settings, assigning erroneously the ranking position of a highly relevant label should be considered more serious than making a mistake in assigning a negligible one. Moreover, an efficient classifier should be able to take into account the similarity between the elements to be ranked. This thesis initially deals with weighted aggregation and prediction of preferences. To this aim, we propose a new element-weighted rank correlation coefficient, $\tau_{x,e}$, as an extension of Emond and Mason (2002) τ_x, and a new element-weighted rank distance, $d_{K,e}$, as an extension of the Kemeny and Snell (1962a) distance d_K. The two proposed measures are then used to build an algorithm to identify the weighted consensus ranking and an algorithm to perform weighted Label Ranking.

[1] It should be noted that there is no single definition of "consensus" in the scientific community. In this thesis, following the statistical community (D'Ambrosio et al., 2015, 2017a,b), the consensus represents the median ranking, i.e. the result of preference aggregation. While in the Social Choice community, the term consensus most commonly refers to the degree of agreement among the set of judges (García-Lapresta and Pérez-Román, 2010).

Preference rankings consider a preferential order among alternatives without distinguishing between acceptable and unacceptable alternatives. That is, if a is ranked above b, we can only infer that a is preferred to b, but we cannot infer anything about their absolute acceptability. In several surveys of real data, voters are asked to indicate a binary qualitative judgement on alternatives in addition to their preference rankings. For instance, evaluators in a project finance review process are required to rank the potential projects and determine which ones should receive financial support. The approval voting system (Brams and Fishburn, 1978) separates the set of acceptable alternatives from the unacceptable alternatives without considering any preferential ordering. In other words, voters draw an imaginary cut-off line that separates acceptable and unacceptable alternatives. Combining preference rankings and approval voting gives preference-approval structures. When dealing with preference-approvals, the expressivity of voters explodes. To address the complexity of the preference-approval space, developing clustering methods is one of the most urgent issues. This thesis proposes the definition of new distances in the preference-approval context to deal with the clustering task. Firstly, a new metric between preference-approvals is proposed and compared to the existing distance functions to deal with voters' clustering. Secondly, a pseudometric on the set of alternatives is presented and used for clustering alternatives to reduce the complexity of the preference-approval space and provide a more accessible interpretation of the data.
The task of providing a ranking of alternatives is valuable in several scientific fields. For example, in the field of textual analysis, when dealing with topic models, providing a ranking of the estimated latent topics is beneficial. In fact, many times, not all of a model's estimated topics are semantically coherent and correspond to genuine domain themes. Some topics can be a collection of irrelevant or unchained words representing insignificant themes. Therefore, developing a ranking method that automatically ranks learned topics closely matching human judgments is desirable. The last chapter of the thesis offers a new coherence metric for determining a final ranking of topics based on their semantic interpretability.

1.1 Outline of the thesis

We report a brief description of the thesis chapters;

- **Introduction**
  Chapter 1 outlines all preliminary concepts that will be used throughout the dissertation and provides a quick introduction to the issues discussed in the thesis.

- **Preliminaries on rankings**
  Chapter 2 highlights the primary classical distances used to measure disagreement between rankings and the correlation coefficients associated with each distance. It also addresses two critical issues: ranking prediction and aggregation.

- **Element weighted Kemeny distance for ranking data**
  The third chapter investigates the consensus between rankings taking into account the importance of items (element weights). For this purpose, it includes a new element-weighted rank correlation coefficient, and its corresponding element weighted ranking distance. The procedure to obtain the weighted consensus ranking among several individuals is described, and its performance is studied by simulation and application to real datasets. A scientific paper extracted from this chapter (Albano and Plaia, 2021) has already been published.

- **A weighted distance-based approach with boosted decision trees for Label Ranking**
  The main contribution of the fourth chapter is to formulate a flexible Label Ranking ensemble model which encodes the similarity structure and a measure of the individual label importance to predict rankings. Precisely, the proposed method consists of three item-weighted versions of the AdaBoost boosting algorithm for label ranking. Our proposal's predictive performance is investigated through simulations and applications to three real datasets. A scientific paper extracted from this chapter (Albano et al., 2022b) has already been published.

- **A family of distances for preference-approvals**
  The fifth chapter proposes a new method for defining the distance between preference-approvals, taking into account the disagreements in preferences and approvals for each pair of alternatives jointly. The proposed distance is compared to the existing distance functions to deal with clustering problems. Specifically, we prove that our metric improves the estimated clusters in terms of stability and accuracy. A scientific paper extracted from this chapter (Albano et al., 2022a) has already been published.

- **A new pseudometric for clustering alternatives in preference-approvals**
  The sixth chapter proposes a new procedure for clustering alternatives in order to reduce the complexity of the preference-approval space and provide a more accessible interpretation of data. To that end, we present a new family of pseudometrics on the set of alternatives that take into account voters' preferences

via preference-approvals. A scientific paper extracted from this chapter 6 was submitted to a scientific journal.

- **Ranking coherence in Topic Models using Statistically Validated Networks**
  The seventh chapter offers a new ranking method, based on Statistically Validated Networks (SVNs), to explore the quality of topic models. The proposed method allows one to distinguish between high-quality and low-quality topics using a battery of statistical tests. We demonstrate the method's effectiveness through an analysis of a real text corpus, showing that the proposed measure correlates more with human judgement than the state-of-the-art coherence measures. A scientific paper extracted from this chapter has been accepted for publication in a scientific journal.

- **Conclusions**
  In the last Chapter the conclusions are drawn.

CRediT author statement

- Alessandro Albano and Antonella Plaia (2021). Element weighted Kemeny distance for ranking data. *Electronic Journal of Applied Statistical Analysis*, 14(1), 117-145. **Published.**
  Alessandro Albano: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Visualization. Antonella Plaia: Conceptualization, Methodology, Supervision.

- Albano, Alessandro, Mariangela Sciandra, and Antonella Plaia (2022). A weighted distance-based approach with boosted decision trees for label ranking. *Expert Systems with Applications*, 213:119000. **Published.**
  Alessandro Albano: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Visualization. Mariangela Sciandra: Conceptualization, Supervision. Antonella Plaia: Conceptualization, Formal analysis, Methodology, Supervision.

- Albano, A., García-Lapresta, J. L., Plaia, A., and Sciandra, M. (2022). A family of distances for preference–approvals. *Annals of Operations Research*, 1-29. **Published.**
  Alessandro Albano: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Visualization. José Luis García Lapresta: Conceptualization, Methodology, Writing - Review &

Editing, Supervision. Mariangela Sciandra: Conceptualization, Methodology, Supervision. Antonella Plaia: Conceptualization, Formal analysis, Supervision.

- Alessandro Albano, José Luis García Lapresta, Mariangela Sciandra, Antonella Plaia (2022). A new pseudometric for clustering alternatives in preference-approvals. **Under review**
  Alessandro Albano: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Visualization. José Luis García Lapresta: Methodology, Writing - Review & Editing, Supervision. Mariangela Sciandra: Conceptualization, Formal analysis, Supervision. Antonella Plaia: Conceptualization, Formal analysis, Supervision.

- Andrea Simonetti, Alessandro Albano, Michele Tumminello and Antonella Plaia (2022). Ranking coherence in Topic Models using Statistically Validated Networks. **Accepted for publication in Journal of Information Science.**
  Andrea Simonetti: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Visualization. Alessandro Albano: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Visualization. Antonella Plaia: Conceptualization, Formal analysis, Supervision. Michele Tuminello: Conceptualization, Methodology, Formal analysis, Supervision

Chapter 2

Preliminaries on rankings

Ranking is one of the most effective cognitive processes used by people to handle many aspects of their lives. It is also a simple and efficient data collection technique to understand individuals' perceptions and preferences for some items. When some subjects are asked to indicate their preferences over a set of alternatives, ranking data are called preference data. Therefore, preference data arise when a group of n individuals (e.g. judges, experts, voters, raters) express their preferences for a finite set of m items (labels, elements or alternatives). Preference data can be expressed by *ordering* the items (when alternatives are placed in order from best to worst) or using *rankings* (when alternatives are fixed in any pre-specified order and preferences are expressed by using integers to indicate the rank of each alternative). In other words, the ordering is the vector of labels ordered from best to worst, while the ranking is the vector of integers indicating the preferential order among the labels. Indeed, it is always possible to derive the ranking from the corresponding ordering and vice versa.

2.1 Notation

Formally, given the finite set of alternatives (or class labels) $Y = \{y_1, \ldots, y_m\}$, the ranking π is a mapping from Y to the set of ranks $\{1, \ldots, m\}$, endowed with the natural ordering of integers; $\pi = \{P_\pi(y_1), \ldots, P_\pi(y_i), \ldots, P_\pi(y_m)\}$, where $P_\pi : Y \longrightarrow \{1, \ldots, m\}$ is the rank of each alternative, being 1 for the alternative ranked first, 2 to the alternative ranked second, and so on.
If the m items, $\{y_1, \ldots, y_m\}$, are ranked in m distinguishable ranks, a complete (full) ranking or linear ordering is achieved (Cook, 2006), $\pi \in L(Y)$ where $L(Y)$ denotes the

set of linear orders on Y. Assigning positions to alternatives in linear orders is trivial because indifferences among distinct alternatives are not allowed. For example, given 5 items, say $Y = \{y_1, y_2, y_3, y_4, y_5\}$, the ordering $(y_2 \succ y_3 \succ y_4 \succ y_1 \succ y_5)$ corresponds to the ranking $\pi_1 = (4, 1, 2, 3, 5)$. The ranking π_1 is, in this case, one of the 5! (or $m!$ with m items) possible permutations of 5 elements.

When some items receive the same preference, then a tied ranking or a weak ordering is obtained, $\pi \in W(Y)$ where $W(Y)$ denotes the set of weak orders on Y. There are different ways of assigning positions to the alternatives in weak orders. Here we follow the one used by García-Lapresta and Pérez-Román (2011), that is based on Smith (1973), Black (1976) and Cook and Seiford (1982).
Given $\pi \in W(Y)$, the position of $y_i \in Y$ in π is assigned through the mapping $P_\pi : Y \longrightarrow [1, m]$ defined as:

$$P_\pi(y_i) = m - \#\left\{y_k \in Y \mid y_i \succ y_k\right\} - \frac{1}{2} \cdot \#\left\{y_k \in Y \setminus \{y_i\} \mid y_i \sim y_k\right\}. \qquad (2.1)$$

Where, given a generic set T, $\#T$ denotes the cardinality of T. For example, the weak ordering $(y_2 \succ y_1 \sim y_3 \succ y_4 \succ y_5)$, where the judge likes y_1 and y_3 equally well (i.e. the items are tied), corresponds to the ranking $\pi_2 = (2.5, 1, 2.5, 4, 5)$. Finally, in real situations, sometimes not all items are ranked: we observe partial rankings when judges are asked to rank only a subset of items (for example, only $m-1$ items), and incomplete rankings when judges can freely choose to rank only some items.

2.2 Distances and correlation for rankings

Because of their data reduction properties and ease of acquisition and representation, rankings have gained significant attention in the past few years. Within this framework, one of the main issues is evaluating the distance and the correlation between two rankings. The most famous correlation measures between rankings include Kendall's τ_b, later generalized by Emond and Mason (2002) τ_x.
As regards the distances, several measures have been proposed for ranking data (Kemeny and Snell 1962a; Spearman 1987). Given a set X, a distance is a function $d : X \times X \to \mathbb{R}$ where, for all π_1 and $\pi_2 \in X$, holds:

1. reflexivity $d(\pi_1, \pi_1) = 0$;
2. positivity $d(\pi_1, \pi_2) \geq 0$

3. symmetry $d(\pi_1, \pi_2) = d(\pi_2, \pi_1)$.

A distance measure is said to be a metric when it satisfies the triangle inequality:

4. triangle inequality $d(\pi_1, \pi_2) \leq d(\pi_1, \pi_3) + d(\pi_3, \pi_2)$, $\forall \pi_3 \in X$.

Finally, d is said to be a pseudometric if it does not satisfy the identity of indiscernibles:

5. identity of indiscernibles $d(\pi_1, \pi_2) = 0$ if and only if $\pi_1 = \pi_2$.

Kemeny and Snell (1962a) introduced a metric defined on linear and weak orders, known as Kemeny distance (or metric), later generalized to the framework of partial orders by Cook et al. (1986), which satisfies the constraints of a distance measure suitable for rankings.
Cook (2006) highlights the difficulties in treating the Kemeny metric, an issue already underlined by Emond and Mason (2002) and connected to the mathematical formulation using absolute values (see Eq.(2.7)). For this reason, the latter introduced a new correlation coefficient, strictly related to the Kemeny distance, and proposed using this coefficient as a basis for deriving a consensus among a set of rankings.
A correlation coefficient takes values between -1 and +1, i.e. rankings in full agreement are assigned a correlation of $+1$, those in full disagreement are assigned a correlation of -1, and all others lie in between. A distance d between two rankings, instead, is a non-negative value, ranging in $[0, Dmax]$, where 0 is the distance between a ranking and itself, while $Dmax$ varies among distances. This makes the correlation coefficient more intuitive as a measure of agreement between rankings. Considering the (finite) set S of all weak orderings of m objects, any rank correlation coefficient on S is also a distance metric on S, and vice versa. A distance metric d can be transformed into a correlation coefficient c (and vice-versa) using the linear transformation $c = 1 - \frac{2d}{Dmax}$.

2.2.1 Kendall's correlation coefficient τ_b

Kendall's correlation coefficient is probably the best-known measure for ranking data (Kendall, 1948). It can be calculated by creating a score matrix of a ranking. A rank vector π with m objects can be transformed into a symmetric $m \times m$ score matrix[1],

[1] It is worth noting that the *score matrix* $O_{\pi_1}(x_i, x_j)$ is also known in the literature as a_{ij} (see Kemeny and Snell 1962a, p. 11 or see Emond and Mason 2002).

whose elements $O_{\pi_1}(y_i, y_j)$ are defined by:

$$O_{\pi_1}(y_i, y_j) = \begin{cases} 1 & \text{if } y_i \text{ is preferred to } y_j \ (y_i \succ y_j) \\ 0 & \text{if } y_i = y_j \text{ or } y_i \text{ is tied with } y_j \ (y_i \sim y_j) \\ -1 & \text{if } y_j \text{ is preferred to } y_j \ (y_j \succ y_i) \end{cases} \tag{2.2}$$

Kendall's correlation coefficient τ_b between two rankings, π_1 with score matrix $O_{\pi_1}(y_i, y_j)$ and π_2 with score matrix $O_{\pi_2}(y_i, y_j)$ is defined as:

$$\tau_b(\pi_1, \pi_2) = \frac{\sum_{i=1}^{m}\sum_{i=1}^{m} O_{\pi_1}(y_i, y_j) O_{\pi_2}(y_i, y_j)}{\sqrt{\sum_{i=1}^{m}\sum_{i=1}^{m} O_{\pi_1}(y_i, y_j)^2 \sum_{i=1}^{m}\sum_{i=1}^{m} O_{\pi_2}(y_i, y_j)^2}}. \tag{2.3}$$

When two rankings are the reversal of each other τ_b is equal to -1. When comparing linear orderings, the denominator always works out to the constant $m(m-1)$. Conversely, when comparing weak orderings the denominator will compute to a lesser value, reduced according to the total number of ties declared in each ranking. Emond and Mason (2002) pointed out that an all-ties ranking results in a zero-filled score matrix and can never be estimated as a solution, because of the zeros in the numerator divided by zeros in the denominator results in an unknown number. Kendall's correlation coefficient is a measure of similarity and can be transformed into a dissimilarity or distance measure via the linear transformation $d_{\tau_b} = 1 - \tau_b$, where d_{τ_b} is Kendall's distance.

2.2.2 Emond and Mason's correlation coefficient τ_x

When dealing with tied rankings, Emond and Mason (2002) showed that Kendall's distance (d_{τ_b}) violates the triangle inequality. To solve this difficulty, they redesigned the elements in Kendall's τ_b score matrix in Eq.(2.2) and renamed it to τ_x. The elements in the new score matrix $o_\pi(y_i, y_j)$ for rank vector π are now defined by:

$$o_\pi(y_i, y_j) = \begin{cases} 1 & \text{if } y_i \text{ is preferred or tied with } y_j \ (y_i \succeq y_j) \\ 0 & \text{if } y_i = y_j \\ -1 & \text{if } y_j \text{ is preferred to } y_i \ (y_j \succ y_i). \end{cases} \tag{2.4}$$

The new correlation coefficient is defined as:

$$\tau_x(\pi_1, \pi_2) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{m} o_{\pi_1}(y_i, y_j) o_{\pi_2}(y_i, y_j)}{m(m-1)}. \tag{2.5}$$

When ties are not allowed, τ_x reduces to τ_b; the former differs from the latter in giving a score of 1 to ties instead of 0; this allows to solve the known Kendall's problems with weak orderings.

2.2.3 Spearman's distance d_s

The Spearman's distance is calculated by taking the square root of the well-known Spearman's ρ. The distance between two rank vectors π_1 and π_2 is defined by:

$$d_s(\pi_1, \pi_2) = \sqrt{\sum_{i=1}^{m} (\pi_1(y_i) - \pi_2(y_i))^2}. \tag{2.6}$$

When a ranking contains tied objects, these objects must be given the average of the corresponding rank values. A problem identified by Emond and Mason (2000) is that Spearman's d_s suffers from what is known as the sensitivity to irrelevant alternatives (an irrelevant alternative is one that is asymmetrically dominated, this means that the object is less preferred in every ranking to another object but not by every other object (Emond and Mason, 2000)). In other words, adding extra irrelevant objects to the ranking exercise could change the maximum agreement solution. This technical flaw arises because Spearman's d_s treats the ranks as numerical values instead of categorical ordered values. Because of this sensitivity to irrelevant alternatives, Spearman's d_s is unsuitable as a rank correlation coefficient in the weighted rankings problem.

2.2.4 Kemeny distance d_K

Kemeny (1959) introduced several properties that a suitable distance measure for rankings should satisfy:

1. reflexivity, positivity, symmetry and the triangular inequality;
2. the measure of distance should not be affected by a relabeling of the set of objects to be ranked;
3. if two rankings are in complete agreement at the beginning and at the end of the list and differ only in the middle, than the distance does not change after deleting both the first and the last objects to be ranked;
4. the minimum positive distance is one,

and introduced a distance, d_K, that satisfies all these constraints.

The Kemeny distance d_K between two rankings of size m, π_1 with score matrix $O_{\pi_1}(y_i, y_j)$ and π_2 with score matrix $O_{\pi_2}(y_i, y_j)$ (defined in Eq.(2.2)) is a city block distance defined as:

$$d_K(\pi_1, \pi_2) = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \left| O_{\pi_1}(y_i, y_j) - O_{\pi_2}(y_i, y_j) \right|. \tag{2.7}$$

The Kemeny metric is a city block distance taking the shortest path between two rankings. The factor $\frac{1}{2}$ takes into account that the two triangular matrices that are created by the sum of absolute differences of the score matrices are identical. The maximum distance from a complete ranking to its reversal is $m(m-1)$ while the maximum distance of a ranking containing t ties is given by: $m(m-1) - 2t$.

Considering the usual relation between a distance d and its corresponding correlation coefficient $\tau = 1 - \frac{2d}{Dmax}$, where $Dmax$ is the maximum distance, d_K is in a one-to-one correspondence to the rank correlation coefficient τ_x proposed by Emond and Mason (2002).

The Kemeny distance is a geodesic distance in the *permutation polytope*, or *permutahedron* (see Thompson 1993; Heiser 2004). The permutation polytope is defined as "the convex hull of all vectors that are obtained by permuting the coordinates of a vector containing the first m integers" (Heiser, 2004). It is a convex figure containing the $m!$ permutations of m objects. The convex hull forms an $(m-1)$-dimensional object, in the intersection of a hypersphere and a hyperplane, graphically representable only for $m \leq 4$. If weak orders are considered, the permutation polytope is extended to include permutations of nondistinct values. Thus, the *generalized permutation polytope* is defined as the convex hull of the points in $\mathbb{R}^m$ whose coordinates are permutations of m not necessarily distinct values. Figure 2.1 shows the generalized permutation polytope in the case of $m = 4$ alternatives, named (A,B,C,D).

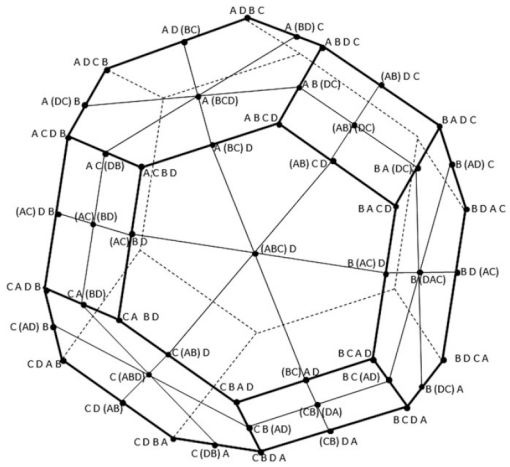


Figure 2.1: Permutation polytope for full and partial rankings of four objects (Heiser and D'Ambrosio, 2013).

The links in the figure indicate a switch from one inequality to equality, except for the lines in the hexagons that connect to partial rankings with tie-blocks of three, which represent two switches. The natural graphical distance in the generalized permutation polytope is the sum of the line segments that must be crossed along the shortest path to get from one node to another, and this distance is equivalent to the count of the smallest number of interchanges of two adjacent elements required to transform one ranking into another. That is, the natural distance measure is the Kemeny distance.

2.3 Aggregation of rankings: the consensus problem

There are many approaches for identifying a ranking representative of a group of judges. Arrow (1951) presented certain desirable features that a ranking system must have and demonstrated that no rule could meet all of them simultaneously. As a result, several preference aggregation models have been proposed, each satisfying subsets of desirable criteria. The first models proposed belong to the class of voting methods (or counting methods). The most popular counting methods are: the Borda count (Borda, 1781), which counts the total rank for each alternative, and the Condorcet method (Condorcet, 1785), which is based on pairwise comparisons of alternatives. More recently, statisticians and computer scientists studied the problem from numerous angles to try to establish an aggregation technique. Kemeny and Snell (1962a) suggested using a distance function to characterize the median ranking as a specific definition of

consensus ranking. They defined the median ranking as that ranking that minimizes the sum of the Kemeny distance between itself and all other orderings in the sample of judges. According to Arrow's Axioms, the *Kemeny ranking rule* is the only rule that meets the independence of irrelevant alternatives and the reinforcement axiom (Young and Levenglick, 1978; Ali and Meilă, 2012). Finding the Kemeny ranking is regrettably a computational difficulty, as the problem is NP-hard even with only four votes (Bartholdi et al., 1989; Cohen et al., 1999). Because the topic is significant in so many domains, many academics have focussed on developing excellent, practical methods to solve it. Emond and Mason (2002) proposed a Branch-and-Bound algorithm to solve the consensus ranking problem. They introduced the Combined Input Matrix (CI), defined as the summation of each input ranking scorematrix. In this way, the rankings information is stored in a single matrix. Then, the consensus ranking is identified, within the set of all weak orderings of m objects, through an iterative process that makes use of a system of increasing penalties. Branch-and-bound algorithms are accurate and helpful in many practical applications, although they are slow, especially when the number of objects is high, or the degree of internal consensus in the data is weak. For this reason, Amodio et al. (2016) and D'Ambrosio et al. (2015) proposed two accurate algorithms (QUICK and FAST) as an alternative to Emond and Mason (2002)'s branch-and-bound algorithm to provide savings in computational time. Later, D'Ambrosio et al. (2017b) developed a differential evolution algorithm, called DECoR, for the median ranking detection under Kemeny's axiomatic framework. They compared their proposal with both branch-and-bound and other heuristic algorithms showing that when the number of objects is larger than 100, the DECoR algorithm is enormously faster than the QUICK, preserving the same degree of accuracy.
Alternative procedures for aggregating rankings based on different axiomatic frameworks and distance metrics have been developed (Cook and Seiford, 1978; Cook, Wade D. et al., 1986; Cook et al., 2007). In this thesis, due to the desirable mathematical properties and geometric interpretation of the Kemeny distance, we focus on the median ranking approach for finding the consensus ranking. This method will be explored further in later sections and extended to incorporate weighted distances.

2.4 Prediction of rankings: the Label Ranking task

Label Ranking (LR) is an emerging non-standard supervised classification problem with practical applications in different research fields. The Label Ranking task aims at building preference models that learn to order a finite set of labels based on a set of predictor features. The Label Ranking (LR) task is an extension of the conven-

tional classification setting. Given an instance $\mathbf{x} = \{x_1, x_2, \ldots, x_K\}$ from the instance space $\mathscr{X}$, we associate $\mathbf{x}$ with a ranking π of the labels in the finite set of class labels $Y = \{y_1, \ldots, y_m\}$. The predictive performance of the LR-classifier is one of the main issues to investigate. Typically, given an instance $\mathbf{x}_l$ with label ranking π_l, and the ranking π_l^* predicted by an LR model, the discrepancy between π_l and π_l^* is measured through a suitable distance function.

Label Ranking is associated with three relevant supervised learning problems: multiclass classification, multilabel classification and multilabel ranking. In multiclass classification, each instance is associated with a single label; this can be seen in our framework as a ranking where only the first (top-1) position in a ranking matters. In multilabel classification, we are interested in the bipartite partition of all class labels into relevant and irrelevant labels: that is to say, a ranking where we are interested only in the top-k positions where all items are tied. Finally, multilabel ranking can be considered a generalization of multilabel classification and label ranking. The goal is to identify relevant labels from a set of predefined class labels and rank them according to their relevance. It is a typical top-k ranking or what we call partial ranking. Ranking data is a simple and efficient data collection technique. The label ranking task has gained significant attention in the last decade to understand the perceptions and preferences of individuals for some items. Due to its practical significance, Label Ranking has been applied in many different fields, and a large number of methods have been proposed or adapted for label ranking. Zhou et al. (2014) conducted a review of existing label ranking methods and provided a basic taxonomy of these methods. In their work, they distinguish: i) *Decomposition* methods such as Log-linear models (Dekel et al., 2003), Constraint classification (Har-Peled et al., 2003) and Ranking by pairwise comparison (Hüllermeier et al., 2008); ii) *Probabilistic* methods such as Mallows Models (Cheng and Hüllermeier, 2009), Plackett-Luce (Cheng et al., 2010), Gaussian mixture model (Grbovic et al., 2012), Decision trees (Cheng et al., 2009); iii) *Similarity* approaches such as Naive Bayes (Aiguzhinov et al., 2010), Association rules (De Sá et al., 2011), Multilayer perceptron (Ribeiro et al., 2012).

The tree-based approaches (Cheng et al., 2009) have become the most popular techniques in the last years due to their ease of interpretation. Specifically, Cheng et al. (2009) proposed Label Ranking Trees (LRT), one of the first label ranking methods based on a decision tree algorithm. As a core component, their approach estimates local models, assuming that the probability distribution of the output is locally constant, by deriving an approximation of an ML estimation based on the Mallows model. Lee and Philip (2010) combined the strength of a tree model and the existing distance-based models to build a model that can handle ranking data. They introduced a recursive par-

titioning algorithm for building a tree model with a distance-based ranking model fitted at each leaf.

Philip et al. (2010) established a new decision tree model for the analysis of ranking data; they modified the existing splitting criteria to let them precisely measure the impurity of a set of ranking data. They also introduced types of impurity measures for ranking data, namely *g-wise* and top-k measures. Similarly, Plaia and Sciandra (2019) proposed using a univariate decision tree for ranking data based on the positional-weighted distances for complete and incomplete rankings and considers the area under the ROC curve as a tool both for pruning and model assessment.

In decision tree learning, the classifier's predictive performance is substantially improved by aggregating many decision trees. For this reason, the LR community has also devoted increasing attention to ensemble methods in recent years.

Aledo et al. (2017), inspired by the decision tree algorithm (LRT), designed two weak tree-based classifiers. They showed through an experimental study that bagging these weak learners, using unsupervised frequency-based discretization to select the split point, is competitive with the ensemble of LRT and state-of-the-art algorithms in terms of accuracy.

de Sá et al. (2017) proposed an ensemble of decision trees for Label Ranking, based on Random Forests, called Label Ranking Forests (LRF). Their method is tested with two base-level methods: Ranking Trees (RT) and Entropy Ranking Trees (ERT). In a similar work, Zhou and Qiu (2018) presented a random forest label ranking method using random decision trees to retrieve nearest neighbours. They developed a two-step rank aggregation strategy based on Borda's method to aggregate neighbouring rankings discovered by the random forest into a final predicted ranking.

Werbin-Ofir et al. (2019) studied the aggregation sub-task of label ranking ensembles. They proposed a novel aggregation method called Voting Rule Selector (VRS), a flexible approach that learns the best rule for a given dataset. The authors claimed that their algorithm can be easily incorporated in every setting, which involves label ranking ensembles to improve their overall prediction performance, and may also be valuable in various other fields concerned with aggregation of rankings.

Dery and Shmueli (2020) presented a novel boosting-based algorithm, BoostLR, for improving the prediction performance of label ranking ensembles. They used Kendall's τ_b coefficient to calculate the loss between the predicted and actual rankings and weighted and Borda's count as aggregation method.

Plaia et al. (2017, 2021a) proposed a theoretical and computational definition of bagging and boosting for label ranking; their approach considers decision tree as weak learners, the Kemeny distance (Kemeny, 1959) as a measure of impurity in the split-

ting process, and its related rank correlation coefficient τ_x (Emond and Mason, 2002) for identifying the median ranking in the final nodes.

Chapter 3

Element weighted Kemeny distance for ranking data

3.1 Introduction

In general, distances between rankings consider all items equally important and are not sensitive to where the disagreement occurs. Kumar and Vassilvitskii (2010) introduced two essential aspects for many applications involving distances between rankings: positional weights and element weights. Positional weights allow to take into account the particular position of disagreement between two rankings when computing their distance/similarity, i.e., for example, the researcher may want to consider swapping elements near the head of a ranking more critical than swapping elements in the tail of the ranking.

Conversely, element weights refer to the role played by the objects that judges are ranking: in certain situations, swapping some particular objects should be less penalizing than swapping others. For example, let us consider a survey in which a group of people is asked to rank ten social networks. In this case, it would be reasonable to assign weights proportional to the social network's stock market value (e.g. Facebook would receive the highest weight) so that a disagreement between two popular platforms receives a larger penalization than an inversion between less famous ones. In other words, if two judges agree in assigning the position of the most important alternatives, they will be highly positively correlated. Another example comes from the voting theory: when ranking politicians, the weights allow taking into account that some candidates are similar (belonging to the same party or political coalition) and that transposing sim-

ilar candidates induces a smaller cost than transposing dissimilar candidates. Here two judges that commit many inversions between politicians coming from different parties will be negatively correlated.

A critical issue involving rankings concerns the aggregation of the preferences in order to identify a compromise or a "consensus" (Kemeny and Snell 1962b, Fligner and Verducci 1990). The most popular approaches to cope with this problem are related to distances/correlations (Kemeny and Snell 1962b,Cook et al. 1986, Fagot 1994, D'Ambrosio and Heiser 2016). As a matter of fact, in order to obtain homogeneous groups of subjects with similar preferences, it is natural to measure the spread between rankings through dissimilarity or distance measures. In this sense, a consensus is defined as the closest ranking (i.e. with the minimum distance) to the whole set of preferences. Another possible way to measure (dis)-agreement between rankings in a consensus problem is a correlation coefficient.

Even in this case, a weighted procedure to perform rank aggregation would prove beneficial. While a position weighted correlation coefficient τ_x^w, for both linear and weak ordering, has been proposed by Plaia et al. (2021b), here we aim at introducing an element weighted correlation coefficient called $\tau_{x,e}$ as an extension of τ_x provided by Emond and Mason (2002), and a new weighted distance called $d_{K,e}$ as an extension of Kemeny distance. We will prove that the proposed correlation coefficient reduces to Emond and Mason's τ_x when equal weights are set, and that it is related to the proposed distance through the linear transformation $\tau_{x,e} = 1 - \frac{2d_{K,e}}{Dmax}$.

The proposed weighted correlation coefficient will be used to deal with a consensus ranking problem, i.e. to find the ranking which best represents the rankings/preferences expressed by a group of judges.

The chapter is organized as follows. The next section deals with the introduction of element weights in the distance definition. In Section 3.3, some intuitive methods to assign weights to elements are discussed. The algorithm for finding the consensus ranking is described in Section 3.4. In Section 3.5, the algorithm is applied to simulated and real data. Finally, the concluding remarks are presented in Section 3.6.

3.2 Item weighted distances and correlation coefficient

Kumar and Vassilvitskii (2010) introduced two issues that are essential for many applications involving distances between rankings, namely, positional weights and element weights. The issue of positional weights has been explored by relevant researches (García-Lapresta and Pérez-Román 2010; Can 2014; Plaia et al. 2018, 2019). In this

chapter, we deal with case ii) and propose the weighted version of the Kemeny metric and the correlation coefficient introduced by Emond and Mason (2002).
The weighting vector $w = (w_1, w_2, \ldots, w_m)$ with $w_i \geq 0$ is used to take into account the importance of the items where w_i is the importance given to the i^{th}-item in a ranking.

3.2.1 Introducing element weights in the Kemeny distance

There are many ways to introduce weights in a distance measure, and each of them corresponds to a different penalization of each inversion between two generic items in two rankings. For example, one can decide that an inversion of elements y_i and y_j should have a penalty proportional to the arithmetic average of their weights, say $\frac{w_i+w_j}{2}$. The corresponding weighted version of the Kemeny distance, in this case, will be:

$$ {}_a d_{K,e}(\pi_1, \pi_2) = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \frac{w_i + w_j}{2} \left| O_{\pi_1}(y_i, y_j) - O_{\pi_2}(y_i, y_j) \right|, \tag{3.1} $$

where, O_{π_1} and O_{π_2} are the score matrices of rankings π_1 and π_2, as defined in the first Chapter, Eq.(2.2). It can be easily demonstrated that the maximum value of Eq.(3.1) is $(m-1)\sum_{i=1}^{m} w_i$. An alternative could be the product of the weights $w_i w_j$; the corresponding weighted Kemeny distance will be defined as:

$$ {}_p d_{K,e}(\pi_1, \pi_2) = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} w_i w_j \left| O_{\pi_1}(y_i, y_j) - O_{\pi_2}(y_i, y_j) \right|, \tag{3.2} $$

the maximum value of Eq.(3.2) being equal to $\sum_{i=1}^{m} \sum_{j=1}^{m} w_i w_j$. It can be proved that, regardless of the choice of weighting procedure, the mathematical properties of the Kemeny distance are preserved. Therefore, these methods should be compared in the light of their impact on the resulting weighted Kemeny distance.
Let's consider, for example, three different rankings of three items, say y_1, y_2 and y_3, $\pi_1 = (1,2,3)$, $\pi_2 = (2,1,3)$, $\pi_3 = (2,3,1)$ and a weighting vector $w = (10,10,1)$ (Tab. 3.1).

Table 3.1: Weighting vector and data matrix

w		
y_1	y_2	y_3
10	10	1

	Elements		
	y_1	y_2	y_3
π_1	1	2	3
π_2	2	1	3
π_3	2	3	1

Let us compute the weighted Kemeny distances between π_1 and the other rankings π_2, π_3 using the two penalization method discussed before (Tab. 3.2).

Table 3.2: Weighted Kemeny distances

Items	${}_a d_{K,e}$	${}_p d_{K,e}$
π_1 vs π_2	20	200
π_1 vs π_3	22	40

According to ${}_a d_{K,e}$, the distance π_1 vs π_3 (22) is slightly higher than π_1 vs π_2 (20) while ${}_p d_{K,e}$ claims the contrary, stating that π_1 vs π_3 (40) is far lower than π_1 vs π_2 (200). π_1 assigns the first position to item y_1, the second one to item y_2 and finally item y_3 is ranked third. With π_1 used as a reference, π_2 switches the ranks of y_1 and y_2 but keeps y_3 in the last position. π_3 changes the rank of every item moving y_3 to the first position, y_1 to the second one, and finally y_2 to the third one.

Apparently, π_3 changes more frequently the position of items, but it keeps unchanged the ordering of y_1 and y_2. That is to say, either π_3 and π_1 prefer y_1 to y_2, while π_2 doesn't. Since y_1 and y_2 are the most important elements according to the weighting vector w, their inversion should be overly penalized. This implies that π_3 resembles π_1 more than π_2 does.

To better understand the results, let us compute the *relative weight*, r_{ij}. The relative weight represents how much each inversion influences the resulting $d_{K,e}$. It is computed as the ratio of the weight of generic inversion between y_i and y_j over the total sum of weights.

When using the arithmetic average, the relative weight of each inversion between two

generic elements i and j is defined as follows:

$$ {}_a r_{ij} = \begin{cases} \frac{w_i + w_j}{(m-1)\sum_{i=1}^{m} w_i}, & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} \tag{3.3} $$

while in the case of the product, the relative weight of each inversion is:

$$ {}_p r_{ij} = \begin{cases} \frac{2 w_i w_j}{\sum_{i=1}^{m} \sum_{j=1}^{m} (w_i w_j)}, & \text{if } i \neq j \\ 0 & \text{if } i = j. \end{cases} \tag{3.4} $$

In both cases the relative weights must sum up to 1; $\sum_{i<j}^{m} r_{ij} = 1$. Let's compute the relative weights with the data of table(3.1):

Table 3.3: Relative weights ${}_a r_{ij}$ of each inversion with arithmetic average

	y_1	y_2	y_3
y_1	0	-	-
y_2	**0.476**	0	-
y_3	0.262	0.262	0

Table 3.4: Relative weights ${}_p r_{ij}$ of each inversion with product

	y_1	y_2	y_3
y_1	0	-	-
y_2	**0.834**	0	-
y_3	0.083	0.083	0

The inversion between y_1 and y_2, when using the arithmetic average, will "cost" approximately the 48% of the maximum obtainable Kemeny distance (table3.3). In contrast, when using the product (table3.4), the same inversion has a more considerable influence, equal to 83%. In this example, the product of weights turns out to be the most appropriate method. In broader terms, the product aggregation ${}_p d_{K,e}$ concentrates the mass of weights on the inversions of the most important items, while the arithmetic average ${}_a d_{K,e}$ distributes it more evenly.

The critical point for the researcher is to think about the relative weight of each inversion when assigning the individual weights. From now on, for the purpose of this thesis, the item-weighted Kemeny distance will be indicated as $d_{K,e}$, and it will employ the product of strictly positive weights ($w_i > 0$) as penalization, keeping in mind that the relative weights are what really matter.

3.2.2 A new weighted rank correlation coefficient

Combining the weighted Kemeny distance proposed, $d_{K,e}$, and the extension of τ_x provided by Emond and Mason (2002), we propose a new weighted rank correlation coefficient between two rankings π_1 and π_2:

$$\tau_{x,e}(\pi,\pi_2) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j o_{\pi_1}(y_i,y_j) o_{\pi_2}(y_i,y_j)}{\max[d_{K,e}]}, \tag{3.5}$$

where, o_{π_1} and o_{π_2} are the score matrices of rankings π_1 and π_2, as defined in the first Chapter, Eq.(2.4). While, the denominator of the formula represents the maximum value of the weighted Kemeny distance $\max[d_{K,e}] = \sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j$.

Correspondence between distance and correlation

Following the relation $\tau = 1 - \frac{2d}{Dmax}$, we prove the following equation:

$$\frac{\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j o_{\pi_1}(y_i,y_j) o_{\pi_2}(y_i,y_j)}{\max[d_{K,e}]} = 1 - \frac{2d_{K,e}}{\max[d_{K,e}]}. \tag{3.6}$$

PROOF:

$$\frac{\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j o_{\pi_1}(y_i,y_j) o_{\pi_2}(y_i,y_j)}{\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j} = 1 - \frac{\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j \left| O_{\pi_1}(y_i,y_j) - O_{\pi_2}(y_i,y_j) \right|}{\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j}$$

$$\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j o_{\pi_1}(y_i,y_j) o_{\pi_2}(y_i,y_j) = \sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j - \sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j \left| O_{\pi_1}(y_i,y_j) - O_{\pi_2}(y_i,y_j) \right|$$

$$\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j o_{\pi_1}(y_i,y_j) o_{\pi_2}(y_i,y_j) = \sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j \left(1 - \left| O_{\pi_1}(y_i,y_j) - O_{\pi_2}(y_i,y_j) \right|\right)$$

the left side and the right side of the equation are equal, the proof is due to Emond and Mason (2002). To prove this equality we will show that over any pair of objects y_i and

y_j the two summations correspond, i.e. that:

$$\begin{aligned}&\cancel{w_i w_j}(1-\left|O_{\pi_1}(y_i,y_j)-O_{\pi_2}(y_i,y_j)\right|)+\cancel{w_i w_j}(1-|O_{\pi_1}(y_j,y_i)-O_{\pi_2}(y_j,y_i)|)=\\&\cancel{w_i w_j}(o_{\pi_1}(y_i,y_j)o_{\pi_2}(y_i,y_j))+\cancel{w_i w_j}(o_{\pi_1}(y_j,y_i)o_{\pi_2}(y_j,y_i))\end{aligned} \tag{3.7}$$

There are nine possible combinations of preferences for objects y_i and y_j between rankings π_1 and π_2, but only four distinct cases must be considered. The other five are equivalent to one of these four through a simple relabelling of the rankings and/or the objects.

Case 1: π_1 prefers object y_i over y_l, as does π_2.
The values are: $O_{\pi_1}(y_i,y_j)=1$, $O_{\pi_1}(y_j,y_i)=-1$, $O_{\pi_2}(y_i,y_j)=1$, $O_{\pi_2}(y_j,y_i)=-1$. These yield the left side: $1-|1-1|+1-|(-1)-(-1)|=2$, the right side values are identical in this case: $o_{\pi_1}(y_i,y_j)=1$, $o_{\pi_1}(y_j,y_i)=-1$, $o_{\pi_2}(y_i,y_j)=1$, $o_{\pi_2}(y_j,y_i)=-1$ yield the same total: $(1)(1)+(-1)(-1)=2$.

Case 2: π_1 prefers object y_i over y_j, while π_2 ranks them as tied.
The values are: $O_{\pi_1}(y_i,y_j)=1$, $O_{\pi_1}(y_j,y_i)=-1$, $O_{\pi_2}(y_i,y_j)=0$, $O_{\pi_2}(y_j,y_i)=0$. These yield the left side: $1-|1-0|+1-|(-1)-0)|=0$. The right side values are: $o_{\pi_1}(y_i,y_j)=1$, $o_{\pi_1}(y_j,y_i)=-1$, $o_{\pi_2}(y_i,y_j)=1$, $o_{\pi_2}(y_j,y_i)=-1$ yield the same total: $(1)(1)+(-1)(1)=0$.

Case 3: π prefers object y_i over y_j, while π_2 prefers y_j over y_i.
The values are: $O_{\pi_1}(y_i,y_j)=1$, $O_{\pi_1}(y_j,y_i)=-1$, $O_{\pi_2}(y_i,y_j)=-1$, $O_{\pi_2}(y_j,y_i)=1$. These yield the left side: $1-|1-(-1)|+1-|-1-1|=-2$. The right side values are: $o_{\pi_1}(y_i,y_j)=1$, $o_{\pi_1}(y_j,y_i)=-1$, $o_{\pi_2}(y_i,y_j)=-1$, $o_{\pi_2}(y_j,y_i)=1$ yield the same total: $(1)(-1)+(-1)(1)=-2$.

Case 4: Both π_1 and π_2 rank the objects as tied.
The values are: $O_{\pi_1}(y_i,y_j)=0$, $O_{\pi_1}(y_j,y_i)=0$, $O_{\pi_2}(y_i,y_j)=0$, $O_{\pi_2}(y_j,y_i)=0$. These yield the left side: $1-|0-0|+1-|0-0|=2$. The right side values are: $o_{\pi_1}(y_i,y_j)=1$, $o_{\pi_1}(y_j,y_i)=1$, $o_{\pi_2}(y_i,y_j)=1$, $o_{\pi_2}(y_j,y_i)=1$ yield the same total: $(1)(1)+(1)(1)=2$. The two methods give identical results in all four distinct cases, completing the proof. ■

Minimum and maximum values of $\tau_{x,e}$

From the previous proofs, $\tau_{x,e}$ takes its maximum value, equal to 1, if and only if *Case 1* or *Case 4* are observed. Therefore, contrary to what happens with τ_x, $\tau_{x,e}$ assumes the maximum value even when a generic all tied ranking is compared with itself. Analogously, $\tau_{x,e}$ can be minimum and equal to -1, if and only for all y_i and y_j only *Case 3* occurs.

Correspondence between weighted and unweighted measures

For equal weights assigned to the items, $w_i = C$ with $i = 1,2,\ldots,m$ the weighted distance is proportional to the classic Kemeny distance.

$$d_{K,e} = C^2 d_K \tag{3.8}$$

PROOF:

$$d_K = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\left|O_{\pi_1}(y_i,y_j) - O_{\pi_2}(y_i,y_j)\right| \qquad d_{K,e} = \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j \left|O_{\pi_1}(y_i,y_j) - O_{\pi_2}(y_i,y_j)\right|$$

if $w_i = C$ for each $i = 1,\ldots,m \Rightarrow w_i w_j = C^2$ and $d_{K,e} = \frac{C^2}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\left|O_{\pi_1}(y_i,y_j) - O_{\pi_2}(y_i,y_j)\right|$.
■

Corollary. Since $\tau_x \equiv d_K$ and $\tau_{x,e} \equiv d_{K,e}$, the weighted rank correlation coefficient is equivalent to the rank correlation coefficient defined by Emond and Mason when equal importance is given to items: $\tau_{x,e} = \tau_x$ with $w_i = C$ for $i = 1,2,\ldots,m$.

3.2.3 The case of 0-weight items

Sometimes the $n \times m$ matrix Π, whose l^{th} row represents the ranking of the l^{th} judge (defined as in Table 3.1), contains some negligible items representing just noise. One may want to compute the weighted Kemeny distance between two or more rankings overlooking the set of irrelevant items. To do this, those elements are assigned a weight equal to 0. To deal with the 0-weight situation, the data matrix Π should be modified in order to lead back to the well-known case $w_i > 0$.
Let us define two rankings and one weighting vector: $\pi_1 = (1,2,3,4,5)$, $\pi_2 = (4,1,2,5,3)$ and $w = (0,1,1,1,0)$. The weighting vector states that elements y_1 and y_5 should not influence the distance between π_1 and π_2. We proceed to remove the first and fifth

Table 3.5: Weighting vector and original data matrix

w				
y_1	y_2	y_3	y_4	y_5
0	1	1	1	0

	Elements				
	y_1	y_2	y_3	y_4	y_5
π_1	1	2	3	4	5
π_2	4	1	2	5	3

columns of the data matrix, thus defining:

- two new rankings π_1' and π_2' that keep just the elements with a non-zero weight and re-assign the positions: $\pi_1' = (1,2,3)$, $\pi_2' = (1,2,3)$;
- a new weighting vector w' with all non-zero entries $w' = (1,1,1)$.

Table 3.6: Weighting vector and modified data matrix

w'		
y_2	y_3	y_4
1	1	1

	Elements		
	y_2	y_3	y_4
π_1'	1	2	3
π_2'	1	2	3

The new rankings π_1' and π_2' concern only the three items with non-zero weight: y_2, y_3 and y_4. It should be noted that element y_2 is ranked 2$^{\text{nd}}$ by π_1, while in the new ranking π_1' y_2 is ranked 1$^{\text{st}}$ since y_1 is removed, a similar situation is met for the other elements y_3 and y_4.
Therefore, the distance $d_{K,e}$ between π_1 and π_2 with weighting vector $w = (0,1,1,1,0)$ reduces to the distance $d_{K,e}$ between π_1' and π_2' with weighting vector $w' = (1,1,1)$, and it's equal to 0. This transformation easily allows us to move from the case of $w_i \geq 0$ to the case of $w_i > 0$.

3.2.4 A variant of element weights: the element similarities

Sometimes, for example, when dealing with multi-level data, the weights can be assigned following the *item similarity criterion*: i.e. swapping two elements that can be considered similar in some aspects should be less penalized than swapping two dissimilar ones. In this setting, a symmetric penalization matrix $\mathbf{P}_{m\times m}$, reflecting the

dissimilarity among the elements, is needed. In other words, the **P** matrix establishes the penalty ($p_{ij} = p_{ji}$) for each inversion of two generic items. The weighted Kemeny distance between two rankings π_1 and π_2 when using the item similarities method becomes:

$$d_{K,e}(\pi_1, \pi_2) = \sum_{i<j}^{m} p_{ij} \left| O_{\pi_1}(y_i, y_j) - O_{\pi_2}(y_i, y_j) \right| \tag{3.9}$$

where p_{ij} is the generic element of the penalization matrix **P**. The relative weight of each generic inversion can still be computed:

$$r_{ij} = \begin{cases} \frac{p_{ij}}{\sum_{i<j}^{m} p_{ij}}, & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} \tag{3.10}$$

with $\sum_{i<j}^{m} r_{ij} = 1$. To illustrate the notion of similarities, let us consider an example from voting theory. When dealing with rankings of politicians, it should be taken into account that candidates are gathered into political parties, which share common objectives and adhere to specific ideological areas. Swapping candidates from the same political party should have a smaller impact on the results of an election than swapping candidates from different parties. Let π_1, π_2, π_3 be three rankings of politicians.

Table 3.7: Data matrix

	Politicians		
	Clinton	Obama	Bush
π_1	1	2	3
π_2	2	1	3
π_3	1	3	2

The rankings π_2 and π_3 differ from π_1 only in one adjacent transposition. In the first case, the swap involves members of the same political party, while in the second case, the transposed candidates belong to two different parties. Hence it is reasonable to assume that the first distance should be smaller than the second one.
In this example, we decided to penalize swapping politicians of the same party with a weight equal to 1 while swapping politicians of different parties with a weight equal to 10. The penalization matrix is shown in Tab. 3.8, and the relative weights in Tab. 3.9.

Table 3.8: Penalization matrix

	Clinton	Obama	Bush
Clinton	0	-	-
Obama	1	0	-
Bush	**10**	**10**	0

Table 3.9: Relative weights r_{ij} of each inversion with item similarities

	Clinton	Obama	Bush
Clinton	0	-	-
Obama	0.04	0	-
Bush	**0.48**	**0.48**	0

Finally, the resulting Kemeny distances are reported in Tab. 3.10.

Table 3.10: Unweighted and weighted Kemeny distances

Items	d_K	$d_{K,e}$
π_1 vs π_2	2	2
π_1 vs π_3	2	20

The introduction of weights allows us to account for the similarities between politicians. In fact, according to the weighted distance $d_{K,e}$, π_2 resembles π_1 more than π_3 does.

The corresponding rank correlation coefficient, which uses item similarities, is defined as:

$$\tau_{x,e}(\pi_1,\pi_2) = \frac{\sum_{i<j}^{m} p_{ij}\, o_{\pi_1}(y_i,y_j) o_{\pi_2}(y_i,y_j)}{\max d_{K,e}}. \tag{3.11}$$

Note that, Eqs (3.9) and (3.11) can be considered as tweaks of (3.2) and (3.5) obtained by replacing $w_i w_j$ with p_{ij}. Both methods involve applying a penalty to the inversion of each pair of labels, but the penalties are derived differently.

In conclusion, the item similarity method is handy when dealing with multi-level data. In this case, the data matrix contains rankings of politicians (level 1) who belong to

political parties (level 2).

3.3 The choice of weights

The choice of the weights is crucial because it determines the relative weight of each inversion, and thus the Kemeny distance and the corresponding correlation coefficient. In general, there is not a unique optimal solution to cope with this problem. Many times is up to the researcher to assign the weights to express his apriori knowledge on the alternatives, while in other situations, some unequivocal parameters allow distinguishing the important elements from the irrelevant ones. This section shows an intuitive method to assign weights when partial rankings are present.

3.3.1 Frequency-based weights

This method uses a deterministic procedure to assign individual weights. Suppose that the $n \times m$ data matrix contains n incomplete rankings of m elements; in this case, not all the items are ranked by all the judges. Assuming that choosing to rank an item is a proxy of the greater importance that a judge gives to that item (with respect to the items not ranked), the weights w_i can be defined as

$$w_i = 100\frac{T_i}{n} \quad \text{for } i = 1, \ldots, m, \tag{3.12}$$

where T_i stands for "number of judges that assigns a non-zero rank to the i^{th}-element", the weights w_i are rounded down so that $w_i \in \mathbb{N}$.
In other words, the frequency-based method assigns higher weights to items that are included several times in partial rankings of the data matrix. When using this method, in order to observe $\tau_{x,e} > \tau_x$, a particular situation must occur: the ordering of the generic elements i and j (e.g. $i \succ j$), who have the highest inclusion probabilities, must be respected by the vast majority of the incomplete rankings. This usually happens when there is a high agreement between judges assigning the first and the last positions, but there is uncertainty about the middle positions.
Let us consider an example from the Ballon d'Or award voting to clarify the notion of frequency-based weights. The Ballon d'Or is an annual football award presented by France Football[1] that is generally regarded as the most prestigious individual award for football players. The winner of the FIFA Ballon d'Or is annually chosen, in a system based on positional voting, by international journalists, the coaches, and the

[1] https://www.francefootball.fr/

FIFA national teams' captains.

Voters are provided with a shortlist of 23 players from which they could select the three players they deemed to have performed the best in the previous calendar year. That is, each judge returns a partial ranking expressing only the top 3 positions.

Our example will focus on the Ballon d'Or award vote that took place in 2018. The total number of judges was 503, while the players who received at least a vote were: Ronaldo, De Bruyne, Griezmann, Hazard, Kane, Mbappé, Messi, Modric, Salah, and Varane. The judges' preferences are reported in Tab. 3.11.

Table 3.11: Ordering data matrix

	Players		
	1	2	3
π_1	De Bruyne	Ronaldo	Modric
π_2	Ronaldo	Modric	De Bruyne
π_3	Modric	Ronaldo	De Bruyne
...	...	...	...
π_{503}	Modric	Mbappé	Griezmann

The weights of each footballer, computed according to Eq.(3.12), are reported in Tab. 3.12. Table 3.13 compares the item-weighted and unweighted Kemeny distances computed between the first three judges.

Table 3.12: Weighting vector

Weights									
Ronaldo	De Bruyne	Griezmann	Hazard	Kane	Mbappé	Messi	Modric	Salah	Varane
56	14	27	20	3	40	24	79	26	11

Table 3.13: Item weighted Kemeny distances $d_{K,e}$

Items	d_K	$d_{K,e}$
π_1 vs π_2	4	5894
π_2 vs π_3	2	10962

The introduction of frequency-based weights allows distinguishing high-relevant footballers from negligible ones. In particular, according to the weighted distance, $d_{K,e}$, the judge rankings π_2 and π_3 appear to be more different than the couple π_1-π_2, since they disagree on the ordering of the most important footballers (with the highest weights), i.e. Ronaldo and Modric. On the contrary, the unweighted distance d_K, which does not consider the importance of items, regards the couple π_2-π_3 more similar than the couple π_1-π_2.

3.4 Reaching the weigthed consensus ranking

Now, we are interested in defining a weighted version of the consensus ranking following the approach based on a measure of distance/correlation; the median ranking approach (Kemeny, 1959). The weighted correlation coefficient $\tau_{x,e}$ (Eq. (3.5) and Eq.(3.11)) and the weighted Kemeny distance (Eq.(3.2 and Eq.(3.9)) are used to deal with the aggregation of preferences.

Given a $n \times m$ matrix Π, whose l-th row represents the ranking associated with the l-th judge, the purpose is to identify the median ranking $\hat{\pi}$ within the universe of the permutations (with ties) of m elements, S^m, that best represents the average consensus of the subjects involved (i.e. the matrix Π). Considering that there is a one-to-one correspondence between a rank correlation coefficient and a distance, the solution ranking is reached by minimizing the average distance or, equally, maximizing the average rank correlation:

$$\hat{\pi} = \arg\min_{\pi \in S^m} \sum_{l=i}^{n} d_{K,e}(\pi^{(l)}, \pi) \tag{3.13}$$

$$\hat{\pi} = \arg\max_{\pi \in S^m} \sum_{l=i}^{n} \tau_{x,e}(\pi^{(l)}, \pi), \tag{3.14}$$

where S^m is the universe of all rankings with m objects.

As introduced in Chapter 2, Emond and Mason (2002) proposed the BB algorithm to deal with the consensus ranking problem. Amodio et al. (2016) and D'Ambrosio et al. (2015) proposed two accurate algorithms, they called QUICK and FAST, for identifying the median ranking when dealing with weak and partial rankings, in the framework of the Kemeny approach.

In this chapter, the procedure to derive a weighted consensus is based on their work, but $\tau_{x,e}$ and $d_{K,e}$ encode the spread among rankings considering label weights.

Indicating as s_{ij} and $\pi_{ij}^{(l)}$ the scoring matrices for S and the l^{th}row of Π, $l = 1,..,n$, the

problem is:

$$\max \sum_{l=1}^{n} \frac{\sum_{i=1}^{m}\sum_{j=1}^{m} w_i w_j s_{ij} \pi_{ij}^{(l)}}{\sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j} = \max \sum_{i=1}^{m}\sum_{j=1}^{m} s_{ij} c_{ij}^{ew} \tag{3.15}$$

where $c_{ij}^{ew} = \sum_{l=1}^{n} w_i w_j \pi_{ij}^{(l)}$. The score matrix $CI^{ew} = [c_{ij}^{ew}]$ is a modified version of the *Combined Input Matrix* (CI) proposed by Emond and Mason. It results from a summation of each input ranking multiplied by the weight. Defined in this way, it summarizes the information about the input rankings and the weights in a single matrix.
Emond and Mason conceived a branch-and-bound algorithm to maximize the numerator of Eq (3.15) (since the denominator is a fixed quantity depending on the number of items and their weights), by defining an upper limit on the value of that dot product. This limit is given by the sum of the absolute values of the elements of CI^{ew}:

$$V = \sum_{i=1}^{m}\sum_{j=1}^{m} \left| c_{ij}^{ew} \right|. \tag{3.16}$$

Let $Q = \mathbf{1}$ be a vector of ones of size m. Let c_{ij}^{ew} be the $m \times m$ element weighted combined input matrix. By taking into account all the combinations of m objects, each pair of items is evaluated once by considering the two associated cells in CI^{ew}. A moderately accurate first candidate to be the median ranking can be computed as follow:

- If sign $c_{ij} = 1$ and sign $c_{ji} = -1$ then $Q_i = Q_i + 1$;
- If sign $c_{ij} = -1$ and sign $c_{ji} = 1$ then $Q_j = Q_j + 1$;
- If sign $c_{ij} = 1$ and sign $c_{ji} = 1$ then $Q_i = Q_i + 1, Q_j = Q_j + 1$

In this way, we obtain the updated rank vector Q containing the number of times each object is preferred to the others in the pairwise comparisons. This vector is the starting point for the algorithm. The detailed algorithm employing the defined quantities can be found in Amodio et al. (2016) and D'Ambrosio et al. (2015).
Data analysis is performed using our code written in `R` language (available upon request). The proposed BB algorithm has been implemented in `R` environment by suitably modifying the corresponding functions of the ConsRank package (D'Ambrosio, 2021).

3.5 Experimental evaluation

This section aims to show the impact of the element weighting procedure on the consensus ranking. As soon as the weighted version of the QUICK algorithm finds the consensus ranking, a numerical measure of agreement is provided: the weighted correlation coefficient $\tau_{x,e}$. In a consensus problem, the value of the corresponding $\tau_{x,e}$ is crucial because it represents the overall agreement between the estimated consensus $\hat{\pi}$ and the input rankings Π. That is to say, if the consensus ranking's $\tau_{x,e}$ is close to 0, then it's uncorrelated with the input rankings. Therefore there is not a real optimal solution. The interest lies in pointing out how the consensus ranking and the corresponding $\tau_{x,e}$ vary according to the weighting vector w employed.
In order to study the performance of the $\tau_{x,e}$ we will consider two simulation models and two real datasets.

3.5.1 Simulation under model I

In the first simulation study (Model I), ranking data were generated according to a vector of random variables with 5 independent components $X = (X_1, X_2, X_3, X_4, X_5)^T$, each one following a Gaussian distribution $X_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$. The vector of means is $\mu = (\mu_1 = 0.8, \mu_2 = 1.2, \mu_3 = 1.6, \mu_4 = 1.6, \mu_5 = 1.7)$, and the vector of standard deviations is $\sigma = (\sigma_1 = 0.4, \sigma_2 = 0.3, \sigma_3 = 0.6, \sigma_4 = 0.6, \sigma_5 = 0.4)$.
Each judge observes one realization of the random vector X; $x = (x_1, x_2, x_3, x_4, x_5)^T$ and produces his ranking by assigning the first position, i.e. rank 1, to the item that has the lowest value of x and so on. For example, a judge observes the k^{th} realization of X, say $x_k = (1.021, 1.521, 1.474, 2.16, 1.857)$ and assigns the following ranking vector $\pi(x_k) = (1, 3, 2, 5, 4)$.
Since $\mu_1 < \mu_2 < \mu_3 = \mu_4 < \mu_5$, item number 1 will be reasonably placed most of the time in first position while item number 5 in the last one, furthermore having ties is improbable.
The item weighting vectors employed are $w_1 = (1, 1, 1, 1, 1)$, $w_2 = (10, 1, 1, 1, 10)$ and $w_3 = (1, 1, 10, 10, 1)$. Let's remind that w_1 will produce an unweighted version of consensus since it assigns the same weight to each item. In contrast, w_2 assigns higher weights to the external items, and finally w_3 assigns higher weights to the internal items.
According to Model I, we generated 1000 samples of size 100, i.e. $X_{100\times 5}$. For each sample, the consensus ranking and the corresponding $\tau_{x,e}$ are estimated according to each weighting vector.

In Tabs. 3.14, 3.15, 3.16 the relative weights of each generic inversion depending on the weighting vector are reported.

Table 3.14: Relative weights r_{ij} of each generic inversion when using w_1

	Item1	Item2	Item3	Item4	Item5
Item1	0	-	-	-	-
Item2	0.1	0	-	-	-
Item3	0.1	0.1	0	-	-
Item4	0.1	0.1	0.1	0	-
Item5	0.1	0.1	0.1	0.1	0

Table 3.15: Relative weights r_{ij} of each generic inversion when using w_2

	Item1	Item2	Item3	Item4	Item5
Item1	0	-	-	-	-
Item2	0.061	0	-	-	-
Item3	0.061	0.001	0	-	-
Item4	0.061	0.001	0.001	0	-
Item5	**0.613**	0.061	0.061	0.061	0

Table 3.16: Relative weights r_{ij} of each generic inversion when using w_3

	Item1	Item2	Item3	Item4	Item5
Item1	0	-	-	-	-
Item2	0.001	0	-	-	-
Item3	0.061	0.061	0	-	-
Item4	0.061	0.061	**0.613**	0	-
Item5	0.001	0.001	0.061	0.061	0

When equal weights are set, each inversion has the same relative weight determining the $d_{K,e}$ and $\tau_{x,e}$ (Tab. 3.14). That is to say, the mass of weights is evenly distributed. On the contrary, vectors w_2 and w_3 mainly emphasize the inversion of the two most important items attributing the 61.3% of the total weight.

Table 3.17: Distribution of consensus ranking vs weighting vector

Consensus ranking	w_1	w_2	w_3	Total
(1, 2, 3, 4, 5)	456	449	458	1363
(1, 2, 3, 5, 4)	112	121	109	342
(1, 2, 4, 3, 5)	413	405	413	1231
(1, 2, 4, 5, 3)	16	13	17	46
(1, 2, 5, 3, 4)	119	127	114	360
(1, 2, 5, 4, 3)	26	23	27	76
Total	1142	1138	1138	3418

Tab. 3.17 counts how many times the i^{th} candidate is chosen to be the consensus ranking by the QUICK algorithm when using the j^{th} weighting vector. Six candidates have been chosen at least once as consensus. It can be noticed that the weighted QUICK algorithm, regardless of the weighting vector employed, picks as the optimal solution predominantly the candidates $(1,2,4,3,5)$ and $(1,2,3,4,5)$ coherently with the generating model parameters. As one may notice, the algorithm finds more than one optimal solution approximately in 10% of the simulations (total ≈ 1100).
Figure 3.1 compares the conditional distributions of $\tau_{x,e}$ for the three different weighting vectors.

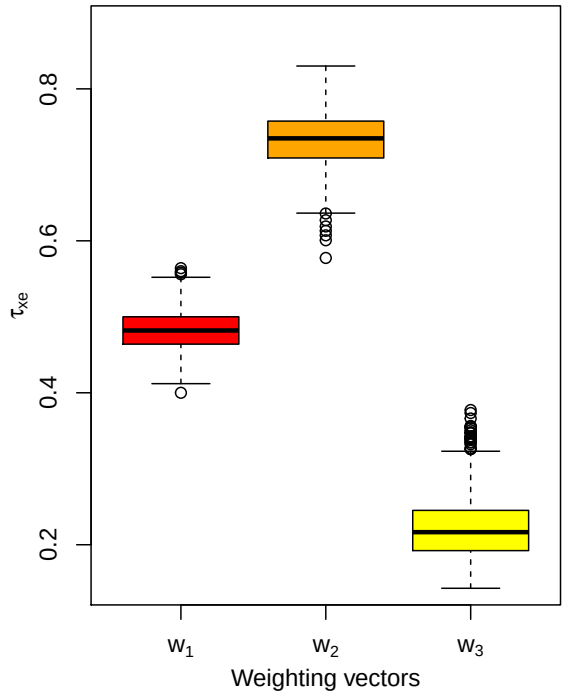


Figure 3.1: Distribution of $\tau_{x,e}$ vs weighting vectors

The conditional distributions of $\tau_{x,e}$ depending on the weighing vectors, are very different. In particular, when using $w_2 = (10,1,1,1,10)$, the corresponding $\tau_{x,e}$ takes high values varying from 0.57 to 0.83 with median and mean approximately equal to 0.73. This happens because the vast majority of the judges prefer item number 1 to item number 5. Thus, there is a strong concordance between them, assigning the ranking of the items with the highest weight. In fact, as pointed out in Tab. 3.15, the inversion of item number 1 with item number 5 has the largest relative weight equal to 61.3%. This implies that if most of the judges do not commit the over-penalized inversion, they will exhibit a firm agreement, as indicated by the $\tau_{x,e}$ of the consensus ranking. That is, the optimal solution is a proper synthesis of the input rankings.
On the contrary, conditioning to $w_3 = (1,1,10,10,1)$, the corresponding $\tau_{x,e}$ takes small values ranging from 0.14 to 0.37, the median is equal to 0.21 and mean equal

to 0.22. Again this is strong evidence of the impact of weights. The weighting vector w_3 brings out the strong disagreement that exists between the judges in the determination of the rank of item number 3 and item number 4. In this case, just over half of the judges prefer item number 3 to item number 4. Therefore, the consensus ranking found is not a proper synthesis of the input rankings.

Such results are due to either the weighting vectors and the weighting aggregation procedure (i.e. product aggregation), mainly emphasizing the inversion of the most important items. If one wants to distribute the mass of weights more evenly, they should decrease the individual weights or use another type of weighting scheme (e.g. arithmetic mean or geometric mean).

3.5.2 Simulation under model II

The second simulation (Model II) is run in order to include ties in the model matrix. Data are again generated according to a vector of random variables with 5 independent components $Y = (Y_1, Y_2, Y_3, Y_4, Y_5)^T$, each one following a Gaussian distribution $Y_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$. The vector of means is $\mu = (\mu_1 = 0.8, \mu_2 = 1.2, \mu_3 = 1.6, \mu_4 = 1.6, \mu_5 = 1.7)$, and the vector of standard deviations is $\sigma' = (\sigma_1 = 0.4, \sigma_2 = 0.3, \sigma_3' = 0.005, \sigma_4' = 0.005, \sigma_5 = 0.4)$. Each judge observes one realization of the random vector Y rounded to the second decimal place $y = (y_1, y_2, y_3, y_4, y_5)^T$ and produces his ranking. The item weighting vectors employed are again $w_1 = (1,1,1,1,1)$, $w_2 = (10,1,1,1,10)$ and $w_3 = (1,1,10,10,1)$.

We generated 1000 samples of size 100, i.e. $Y_{100\times 5}$ according to Model II. For each sample, the weighted QUICK algorithm estimates the consensus ranking and the corresponding $\tau_{x,e}$ according to the weighting vectors. Due to either the rounding of digital places and the choice of small standard deviation of item number 3 and item number 4, many judges will produce ties in their rankings. The results of the simulation are shown in Tab. 3.18 and Figure 3.2.

Table 3.18: Distribution of consensus ranking vs weighting vector

Consensus ranking	w_1	w_2	w_3	Total
(1, 2, 4, 4, 4)	4	4	4	12
(1, 2, 3.5, 3.5, 5)	979	979	980	2938
(1, 2, 4.5, 4.5, 3)	17	17	17	51
Total	1000	1000	1001	3001

Tab. 3.18 shows that the choice of the consensus ranking is unequivocal. Over 97% of the time, consistently with the data generator model, QUICK selects the candidate (1, 2, 3.5, 3.5, 5) as the optimal solution. This is evidence of the goodness of the algorithm's performance.

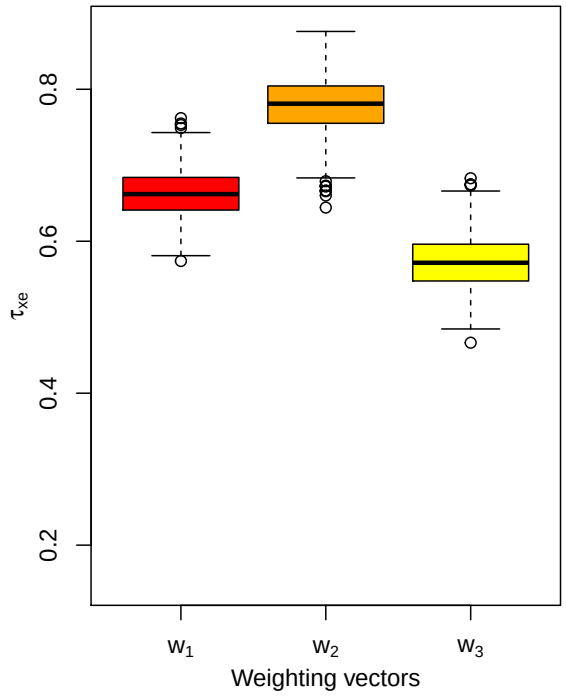


Figure 3.2: Distribution of $\tau_{x,e}$ vs weighting vectors

Once again, the highest agreement between the judges and the consensus ranking is reached using the weighting vector w_2. In fact, the corresponding $\tau_{x,e}$ varies from 0.64 to 0.87 with mean and median equal to 0.78. The lowest agreement is reached with w_3, when the corresponding $\tau_{x,e}$ takes values between 0.47 and 0.68 with median and mean equal to 0.57. The three conditional distributions turn out to be much more similar than they were in the first simulation. This is due to two factors, firstly the standard deviations of item number 3 and item number 4 ($\sigma_3' = \sigma_4' = 0.005$) are much lower than in the first simulation ($\sigma_3 = \sigma_4 = 0.6$). Therefore item number 3 and item number 4

cause less noise, and consequently, their rankings are indeed defined. This is visible in Tab. 3.18, where there is only one real candidate to be the consensus. In other words, there is less uncertainty about the internal items. Secondly, ties are allowed. In this example, item number 3 and item number 4 are equally likeable; therefore, the average agreement among judges will be higher if allowed to express a tie. This is particularly evident in the case of w_3; in the first simulation, the similarity between item number 3 and item number 4 caused strong disagreement between the judges, while in the second simulation, the two factors menage to mediate.

3.5.3 A real data application: the ISTAT dataset

ISTAT[2] dataset concerns the sample survey "Aspetti della vita quotidiana" (aspects of daily life); it provides basic information on the daily lives of individuals and families. Since 2005 it has been conducted annually in February. The information gathered makes it possible to learn about citizens' habits and the problems they face every day. Thematic areas on different social aspects follow each other in the questionnaires, allowing us to understand how individuals live and how satisfied they are with their conditions, their economic situation, the area where they live, the functioning of services, etc. The data matrix dimension is 22×10; the rows are the 20 regions of Italy and the autonomous provinces of Trento and Bolzano, and the columns stand for the problems related to the city, such as parking difficulties (A), inefficiency of public transport (B), traffic (C), poor street lighting (D), poor road conditions (E), dirty roads (F), air pollution (G), noise (H), risk of crime (I), bad smell (L). In the original data X, the x_{ij} cell is the percentage of people in the i^{th} region who feel that their city particularly suffers from the j^{th} problem. We re-arranged the data such that within each row rank 1 is assigned to the problem with the highest percentage, and so on. In other words, there are 22 judges (the regions) expressing their preferences on 10 elements (problems), where the item that is ranked first is the problem that afflicts the region the most.
The aim is to study the influence of the weighting vector on the resulting consensus ranking. Two weighting vectors will be compared; w_1 which assigns the same weight to each element, and w_2 which is based on the item similarity criterion, i.e. swapping similar items should be less penalized than swapping two dissimilar ones.
For this purpose, we found three clusters of items. Cluster number 1 called "Mobility and road conditions" that contains the items: A, B, C, D, E. Cluster number 2, called "Livability" includes: F, G, H, L. Finally, cluster number 3 contains only element I (risk of crime). With w_2, we penalized swapping elements of the same cluster with a

[2] https://www.istat.it/

weight equal to 1 while swapping elements of a different cluster with a weight equal to 50. In this case, the relative weight of each inversion between two generic elements y_i and w_j is defined as follows:

$$r_{ij} = \begin{cases} 0.001 & \text{if } y_i, y_j \text{ belong to the same cluster} \\ 0.034 & \text{if } y_i, y_j \text{ belong to the different clusters} \\ 0 & \text{if } y_i = y_j \end{cases} \tag{3.17}$$

Table 3.19: Relative weight of each inversion

	A	B	C	D	E	F	G	H	I	L
A	0.000	-	-	-	-	-	-	-	-	-
B	0.001	0.000	-	-	-	-	-	-	-	-
C	0.001	0.001	0.000	-	-	-	-	-	-	-
D	0.034	0.034	0.034	0.000	-	-	-	-	-	-
E	0.001	0.001	0.001	0.034	0.000	-	-	-	-	-
F	0.034	0.034	0.034	0.001	0.034	0.000	-	-	-	-
G	0.034	0.034	0.034	0.001	0.034	0.001	0.000	-	-	-
H	0.034	0.034	0.034	0.001	0.034	0.001	0.001	0.000	-	-
I	0.034	0.034	0.034	0.034	0.034	0.034	0.034	0.034	0.000	-
L	0.034	0.034	0.034	0.001	0.034	0.001	0.001	0.001	0.034	0.000

The consensus estimated for each weighting vector is shown in Tab. 3.20

Table 3.20: Consensus ranking for each weighting vectors

	1	2	3	4	5	6	6	8	9	10	$\tau_{x,e}$
w_1	E	A	B	C	D	G	H	F	I	L	**0.69**
w_2	E	A	B	C	D	G	H	F	I	L	**0.78**

The consensus ranking shows that elements of cluster 1 “Mobility and road conditions”, take up the first five positions. In particular, element E (road conditions) worries the citizens the most. The impact of weights is visible. Although the optimal solution remains the same, the value of $\tau_{x,e}$ increases. The value of the correlation coefficient stands for the representativeness of the optimal solution found by the algorithm. In this case, taking into account the element similarities increases the representativeness

of the consensus ranking. The positive variation of $\tau_{x,e}$ reveals that most of the time, the disagreement among the regions' rankings occurs between similar elements, i.e. belonging to the same cluster. Therefore the general weighted agreement, computed with w_2, is higher than the unweighted one computed using w_1.

3.5.4 A real data application: the Quiz dataset

The quiz dataset (Jacques et al., 2014) contains the answers of 70 students (40 of the third year and 30 of the fourth year) from Polytech'Lille (Statistics Engineering School, France) to the four following quizzes: Literature Quiz, Football Quiz, Mathematics Quiz and Cinema Quiz. In this study, the Mathematics Quiz will be analysed; it consists of ranking four numbers according to increasing order: $A = \frac{\pi}{3}$, $B = \log(1)$, $C = e^2$, $D = \frac{1+\sqrt{5}}{2}$.

Each student provides his ranking without using the calculator such that the data matrix has 70 rows and 4 columns. Differently from the previous examples, the exact order of items is known, that is; $\log(1) < \frac{\pi}{3} < \frac{1+\sqrt{5}}{2} < e^2$, i.e. $B < A < D < C$.

The QUICK algorithm allows us to find out the unweighted consensus ranking, $\{\text{B} \succ \text{A} \succ \text{D} \succ \text{C}\}$ with correlation coefficient $\tau_x = 0.85$. Therefore the global solution is the right one. Furthermore, the degree of concordance between students is high.

Now we assume that the students had no difficulty in realising that the elements B ($log(1)$) and C (e^2) had to be placed in the first and last position respectively, and maybe this "easy choice" let the correlation coefficient grows. Therefore we want to test whether the students were good enough to recognise the exact order of elements A ($\frac{\pi}{3}$) and D ($\frac{1+\sqrt{5}}{2}$). A way of doing that is to define a vector of weights $w = (10,1,1,10)$ that emphasises the inversion between A and D and then to compute the weighted consensus ranking and the corresponding correlation coefficient $\tau_{x,e}$. The relative weight of each inversion is reported in table 3.21.

Table 3.21: Relative weights r_{ij} of each generic inversion

	A	B	C	D
A	0.000	-	-	-
B	0.071	0.000	-	-
C	0.071	0.007	0.000	-
D	**0.709**	0.071	0.071	0.000

The weighted consensus ranking is $\{\text{B} \succ \text{A} \succ \text{D} \succ \text{C}\}$ and the corresponding $\tau_{x,e}$ is

0.72. What does it mean? Although decreased, the value of $\tau_{x,e}$ is still quite high, indicating that the unweighted consensus was robust and not mainly influenced by the "easy choice". At the same time, items A and D are indeed the most difficult values to rank: $\tau_{x,e}$ assumes its minimum value, 0.72, with a vector of weights $w = (10,1,1,10)$ and its maximum value, 0.90, with a vector of weights $w = (1,10,10,1)$ (items B and C are the easiest to rank). In this way, $\tau_{x,e}$ can also be helpful to verify where (i.e. referring to which items) the disagreement between rankings mainly occurs.

3.6 Concluding remarks

Within the framework of preference data, where individuals express their preferences over a set of items, the main interest lies in evaluating the agreement between them and obtaining a synthesis of their preferences by computing a consensus ranking. Different approaches have been proposed in the literature to cope with this problem, but the most popular one is probably related to distances/correlations. Usually, these are not sensitive to the importance of items since each inversion is considered equally important. In many cases, this assumption could be simplistic. For this reason, in this chapter, we provided an element weighted rank correlation coefficient $\tau_{x,e}$ for linear, weak and incomplete orderings. We demonstrated the correspondence between $\tau_{x,e}$ and the corresponding weighted Kemeny distance $d_{K,e}$. Finally, we showed that in the case of equal weights for all items $w_i = C$, the weighted rank distance $d_{K,e}$ is proportional to the well-known Kemeny distance d_K, while the correlation coefficient $\tau_{x,e}$ is equal to the Emond and Mason's τ_x. From the simulation study and the real data examples, we demonstrated that the BB algorithm allows us to find the true consensus and to show how the weighting vector affects the representativity of the median ranking. The weighted consensus algorithm's computational effort was investigated by considering some simulations, shown in Appendix A.1. We progressively increased the sample size (from 200 to 1000) and the number of items (from 3 to 10). Compared with the unweighted algorithm, the weighted consensus algorithm entails a slight increase in computational time which has never exceeded 30%.

Future studies could further explore this issue by including the element weighting procedure in a cluster analysis of ranking data. When dealing with preference data, cluster analysis attempts to identify homogeneous groups of rank choices (clusters). Using weights allows for considering the importance of alternatives minimizing the distances between cluster members.

Chapter 4

A weighted distance-based approach with boosted decision trees for Label Ranking

4.1 Introduction

Preference data commonly arises when n judges (raters, voters or experts) order m different items (labels, alternatives or elements) from the most to the least preferred. In many real-world cases, the ranking responses are paired with additional features which characterize the judges, e.g., socioeconomic and socio-demographic characteristics. In these cases, the goal is learning a function that predicts the ranking responses of new instances, and determining how covariates affect the response rankings.

In the literature, especially in the computer science community, the same issue is frequently referred to as Label Ranking (LR). Specifically, Label Ranking is defined as a non-standard supervised classification problem that aims at learning a mapping from instances (or judges) to rankings over a finite set of predefined elements (labels) (Zhou et al., 2014). In this framework, instances are therefore defined by a set of features, or, equivalently, considering the statistical community, judges paired with a set of independent variables. Throughout this chapter, we will use interchangeably the terms *judges*, *raters* and *instances* to refer to the set of preferences possibly paired with independent variables.

LR can be thought of as a variant of the standard classification problem since it requires

ranking all labels for each instance rather than assigning a single response label. The set of models aiming to study individual or collective decision processes and procedures to predict a preference relation on a set of elements is called *Preference Learning* (Fürnkranz and Hüllermeier, 2010). In this context, LR can be considered as a branch of preference learning along with other methodologies such as Recommender Systems (Cohen et al., 1999) and Learning to Rank (Liu, 2011).

Many methods were proposed in the last decade to tackle the LR problem, such as decomposition approaches (Dekel et al., 2003; Har-Peled et al., 2003; Hüllermeier et al., 2008), probabilistic approaches (Cheng and Hüllermeier, 2008; Cheng et al., 2010; Grbovic et al., 2012; Rodrigo et al., 2021), but the tree-based approaches (Cheng et al., 2009) have become the most popular techniques in the last years due to their ease of interpretation (Heiser and D'Ambrosio, 2013; Aledo et al., 2017; de Sá et al., 2017; Werbin-Ofir et al., 2019; Dery and Shmueli, 2020; Plaia et al., 2021a). Decision trees are typically fast to train and relatively easy to interpret since the model can be pictured as a tree structure. Still, unfortunately, as stated by the authors of the leading decision tree learning books (Breiman et al., 1984), they are *unstable* suffering from high variance. The decision trees learned from different data sub-samples may be quite different. Decision trees are often referred to as *weak learners*; indeed they are often combined to build a *strong learner*.

The best-performing ensemble methods in the LR literature are bagging (Aledo et al., 2017; Plaia et al., 2021a), random forest (de Sá et al., 2017; Zhou and Qiu, 2018) and boosting (Dery and Shmueli, 2020; Plaia et al., 2021a). The main idea is to improve the precision of the decision trees by perturbing the training set, using bootstrapping, and then combining the results of several decision trees into a single predictor. These procedures belong to the class of methods called Perturb and Combine (Breiman, 1996).

In order to fit ensemble methods into the LR framework, the characterization of distance and correlation measures suitable for ranking data is needed. Distance and correlation measures for rankings are then used to i) construct a measure of impurity in the tree splitting process; ii) specify an aggregation method for the preferences in order to identify the *consensus ranking* (defined as the best representative ranking of the whole set of preferences); iii) introduce a loss function to assess the predictive performance of the proposed label ranker.

Although tree-based approaches in the literature differ in several aspects, they all rely on unweighted distance and correlation measures. Therefore, these measures are neither sensitive to the importance nor to the similarity of labels. Nevertheless, in many settings, failing to predict the ranking position of a highly relevant label should be considered more serious than failing to predict a negligible one. Moreover, an efficient

classifier should be able to take into account the similarity between the elements to be ranked. In such situations, a weighted distance is needed to deal with the similarity of labels (i.e. politicians). Similarly, it would be desirable to introduce weighted distances in label ranking algorithms to take into account additional information on the labels of a ranking. The features, properties and importance of weighted ranking distances are widely discussed in the literature (Kumar and Vassilvitskii 2010; García-Lapresta and Pérez-Román 2010; Can 2014; Plaia et al. 2018, 2019; Albano and Plaia 2021), but no effort has been made to incorporate them into label ranking algorithms. This chapter proposes a new item-weighted version of the boosting ensemble algorithm for the label ranking task.

To do this, we extend the work of Plaia et al. (2021a), where they proposed a boosting algorithm for label ranking, by introducing the item-weighted Kemeny distance (Albano and Plaia, 2021) as a measure of impurity in the splitting process and its related rank item-weighted correlation coefficient for identifying the consensus ranking in the final nodes.

We evaluate the performance of the proposed method on real data and assess its robustness to outliers on simulated data with increasing noise levels and different weighting structures.

The chapter is organized as follows: Section 4.2 discusses the importance of ensemble methods for improving tree prediction performance. Section 4.3 presents our main proposal, which is three item-weighted boosting algorithms to deal with label ranking, and the steps for its implementation in the `R` statistical software environment are described. Finally, in Section 4.4, the procedures are compared through application to both real and simulated datasets. Conclusions conclude the chapter.

4.2 Decision trees and boosting methods

Breiman et al. (1984) developed Classification and Regression Trees (CART) as an alternative non-parametric approach to classification and parametric regression procedures. Decision trees perform hierarchical partitions of the feature space $\mathscr{X}$ into a set of T rectangular, non-overlapping regions $R_1, \ldots, R_T$ to predict the response value of any instance $x \in \mathscr{X}$. Each leaf defines a region of $\mathscr{X}$ formed by the set of instances corresponding precisely to the same node responses and, thus, the same traversal of the tree in such a way that all observations belong to exactly one region. Decision tree ensembles were initially designed for classification tasks and applied to regression problems shortly after. Besides, some profitable attempts have been made to extend

them to ranking/preference data in the last decade.

Boosting, one of the best-known Perturb and Combine methods, originated from the question posed by Kearns and Valiant (1994) of whether a set of weak classifiers could be converted into a robust classifier. Freund and Schapire (1996, 1997) designed AdaBoost (which stands for adaptive boosting), an ensemble algorithm aiming to drive the training set error rapidly to zero. The key idea consists of repeatedly using the base weak learning algorithm on differently weighted versions of the training data, yielding a sequence of weak classifiers that are finally combined (Galar et al., 2011). That is, starting with the same probability to pick up each instance in the sample, $p(l) = \frac{1}{N}$, form the first training set $T^{(1)}$ by iteratively resampling from T. The sequence of classifiers and training sets is built, and $p(l)$ is increased for those cases that have been most frequently misclassified. At termination, classifiers are combined by a weighted or simple voting. Breiman (1998) refer to algorithms of this type as "adaptive resampling and combining," or "arcing" algorithms.

Boosting is regarded as "one of the most powerful learning ideas introduced in the last twenty years" (Hastie et al., 2009), and remains one of the most widely used and studied ensemble methods with applications in numerous fields. Probably the most severe disadvantage of AdaBoost is that it can be very susceptible to noise, even with regularization, at least on artificially constructed datasets (Schapire, 2013). Mohri et al. (2018) pointed out that in the presence of noise, the distribution weight assigned to examples that are harder to classify substantially increases with the number of rounds of boosting; these examples end up dominating the selection of the base classifiers. They also stated that "empirical results suggest, however, that the performance of AdaBoost tends to degrade more than that of other algorithms for this uniform noise model".

Variations of AdaBoost were developed for multi-class problems (Freund and Schapire, 1997), multi-label problems (Schapire and Singer, 2000), regression problems (Drucker, 1997; Solomatine and Shrestha, 2004), learning to rank problems (Cohen et al., 1999; Xu and Li, 2007; Wu et al., 2010) and finally, to label ranking (Dery and Shmueli, 2020; Plaia et al., 2021a).

4.3 Building an item-weighted tree ensemble for label ranking

This section aims to outline our main proposal, which is a variation of AdaBoost based on the item-weighted Kemeny distance $d_{K,e}$ (Eqs. 3.2-3.9) to perform the LR task.

The proposed approach aggregates the predictions of several weighted distance-based trees in order to obtain a strong learner. Section 4.3.1 defines the decision tree modelling splitting criteria used to build each tree, while Section 4.3.2 describes the proposed ensemble method. Furthermore, covariates are considered to explain individual differences in evaluating choice alternatives. For this reason, defining how much each covariate contributes to identifying clusters of homogeneous respondents is a crucial issue to be addressed.

4.3.1 Weighted distance-based trees: splitting and labelling criteria

To extend classical univariate classification trees to deal with rankings (a vector of multiple responses), the definition of the partitioning metric itself must be generalized. In this work, following Sciandra et al. (2017) and Plaia and Sciandra (2019), in order to avoid the problem related to the multivariate nature of the ranking vector, each vector of preferences will be considered as a unique multivariate entity, i.e., a *categorical tag* is assigned to each distinct ranking. For example, when dealing with the universe of weak orderings of 4 items the tagging strategy employs 75 different categorical tags, Tab. 4.1.

Table 4.1: Tagging strategy applied to the universe of permutations (with ties) of 4 items

Ranking	Tag
$\pi_1 = \{1,1,1,1\}$	1
$\pi_2 = \{1,1,1,2\}$	2
...	...
$\pi_{75} = \{4,3,2,1\}$	75

Note that, when the dissimilarity between two categorical tags needs to be computed, we resort to the item-weighted distance 3.2, 3.9 between the corresponding rankings. In other words, the categorical tags act as identifiers:

$$d(Tag_1, Tag_{75}) \equiv d_{K,e}\Big((1,1,1,1),(4,3,2,1)\Big).$$

The standard recursive binary partitioning process, used by the CART methodology, is applied to build a classification tree for preference rankings, where the tagging strategy is used both in the splitting and labelling phases.
The recursive binary partitioning process is a top-down algorithm. The root node,

containing all observations, is split into a nested sequence of subtrees:

$$T_m = \{\text{root} - \text{node}\} \subset \cdots \subset T_0 = \{\text{full} - \text{tree}\}.$$

The partition of data follows a splitting criterion which consists of maximizing the reduction in the impurity, $\Delta i(s,t)$, resulting from the split s in node t:

$$\Delta i(s,t) = i(t) - p_L i(t_L) - p_R i(t_R),$$

where p_L and p_R are the proportions of units in node t assigned to the left child node t_L and to the right child node t_R respectively at the s-th split. The child nodes will then be further split recursively. All nodes that cannot be further split are called terminal nodes or leaves, and the others are called internal nodes.

A decrease in node impurity at each step must be evaluated considering all covariates and their potential split points. The problem has already been addressed in the literature; Piccarreta (2010) proposes an impurity function based on dissimilarities, but the proposed measures cannot be used to handle preference data. Impurity measures properly suited for rankings are needed. The novelty of our work is to use the item-weighted Kemeny distance (Albano and Plaia, 2021) as an impurity function $i(t)$ in a node t:

$$i(t) = \sum_{\substack{i,j \in t \\ i \neq j}} d(Tag_i, Tag_j) = \sum_{\substack{i,j \in t \\ i \neq j}} d_{K,e}(\pi_i, \pi_j), \tag{4.1}$$

where $d_{K,e}$ is the item-weighted distance measure between the rankings belonging to the same node t via the set of predictors X. Note that, the formula to compute $d_{K,e}$ depends on the weighting scheme chosen. Specifically, Eq. (3.2) is used when weights are assigned to labels individually, while Eq. (3.9) is employed for weights based on the similarity of labels.

An exhaustive search algorithm is used to determine the best splitting rule that gives the greatest reduction of impurity, i.e. the smallest sum of weighted distances of the two child nodes.

To conclude the process, a class response or class ranking is assigned to each terminal node during the labelling phase. A response tag represents the predicted value for all the instances within the same node. In this work, following the median ranking approach described in Section 3.4, we identify the resulting class response as the tag associated with the corresponding consensus ranking in a leaf.

Given N_t rankings in a terminal node t, the consensus ranking $\hat{\pi}_t$ is the solution of the

minimization of Eq.(3.13), the sum being extended to all the rankings $\pi_l \in t$ in the leaf:

$$\hat{\pi}_t = \arg\min_{\pi_t \in S^m} \sum_{l=1}^{N_t} d_{K,e}(\pi_l, \pi_t).$$

In this way, the final outcome of the classification tree is a categorical tag, Tag_t, which identifies a rank vector, $\hat{\pi}_t$ (a vector of multiple responses).

4.3.2 Item-weighted boosting algorithm

While developing a new weighted label ranking algorithm, much effort should be devoted to defining a noise-robust procedure. To this aim, we introduce a first version of the LR algorithm that reproduces the one proposed by Plaia et al. (2021a) by replacing the rank correlation coefficient τ_x with the item-weighted rank correlation coefficient $\tau_{x,e}$ and using the impurity function defined in Eq.4.1 (AdaBoost.R.M1 (1)). Secondly, we propose two tweaks (AdaBoost.R.M2 (2) and AdaBoost.R.M3 (3)), both aiming to reduce the influence of label noise and improve predictive performance.
The proposed algorithms combine classifiers, iteratively created from weighted versions of the learning sample, with weights adaptively modified, iteration by iteration, so that previously misclassified rankings have a higher probability of being sampled in subsequent iterations. The final predicted rankings are computed by a weighted combination of the intermediate rankings of the iterative process. In the following, two vectors of weights will be used:

- **the vector of working weights $\mathbf{p}_b$**: updated at each b iteration of the algorithm. More specifically, $\mathbf{p}_b$ represents the probability of each instance being included in the bootstrap sample;
- **the vector of label weights $\mathbf{w}$**: representing the importance of each item in the ranking (as defined in section 3.4). The importance of each item stays constant during the procedure. If the item-weighting scheme follows the item similarity criterion, the vector of weights $\mathbf{w}$ is replaced by the penalization matrix $\mathbf{P}$; thus $w_i \cdot w_j$ is replaced by p_{ij}.

AdaBoost.R.M1

The first weighted LR algorithm AdaBoost.R.M1 is based on AdaBoost.R (Plaia et al., 2021a), opportunely adapted to item-weighted ranking data (Algorithm 1).

Algorithm 1 AdaBoost.R.M1 - Item-weighted boosting for ranking data

Input: A training set T, a number of iterations B, a vector of weights $\mathbf{w}$
Output: a ranker $C_f(.)$ that maps a given $\mathbf{x}$ to a ranking of the labels

1: initialize $p_b(l) = 1/n \; \forall l = 1,2,...n$
2: **for** $b \leftarrow 1$ to B **do**
3: take a sample T_b, drawn from the training set T using weights $p_b(l)$
4: fit a ranking tree $C_b(.)$ to T_b
5: $e_b = \sum_{l \in T_b} p_b(l) \left(1 - \frac{\tau_{x,e}(l)+1}{2}\right)$ where $\tau_{x,e}(l) = \tau_{x,e}(C_b(x_l), \pi_l)$
6: $\nu_b = \frac{1}{2}\ln\big((1-e_b)/e_b\big)$
7: update the weights $p_{b+1}(l) = p_b(l)\exp\left(\nu_b\left(1 - \frac{\tau_{x,e}(l)+1}{2}\right)\right)$ and normalize them
8: **end for**
9: $C_f(x_l) = \arg\max_{\pi_l \in S^m} \sum_{b=1}^{B} \nu_b \tau_{x,e}(C_b(x_l), \pi_l))$

The algorithm requires as input a vector of weights $\mathbf{w}$ representing label importance and to fix the number of iterations B. The first step consists of initializing the working weights $p_b(l) = 1/n$, which are assigned to each observation in the training set T of size n.

At each iteration b, a tree $C_b(.)$ is trained on T_b leading to a predicted ranking for each instance $\tilde{\pi}_l^b = C_b(x_l)$ (steps 3 and 4).

The ranking error e_b of the ranking tree $C_b(.)$ is estimated employing the item-weighted correlation coefficient $\tau_{x,e}$ between each predicted ranking $\tilde{\pi}_l^b$ and its real value π_l (step 5) given the vector of weights $\mathbf{w}$:

$$e_b = \sum_{l=1}^{n} p_b(l)\left(1 - \frac{\tau_{x,e}(\tilde{\pi}_l^b, \pi_l)+1}{2}\right). \tag{4.2}$$

Since $\left(1 - \frac{\tau_{x,e}(\tilde{\pi}_l^b,\pi_l)+1}{2}\right) \in [0,1]$ and $\sum_{l=1}^{n} p_b(l) = 1$, then e_b variable is a convex combination, such that $e_b \in [0,1]$.

If the prediction $\tilde{\pi}_i^b$ and the observed value π_i are in full disagreement then $\tau_{x,e} = -1$ and $1 - (\tau_{x,e}+1)/2 = 1$.

Step 6 consist in computing a factor ν_b, as a function of e_b, for updating the weights $p_b(l)$.

$$\nu_b = \frac{1}{2}\ln\big((1-e_b)/e_b\big). \tag{4.3}$$

The ν_b value can be interpreted as the specific iteration model weight, derived as a function of the error in each iteration. The lower the error e_b, the higher the weight ν_b.

Moreover, this value is also used in the final decision rule, giving more importance to the trees that made a lower error.
Finally, the weights $p_b(l)$ are updated through a multiplier, and normalized after each iteration (step 7).

$$p_{b+1}(l) = p_b(l) \exp\left(\nu_b \left(1 - \frac{\tau_{x,e}(l)+1}{2}\right)\right). \tag{4.4}$$

The way the working weights are updated ensures that the procedure focuses more on hard-to-predict instances. Formally, the lower the item-weighted correlation coefficient between the predicted and the observed ranking, the higher the probability that this observation is resampled in the new iteration. The iterative procedure continues until a stopping criterion (i.e., $\nu_b \geq 0.5$) or the maximum number of trees is reached.
The final prediction (step 9) comes from the last step of the procedure. The item-weighted boosting uses rank aggregation (D'Ambrosio et al. 2015; Amodio et al. 2016; D'Ambrosio 2021) to combine the predictions of each individual tree.
The aggregated ranking for a generic l-th observation at the b-th iteration is

$$\hat{\pi}_{lb} = \underset{\pi_l \in S^m}{\arg\max} \sum_{k=1}^{b} \nu_b \tau_{x,e}(\tilde{\pi}_l^{\,k}, \pi_l), \text{ with } b = 1, 2, \ldots, B, \tag{4.5}$$

where the factor ν_b, is the weight related to the b-th tree. In other words, the trees providing better estimates will receive more voting power in the final prediction.
Once each unit has been assigned a predicted ranking tree by tree, the aggregated error at the b iteration is

$$err(b) = 1 - \frac{\tau_{x,e}(b)+1}{2}, \tag{4.6}$$

where $\tau_{x,e}(b) = \frac{1}{n}\sum_{l=1}^{n} \tau_{x,e}(\hat{\pi}_{lb}, \pi_l)$ is the average of $\tau_{x,e}$ of the b-th tree over all the units in the training set T. Then, we define a "predictor matrix" (Tab. 4.2) that summarizes all predictors. Furthermore, the procedure allows determining the overall importance of covariates by averaging over their importance, resulting in each of the b trees, with weights ν_b.
Once the structure of the first version of weighted boosting has been defined, we define two tweaks algorithms (AdaBoost.R.M2 and AdaBoost.R.M3) intending to improve the predictive performance of the procedure. As stated earlier, outliers in the training set are hard-to-predict observations. Therefore, the boosting procedure may focus too much on these observations and give them a very high weight by increasing the number of trees. As a result, the overall performance of the algorithm can be worsened in some

Table 4.2: Predictor matrix structure

Weights	ν_1	ν_2	...	ν_b	...	ν_B
Trees	$C_1(.)$	$C_2(.)$	...	$C_b(.)$	...	$C_B(.)$
1	$\hat{\pi}_{11}$	$\hat{\pi}_{12}$	...	$\hat{\pi}_{1b}$	...	$\hat{\pi}_{1B}$
2	$\hat{\pi}_{21}$	$\hat{\pi}_{23}$	...	$\hat{\pi}_{2b}$	...	$\hat{\pi}_{2B}$
.	.	.	...	.	...	.
.	.	.	...	.	...	.
.	.	.	...	.	...	.
n	$\hat{\pi}_{n1}$	$\hat{\pi}_{n2}$	...	$\hat{\pi}_{nb}$	...	$\hat{\pi}_{nB}$
Error	$err(1)$	$err(2)$	...	$err(b)$	...	$err(B)$

scenarios.

AdaBoost.R.M2

Adaboost.R.M2 stems from the idea of modifying the loss function so that the working weights $\mathbf{p}_b$ are updated following a "less aggressive" function. In other words, a function that is less sensitive to outliers. In fact, step 7 of the algorithm shows that weights are updated through a multiplier, such as $p_{b+1} = p_b \cdot M_1$, where the multiplier is defined as

$$M_1 = \exp\left(\nu_b\left(1 - \frac{\tau_{x,e}(l)+1}{2}\right)\right).$$

However, the exponential increase in the error could result in excessive weight being assigned to outliers. An alternative (see Schapire and Freund 2013) is to use a binary logistic function such as

$$M_2 = \log_2(1+\exp\left(\nu_b\left(1 - \frac{\tau_{x,e}(l)+1}{2}\right)\right)).$$

Fig. 4.1 compares M_1 and M_2 as a function of $\tau_{x,e}$. Indeed, it is clear that the binary logistic multiplier M_2 is upper bounded by exponential multiplier M_1, for each value of $\tau_{x,e}$.

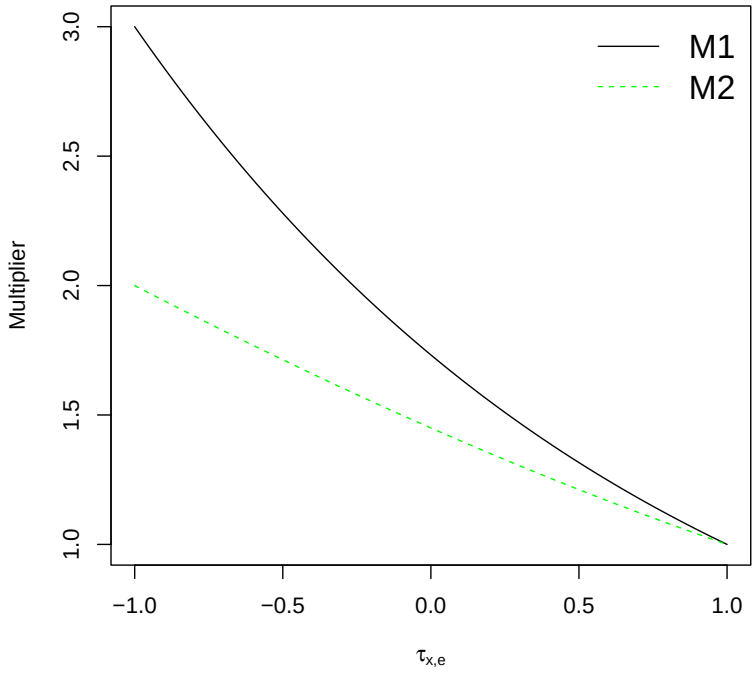


Figure 4.1: Exponential multiplier M_1 and binary logistic multiplier M_2, vs the ranking correlation coefficient $\tau_{x,e}$, with $e_b = 0.1$.

Adaboost.R.M2 is detailed in the algorithm 2, that compared with AdaBoost.R.M1 only modifies step 7. The function to update the weights is now $p_{b+1}(l) = p_b(l)\log_2(1+\exp\left(v_b\left(1-\frac{\tau_{x,e}(l)+1}{2}\right)\right))$.

Algorithm 2 AdaBoost.R.M2 - Item-weighted boosting for ranking data

Input: A training set T, a number of iterations B, a vector of weights w
Output: a ranker $C_f(.)$ that maps a given x to a ranking of the labels

1: initialize $p_b(l) = 1/n \; \forall l = 1, 2, ...n$
2: **for** $b \leftarrow 1$ to B **do**
3: take a sample T_b, drawn from the training set T using weights $p_b(l)$
4: fit a ranking tree $C_b(.)$ to T_b
5: $e_b = \sum_{l \in T_b} p_b(l) \left(1 - \frac{\tau_{x,e}(l)+1}{2}\right)$ where $\tau_{x,e}(l) = \tau_{x,e}(C_b(x_l), \pi_l)$
6: $v_b = \frac{1}{2} \ln\big((1-e_b)/e_b\big)$
7: update the weights $p_{b+1}(l) = p_b(l) \log_2(1 + \exp\left(v_b \left(1 - \frac{\tau_{x,e}(l)+1}{2}\right)\right)$ and normalize them
8: **end for**
9: $C_f(x_l) = \arg\max_{\pi_l \in S^m} \sum_{b=1}^{B} v_b \tau_{x,e}(C_b(x_l), \pi_l))$

AdaBoost.R.M3

The key idea of AdaBoost.R.M3 is to let the outliers dominate the growth of the last trees but to strongly penalise the models with a high error rate in the final prediction phase. For instance, Tab. 4.3 reports a demonstrative example in which the procedure is run for ten trees, and the last ones are dominated by outliers; thus, they have low predictive power on the test set. Then, giving each model a weight of v_b^2 rather than v_b allows penalising trees dominated by outliers by reducing their voting power on the final prediction.
Tab. 4.3 shows the model weights $\left(v, v^2\right)$ and the corresponding relative weights, $\left(v_b/(\sum_{k=1}^{B} v_k), v_b^2/(\sum_{k=1}^{B} v_k^2)\right)$ $b = 1, \ldots, B$, of the demonstrative example.

Tree	*Weight*		*Relative weight*	
	v_b	v_b^2	$v_b/(\sum_{k=1}^{B} v_k)$	$v_b^2/(\sum_{k=1}^{B} v_k^2)$
1	0.45	0.20	0.09	0.06
2	0.60	0.36	0.11	0.11
3	0.70	0.49	0.13	0.14
4	0.70	0.49	0.13	0.14
5	0.95	0.90	**0.18**	**0.26**
6	0.77	0.59	0.15	0.17
7	0.50	0.25	0.09	0.07
8	0.25	0.06	**0.05**	**0.02**
9	0.20	0.04	**0.04**	**0.01**
10	0.15	0.02	**0.03**	**0.01**

Table 4.3: Model weights and relative model weights of a demonstrative example.

It is clear that the effect of the last three bags on the final prediction has been considerably turned down. In fact, their cumulative relative weight approximately reduced from $0.12\,(0.05+0.04+0.03)$ to $0.04\,(0.02+0.01+0.01)$. In contrast, the relative weight of the fifth tree has significantly increased from 0.18 to 0.26.
Adaboost.R.M3 is detailed in the algorithm 3, that compared with AdaBoost.R.M1 only modifies step 9, where the final prediction is obtained.

Algorithm 3 AdaBoost.R.M3 - Item-weighted boosting for ranking data

Input: A training set T, a number of iterations B, a vector of weights w
Output: a ranker $C_f(.)$ that maps a given x to a ranking of the labels

1: initialize $p_b(l) = 1/n \;\forall l = 1,2,...n$
2: **for** $b \leftarrow 1$ to B **do**
3: take a sample T_b, drawn from the training set T using weights $p_b(l)$
4: fit a ranking tree $C_b(.)$ to T_b
5: $e_b = \sum_{i \in T_b} p_b(l)\left(1 - \frac{\tau_{x,e}(l)+1}{2}\right)$ where $\tau_{x,e}(l) = \tau_{x,e}(C_b(x_l), \pi_l)$
6: $v_b = \frac{1}{2}\ln\big((1-e_b)/e_b\big)$
7: update the weights $p_{b+1}(l) = p_b(l)\exp\left(v_b\left(1 - \frac{\tau_{x,e}(l)+1}{2}\right)\right)$ and normalize them
8: **end for**
9: $C_f(x_l) = \arg\max_{\pi_l \in S^m} \sum_{b=1}^{B} v_b^2 \tau_{x,e}(C_b(x_l), \pi_l))$

4.4 Experimental evaluation

This section aims to show the impact of label weights on the LR boosting algorithm. The main interest lies in pointing out how the prediction error varies according to the weighting scheme employed. In order to compare the performance of the three proposed methods, twelve simulated and three real datasets will be considered.

The methods are experimentally evaluated through a five-fold cross validation procedure. That is, each dataset was randomly partitioned into five separate folds. Four folds were used as the training set in each branch, and the last fold (a different one for each branch) was used as the test set. For each training set, the proposed boosting procedure with $B = 50$ iterations is run, and at the end, the fifty predictions are aggregated, as shown in step 9 of each algorithm. To compute the difference between the final predicted ranking and the real ranking, a linear transformation of the weighted ranking correlation coefficient $err(b) = \left(1 - \frac{\tau_{x,e}(b)+1}{2}\right)$ was utilized for each test instance. Then, the average over all test instances in the five test folds was computed to give a final measure for each dataset.

We would like to emphasise that simulations play a fundamental role in assessing the performance of the proposed methods. In fact, simulations make it possible to set the level of label noise a priori and keep other noise parameters under control. Indeed, one of our interests is to develop an item-weighted LR algorithm that is robust to label noise; thus, we will study how the performance of the proposed methods varies at different noise levels. Often, in the literature, the experimental evaluation is performed on several real datasets (Aledo et al., 2017; de Sá et al., 2017; Werbin-Ofir et al., 2019; Dery and Shmueli, 2020). Actually, even if the number of datasets considered is large, the true data generating process is unknown. That is, it is hard to verify why, for a particular dataset, one method outperforms the others. On the other hand, simulated data come from a controlled environment, which allows one to set specific parameter settings (e.g. level of heterogeneity) and verify in which conditions a method is better than the others.

To the best of our knowledge, we are the first to propose a boosting algorithm for the label ranking task designed to include label weights. For this reason, a comparison with state-of-the-art unweighted LR algorithms can be carried out only under the assumption of indifference among alternatives, i.e. unitary weights $w_1 = w_2 = \cdots = w_m = 1$ (Section 4.4.1, Fig. 4.3). Indeed, the prediction errors of state-of-the-art LR algorithms are computed using unweighted distances, whereas our methods are evaluated through item-weighted distances. Clearly, a weighted distance is not comparable with an unweighted distance, so a comparison with state-of-the-art algorithms appears in-

consistent when weights are not unitary. Furthermore, the final aim is not to verify that the introduction of weights produces lower errors, but the actual novelty is the introduction of a flexible procedure that allows inhomogeneous importance to the items.
As regards the computing time, the rank aggregation step (step 9) in the procedure entails a slight increase in average computation time, which does not exceed 30%, with respect to QUICK and FAST's ones (a graphical comparison, considering 3 to 10 items, can be found in the Appendix, fig. A.1). The computational efficiency of the QUICK and FAST algorithms is widely described in D'Ambrosio et al. (2015); Amodio et al. (2016) while the computational efficiency of their weighted versions in Albano and Plaia (2021).
Data analysis is performed using our code written in `R` language (available upon request) by opportunely modifying available functions in the `R` packages `ConsRank` (D'Ambrosio, 2021), `rpart` (Therneau et al., 2015) and `adabag` (Alfaro et al., 2013).

4.4.1 Simulation study

Following D'Ambrosio and Heiser (2016) and Plaia et al. (2021a), we considered a predictor space (X_1 and X_2), with $X_1 \sim U(0,10)$ and $X_2 \sim U(0,6)$, which was partitioned into five regions as shown in Figure 4.2.

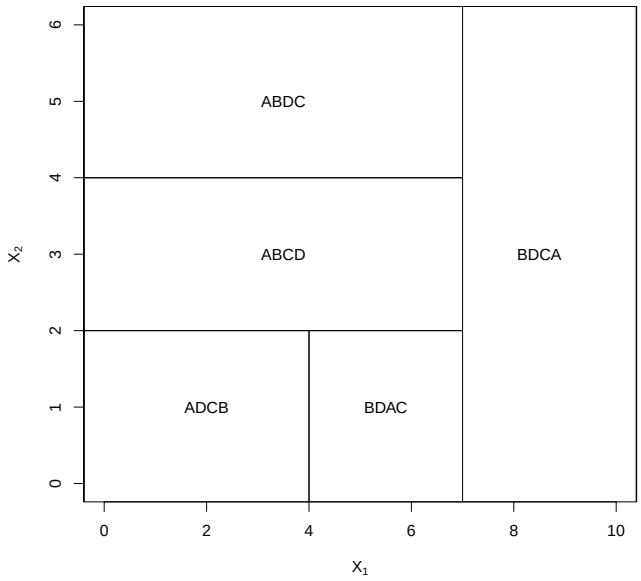


Figure 4.2: Theoretical partition of the predictor space (X_1, X_2) with 4 items

The number of datapoints, i.e. instances, (of 4 items) within each sub-partition, was determined by: i) randomly drawing from a normal distribution $\mathcal{N}(0,10)$, ii) dividing them by their summation, iii) multiplying by the true sample size (N = 200; N = 500). The rankings (datapoints) within each sub-partition were generated from a Mallows Model (Mallows, 1957), one of the first probability models proposed for rankings, frequently used in both theoretical and applied studies. It is an exponential model defined by a central permutation π_0 and a dispersion parameter θ.

When $\theta > 0$, π_0 represents the mode of the distribution, i.e., the permutation with the highest probability to be generated. The π_0 values for our simulation studies are shown in Figure 4.2. The probability of any other ranking decays exponentially with increasing distance to the central permutation. The dispersion parameter controls the steepness of this decline. Assuming that π is a generic ranking, the probability for this ranking is given by

$$Pr(\theta) = \frac{\exp\big(-\theta d(\pi, \pi_0)\big)}{\psi(\theta)}, \tag{4.7}$$

where d is a ranking distance measure and $\psi(\theta)$ is a normalization constant.

In this chapter, we set two simulation scenarios, in the first one (Model I), we generated

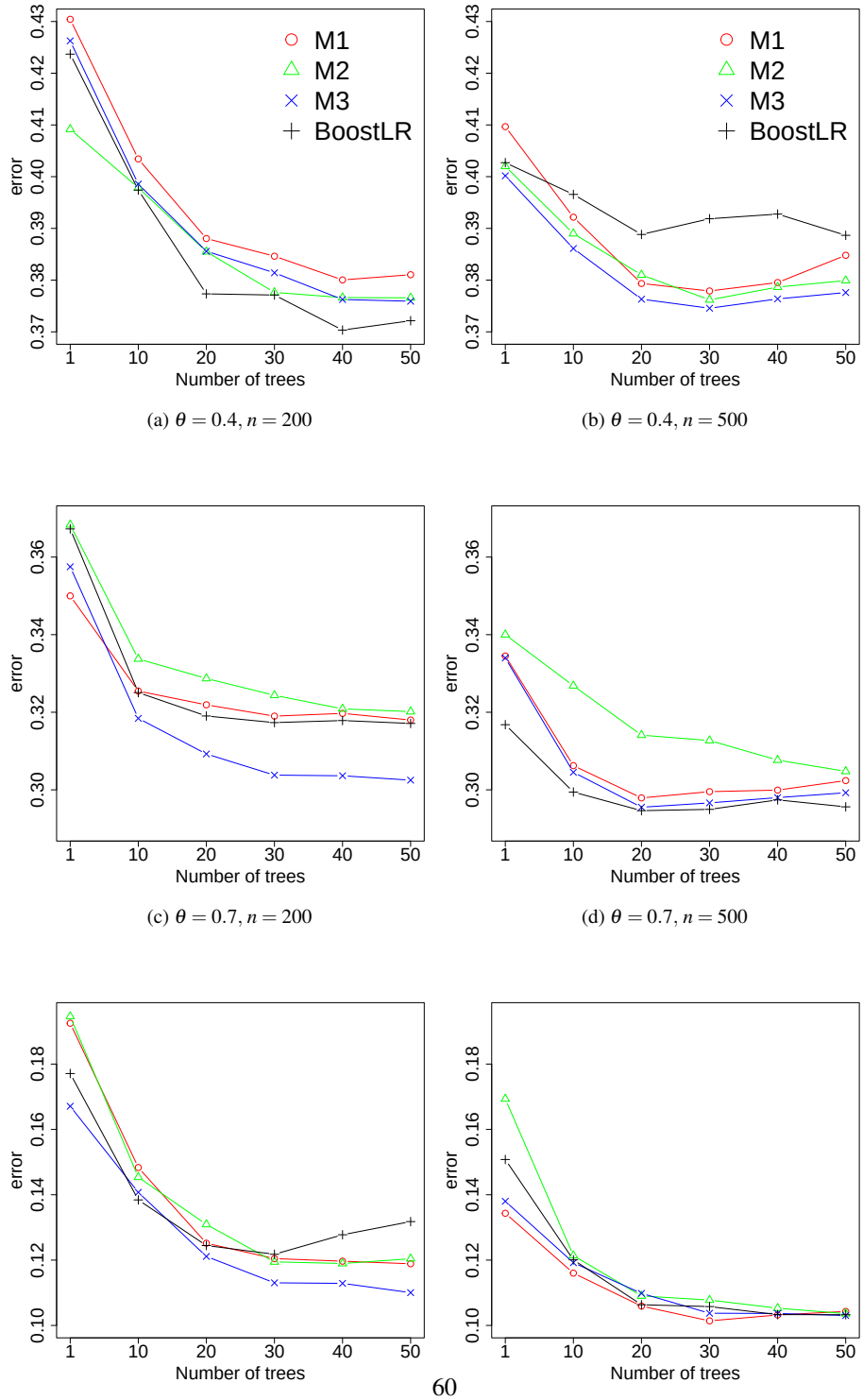
rankings assuming the Kemeny distance d_K as defined in Eq. (3.2). In the second simulation study, we assume the item-weighted Kemeny distance $d_{K,e}$ as defined in Eq. (3.9). In both scenarios, we let the dispersion parameter θ vary according to three different levels of noise (low with $\theta = 2$, medium with $\theta = 0.7$ and high with $\theta = 0.4$). In this way, we compare the robustness to label noise of the three proposed algorithms. Finally, we consider two levels for the sample size ($N = 200$, $N = 500$) in the experimental design. We fix the number of trees B to 50, the maximum depth of each tree to 4, and run a five-fold cross validation to provide confident results.

Simulation under model I

In the first simulation scenario, we generate rankings using the unweighted Kemeny distance d_K. We produce six different datasets, considering two levels for the sample size N in the experimental design and three different noise levels θ. We applied the three ensemble methods defined in Section 4 to all six datasets, considering three different weighting vectors: $\mathbf{w}_1 = (1,1,1,1)$, $\mathbf{w}_2 = (5,2,2,5)$ and $\mathbf{w}_3 = (10,1,1,10)$. Let us remind that $\mathbf{w}_1$ will produce an unweighted version of the algorithm since it assumes indifference over alternatives. Hence, under $\mathbf{w}_1$, the comparison of our method with the state-of-the-art, Dery and Shmueli (2020)'s BoostLR who, showed to outperform other tree-based label ranking algorithms in the literature, can be carried out.
In contrast, $\mathbf{w}_2$ assigns higher weights to the external items, and finally, $\mathbf{w}_3$ concentrates most of the weight mass on the external items, so the inversion between the first and the last item will be over-penalised (see Albano and Plaia (2021) for a detailed discussion on the item-weighting scheme).
Figs.4.3, 4.4, 4.5 compare AdaBoost.R.M1, AdaBoost.R.M2 and AdaBoost.R.M3 in all the simulated datasets, plotting the five-fold cross validation error ($err(b)$ Eq.4.6) vs the number of trees b. For the sake of readability, the scale of the y-axis changes when θ varies but remains the same when n varies. In this way, it is possible to compare the graphs row by row visually. The starting point of each line is the error corresponding to the first tree; therefore, the plot shows the improvement produced by the boosting algorithms when the number of trees grows up (after each block of ten trees, the average $err(b)$ is displayed).
Figure 4.3 allows us to compare our proposal with the state-of-the-art BoostLR. In four out of six scenarios, the prediction errors of Adaboost.R.M3 are lower than those of BoostLR. Specifically, BoostLR produces consistently superior results in the first scenario, characterised by a significant degree of variability ($\theta = 0.4$) and small sample size ($n = 200$).

In general, the AdaBoost.R.M3 algorithm has high predictive performance. Indeed, in 70% of the scenarios, AdaBoost.R.M3 performs best; this is particularly evident with a mild level of label noise ($\theta = 0.7$). However, in some scenarios, e.g. high noise levels ($\theta = 0.4$) and small sample size $n = 200$, AdaBoost.R.M2 and AdaBoost.R.M1 can achieve comparable or lower errors than AdaBoost.R.M3.

The simulation study allows highlighting the impact of the weights. Indeed, under Model I, rankings were generated using the unweighted Kemeny distance d_K in the Mallows mode; thus, a strongly unbalanced weighting vector (such as $\mathbf{w}_3$) tends to destabilise the prediction errors leading to unstable trends. In other words, as we move away from the assumptions of the data-generating process, the results get worse.

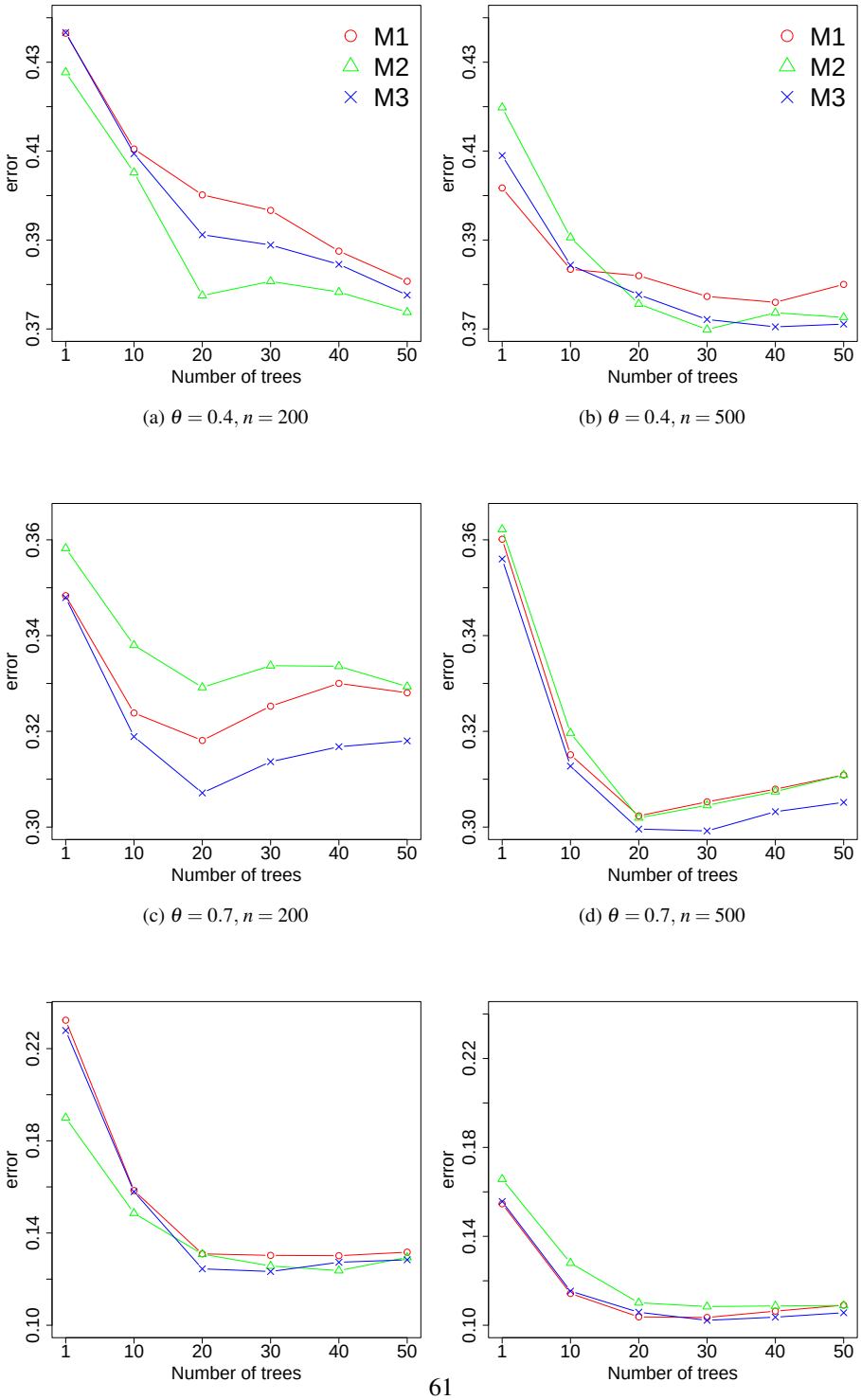


(a) $\theta = 0.4, n = 200$ (b) $\theta = 0.4, n = 500$

(c) $\theta = 0.7, n = 200$ (d) $\theta = 0.7, n = 500$

(e) $\theta = 2, n = 200$ (f) $\theta = 2, n = 500$

Figure 4.3: AdaBoost.R.M1, AdaBoost.R.M2, AdaBoost.R.M3 and BoostLR (Dery and Shmueli, 2020) for all the simulated scenarios with 50 trees. Different levels of homogeneity among the rankings, $\theta = (0.4, 0.7, 2)$, two sample sizes, $n = (200, 500)$ and weights $\mathbf{w}_1 = (1, 1, 1, 1)$, Model I.



(a) $\theta = 0.4, n = 200$

(b) $\theta = 0.4, n = 500$

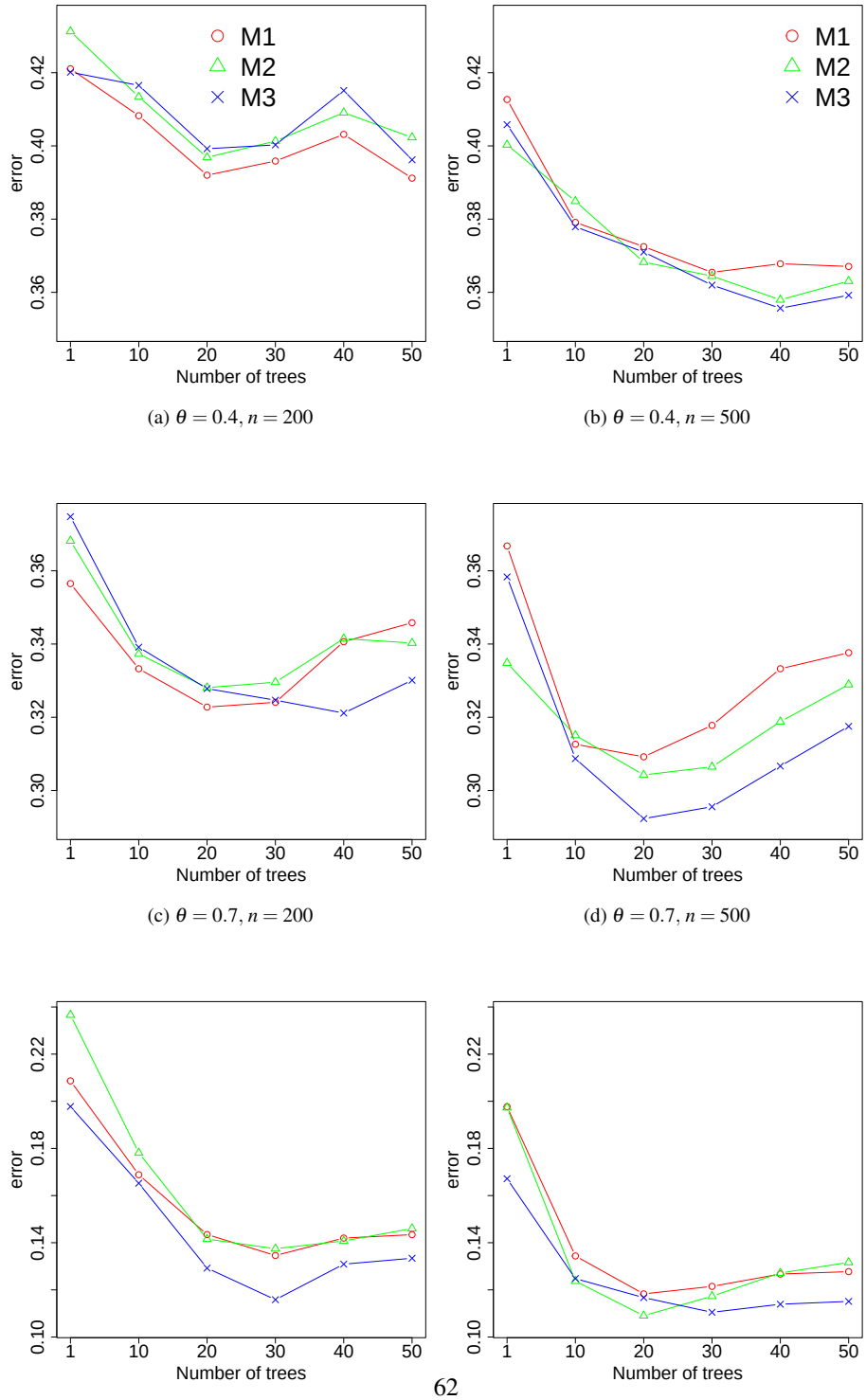
(c) $\theta = 0.7, n = 200$

(d) $\theta = 0.7, n = 500$

(e) $\theta = 2, n = 200$

(f) $\theta = 2, n = 500$

Figure 4.4: AdaBoost.R.M1, AdaBoost.R.M2 and AdaBoost.R.M3 for all the simulated scenarios with 50 trees. Different levels of homogeneity among the rankings, $\theta = (0.4, 0.7, 2)$, two sample sizes, $n = (200, 500)$ and weights $\mathbf{w}_2 = (5, 2, 2, 5)$, Model I.



(a) $\theta = 0.4, n = 200$

(b) $\theta = 0.4, n = 500$

(c) $\theta = 0.7, n = 200$

(d) $\theta = 0.7, n = 500$

(e) $\theta = 2, n = 200$

(f) $\theta = 2, n = 500$

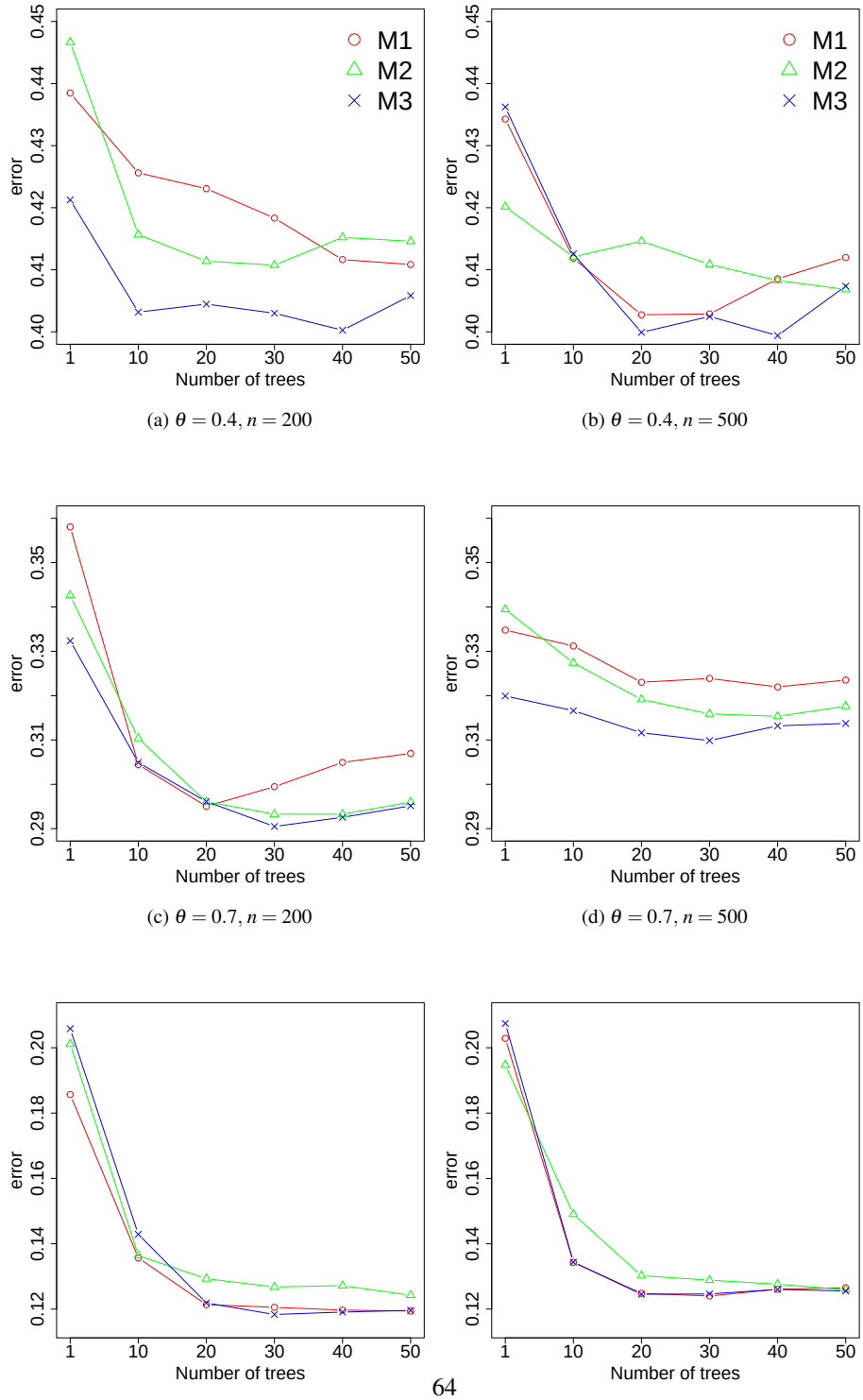
Figure 4.5: AdaBoost.R.M1, AdaBoost.R.M2 and AdaBoost.R.M3 for all the simulated scenarios with 50 trees. Different levels of homogeneity among the rankings, $\theta = (0.4, 0.7, 2)$, two sample sizes, $n = (200, 500)$ and weights $\mathbf{w}_3 = (10, 1, 1, 10)$, Model I.

Simulation under model II

In the second simulation study scenario, we analyse the three proposed algorithms assuming consistency with the data generating process. That is, we generate rankings using the item-weighted Kemeny distance $d_{K,e}$ into the Mallows model (4.7) and consider the same weighting scheme in the boosting phase. For setting the simulation study, we exploit the *item similarity criterion*; as stated in Section 2.2, the weights are assigned following the idea that swapping two elements that can be considered similar in some aspects should be less penalised than swapping two dissimilar ones. Therefore,we define a symmetric penalization matrix **P** (Tab.4.4, reflecting the dissimilarity between the four elements to be ranked $\{A,B,C,D\}$).
Such a weighting scheme is easily fitted in the real world; for example, suppose n students are asked to rank 4 different subjects, namely $items = \{$A= Maths, B=Physics, C=History, D=Philosophy$\}$. It is reasonable to assume that the rankings of two students who disagree on very different subjects (e.g. mathematics and philosophy) are more different than those of two students who disagree on similar subjects (e.g. mathematics and physics or philosophy and history).
We consider two levels for the sample size N, three different noise levels θ and the item similarity weighting scheme with penalisation matrix **P** defined in Tab.4.4.
Figure 4.6 shows that, on average, AdaBoost.R.M3 has the best predictive performance, although the differences between the three procedures are slight in setting with low label noise.

Table 4.4: Simulated data penalization matrix **P**.

	A	B	C	D
A	0	5	20	20
B	5	0	20	20
C	20	20	0	5
D	20	20	5	0



(a) $\theta = 0.4, n = 200$

(b) $\theta = 0.4, n = 500$

(c) $\theta = 0.7, n = 200$

(d) $\theta = 0.7, n = 500$

(e) $\theta = 2, n = 200$

(f) $\theta = 2, n = 500$

Figure 4.6: AdaBoost.R.M1, AdaBoost.R.M2 and AdaBoost.R.M3 for all the simulated scenarios with 50 trees. Different levels of homogeneity among the rankings, $\theta = (0.4, 0.7, 2)$, two sample sizes, $n = (200, 500)$ and a penalization matrix $\mathbf{P}$, Model II.

4.4.2 Real Data applications

The performance of the item-weighted boosting method are also investigated through an application to three real datasets summarized in Tab. 4.5 and analyzed also by Aledo et al. (2017); de Sá et al. (2017, 2018), Werbin-Ofir et al. (2019), Dery and Shmueli (2020) and Plaia et al. (2021a).

Dataset	Judges	Covariates	Labels
German2005	402	31	5
German2009	407	33	5
Top7Movies	300	7	7

Table 4.5: Characteristics of the real-world datasets.

The first two datasets regard election data and contain socio-economic information from regions of Germany and its electoral results, which took place in 2005 and 2009. The 413 records correspond to the administrative districts of Germany, which are described by 39 covariates. The outcome is the set of rankings on five items: CDU (conservative), SPD (centre-left), FDP (liberal), Green (centre-left) and Left (left-wing). To define the weighting scheme, we follow the intuition that swapping similar parties (e.g. two different parties belonging to the centre-left) should have a smaller impact on the results of an election than swapping two dissimilar ones (e.g. a conservative and a left-wing). Therefore, we arrange parties in a straight line, Fig. 4.7, where the Left-wing and Conservative parties are at the extremes.

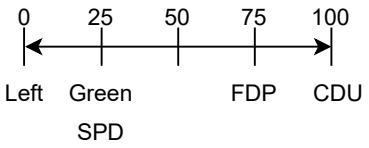


Figure 4.7: Parties similarity

Then, we introduce the penalisation matrix shown in Tab. 4.6 which approximates the similarity structure of parties. Let z_i be the co-ordinate of the i-th political party along the line defined in Fig. 4.7, then the dissimilarity between party i and party j will be found as

$$Dis(i,j) = |z_i - z_j|.$$

In the case of parties Green and SPD, which have the same co-ordinate, their dissimilarity is assumed to be equal to 5.

Table 4.6: Political parties penalization matrix **P**.

	CDU	SPD	FDP	Green	Left
CDU	0	75	25	75	100
SPD	75	0	50	5	25
FDP	25	50	0	50	75
Green	75	5	50	0	25
Left	100	25	75	25	0

The item-weighted boosting algorithm was performed on a limited number of trees ($B = 50$) and considering a maximum depth (number of the splits in the tree) equal to 4. Figs. 4.8, 4.9 show the prediction error of the item-weighted boosting applied to German Elections datasets with penalization matrix **P** (Tab. 4.6), while variable importance can be found in the appendix (Fig. B.1).
The boosting procedures were evaluated through a five-fold cross validation. As expected, the accuracy improves when the number of trees grows up. Therefore, the item-weighted boosting minimizes the prediction error taking into account the similarity structure of items.
As regards German2005 dataset, AdaBoost.R.M3 achieves the best results. Moreover, the prediction errors $err(b)$ of the three procedures become generally quite stable after 30 trees. In contrast, AdaboostM2 and AdaboostM3 have comparable errors in the German2009 dataset.

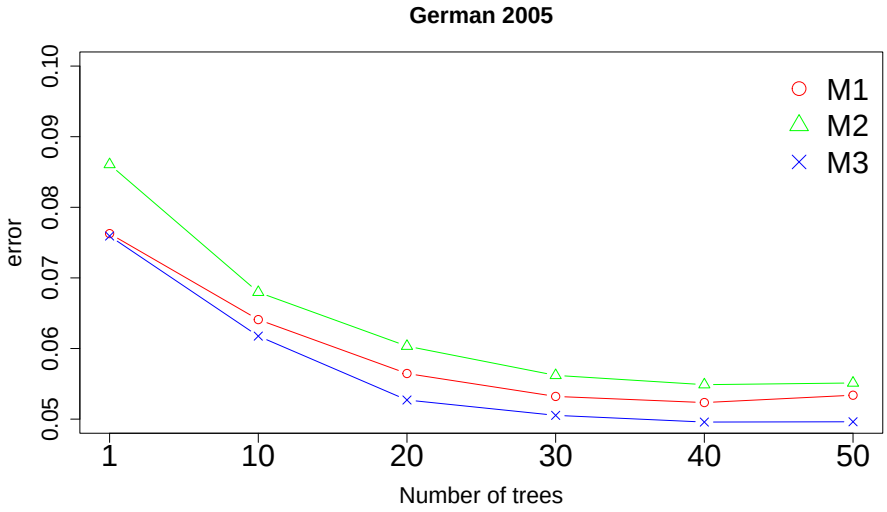


Figure 4.8: Item-weighted boosting applied to German Elections dataset 2005: $err(b)$.

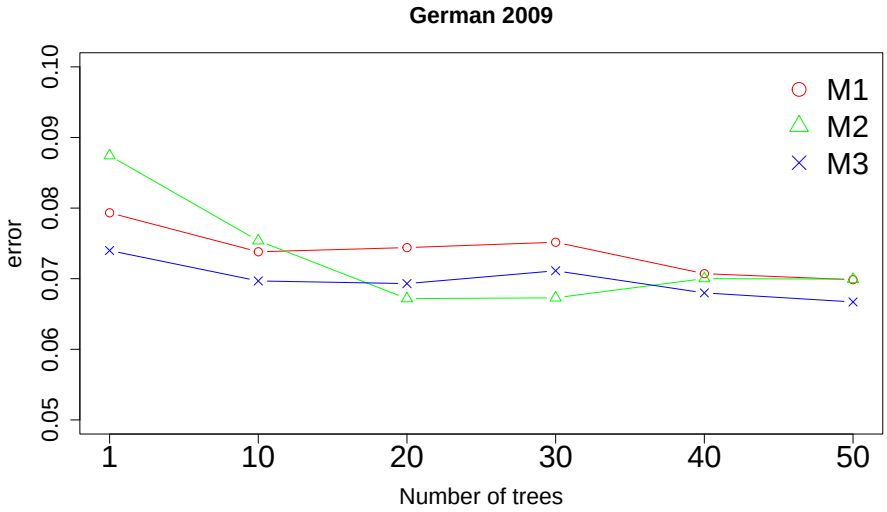


Figure 4.9: Item-weighted boosting applied to German Elections dataset 2009: $err(b)$.

The last dataset, The Top7movies, was presented in the chapter by de Sá et al. (2018). It has been derived as a subset of the MovieLens 1M Dataset (Harper and Konstan,

2015) [1]. The original dataset has 1 million ratings from 6000 users on 4000 movies. Demographic data such as gender, age, occupation, city, state, latitude and longitude are available for each user. de Sá et al. (2018) selected the subset of users who have rated all the seven most rated movies. This means that demographic data and a ranking of 7 movies per user are obtained. The labels in this dataset represent the following movies:

A: American Beauty (1999) (Drama)

B: Star Wars: Episode IV—A New Hope (1977) (Action, Adventure, Fantasy)

C: Star Wars: Episode V—The Empire Strikes Back (1980) (Action, Adventure, Fantasy)

D: Star Wars: Episode VI—Return of the Jedi (1983) (Action, Adventure, Fantasy)

E: Jurassic Park (1993) (Action, Adventure, Sci-Fi)

F: Saving Private Ryan (1998) (Drama, War)

G: Terminator 2: Judgment Day (1991) (Action, Sci-Fi)

In the dataset, many rankings contain ties. In addition, we decided to take a random sample of $N = 300$ from the original dataset.
For defining a Penalisation matrix $\mathbf{P}$, we retrieved from IMDb [2] website the genres of each film and compared them to obtain the similarity structure. Note that the number of genres provided for each film varies from one to three.
Let G_i denote the set whose elements are the different genres of film i, for example, $i = E$ (where E=Jurassic Park) then $G_E = \{\text{Action, Adventure, Sci-fi}\}$. The dissimilarity between two generic movies i and j is computed according to:

$$Dis(i,j) = \begin{cases} 90 - 30 \cdot \#(G_i \cap G_j) & \text{if } G_i \cap G_j \neq G_i \cup G_j \\ 5 & \text{if } G_i \cap G_j = G_i \cup G_j \end{cases} \tag{4.8}$$

For example, since $G_E = \{\text{Action, Adventure, Sci-fi}\}$ and $G_B = \{\text{Action, Adventure, Fantasy}\}$ then $Dis(E,B) = 30$. The Penalization matrix is shown in detail in Tab. 4.7.
In Fig.4.10, the error $err(b)$ (4.6) vs the number of trees b, both in training and in the test set, are plotted, while Tab.B.1, in the appendix, reports the importance of the variables.

[1] https://grouplens.org/datasets/movielens/1m/
[2] https://www.imdb.com/

Table 4.7: Top7 movies penalization matrix **P**.

	A	B	C	D	E	F	G
A	0	90	90	90	90	60	90
B	90	0	5	5	30	90	60
C	90	5	0	5	30	90	60
D	90	5	5	0	30	90	60
E	90	30	30	30	0	90	30
F	60	90	90	90	0	90	90
G	90	60	60	60	30	90	0

Fig. 4.10 shows that AdaBoost.R.M3 and AdaBoost.R.M1 prediction error are very similar, lower than AdaBoost.R.M2 until 20 trees, where a local minimum is hit. It can be considered as a stopping point for the iterative process; in fact, after the 20th iteration, AdaBoost.R.M3 and AdaBoost.R.M1 prediction error starts to increase and then stabilises.

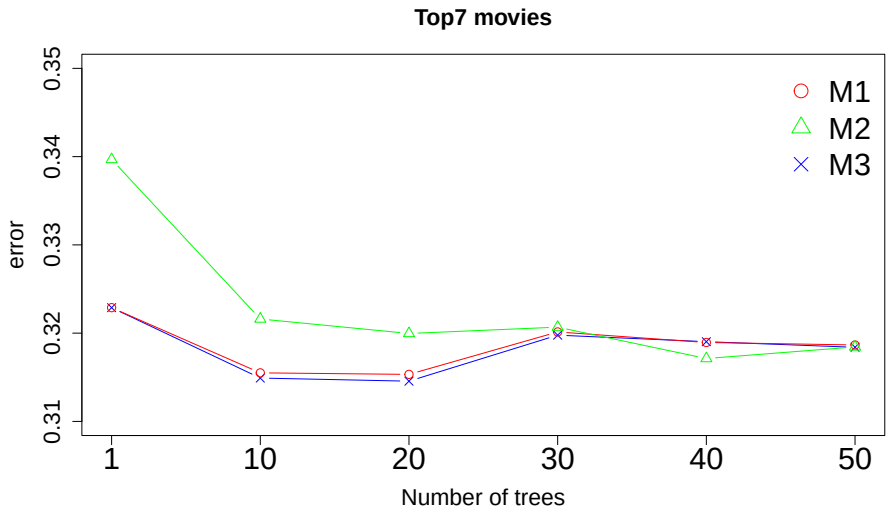


Figure 4.10: Item-weighted boosting applied to Top7movies dataset: $err(b)$.

4.5 Concluding remarks

The novelty of this chapter is to propose an item-weighted approach for Label Ranking where the similarity structure and the importance of labels in each ranking are considered. Specifically, we define three item-weighted versions (AdaBoost.R.M1, AdaBoost.R.M2, and AdaBoost.R.M3) of the boosting algorithm AdaBoost for label ranking. The procedures are implemented in `R`, by incorporating some functions of `ConsRank` package (D'Ambrosio, 2021) within a user-written split function of `rpart` library (Therneau et al., 2015) and `adabag` (Alfaro et al., 2013). The algorithms combine many weighted distance-based trees for ranking data to obtain a flexible, strong learner. The key idea is to consider the item-weighted Kemeny distance $d_{K,e}$ (Albano and Plaia, 2021) as a measure of impurity in the splitting process and its related rank correlation coefficient $\tau_{x,e}$ to identify the median ranking in the final nodes. This approach is particularly fitting when dealing with multi-level data, as shown in Section 4.4, where the data matrix contains rankings of political parties (level 1) who belong to political coalitions (level 2). In this case, an unweighted label ranking procedure, which assumes indifference among alternatives, would merely minimise the total distance without considering the similarity of political parties.

The three proposed methods differ in how they deal with label noise; that is AdaBoost.R.M1 (1) does not counteract label noise, while AdaBoost.R.M2 (2) updates the working weights at each iteration b using a "gentle" function, and AdaBoost.R.M3 (3) obtains the final predicted rankings by penalising the bad intermediate trees of the iterative process. The three methods are compared, investigating their performance through real and simulated data applications. In particular, we demonstrate that AdaBoost.R.M3 performs best in many scenarios, having the lowest prediction errors even at a high noise level. We also investigate the computational effort of the item-weighted consensus procedure, highlighting a slight increase in average computation time.

Moreover, although the goodness of LR learners is mainly evaluated through their predictive performance, we highlight that the proposed item-weighted LR algorithm can also be used as an interpretative method to select and measure the overall covariates' importance, rather than a "black box" that forecasts without a clear understanding of the underlying rules.

Future research should consider the potential effects of weights to improve the algorithm scalability concerning the number of items and might include a component that automatically learns label weights to generalise the answer to any label ranking issue avoiding the need for a domain expert.

Chapter 5

A family of distances for preference-approvals

5.1 Introduction

In social choice theory, preference rankings and approvals are two popular ways to collect the preferences of a group of agents on a set of alternatives. Preference rankings order the alternatives from best to worst without distinguishing between acceptable and unacceptable alternatives. That is, if a is ranked above b, we can only infer that a is preferred to b, but we cannot infer anything about their absolute acceptability. In contrast, the approval voting system (Brams and Fishburn, 1978) consists of separating the set of acceptable alternatives from the set of unacceptable alternatives without considering preferences neither over acceptable nor over unacceptable alternatives.

Preference rankings and approval voting are related, but they are basically different types of information and cannot be inferred from each other.

In this chapter, we focus on preference-approval structures. They combine preferences over the alternatives, through a weak order, and establish which alternatives are acceptable (Brams 2008, Chapter 3; Brams and Sanver 2009; Sanver 2010). In preference-approval structures, voters can pay attention to which alternatives are acceptable and simultaneously rank-order them. Voters may either rank-order unacceptable alternatives or avoid declaring their preferences about them[1] by (implicitly) showing indifference

[1] This is the case of fallback voting in Brams and Sanver (2009).

between these alternatives[2].

Generally, extensions to ranking measures have mainly focused on the definition of weighted distances (see García-Lapresta and Pérez-Román 2010; Albano and Plaia 2021; Plaia et al. 2021b). In the last years, there has been a dramatic increase in recent publications about preference-approval structures and the introduction of consensus and distance measures in that setting.

Erdamar et al. (2014) introduced a family of distances in the preference-approval setting, and they applied them to measure the consensus in that framework. Kamwa (2019) studied the propensity of the preference-approval voting of electing the Condorcet winner/loser when they exist.

Dong et al. (2021) established some axioms implying the existence of a distinct distance function of preference-approval systems. They investigated a preferences aggregation model in the context of group decision-making based on the proposed axiomatic distance function.

Kruger and Sanver (2021) investigated the compatibility between ordinal and evaluative approaches to social choice theory under two weak assumptions: respect for unanimity and independence of evaluation of each alternative. They claimed that there is an incompatibility between the two and described some options whenever the second assumption is relaxed.

Long et al. (2021) developed a two-stage consensus reaching method for multi-attribute group decision making problems with preference-approval structures, promoting the efficiency of consensus reaching.

Barokas and Sprumont (2022) extended the classing Borda count to rank alternatives in preference-approval setting, constructing an axiomatization of a new aggregation procedure called *broken Borda rule*.

In this chapter, we propose a new distance for preference-approvals, following the axiomatic approach of the Kemeny distance. However, while the Kemeny distance can only consider the *preference-discordance*, our approach takes into account the *approval-discordance* as well and use an aggregation function to combine the two types of information for each pair of alternatives.

We show that using, as an aggregation function, the family of *weigthed power means* (a class of weighted quasiarithmetic means) brings the benefit of many interesting properties. The final aggregated distance will thus be the sum of the pairwise preference-approval discordances. Furthermore, we show that our distance respects the funda-

[2] If the number of alternatives is large, voters may have difficulties to rank-order all the alternatives (see Dummett 1984, p. 243).

mental properties to be defined as a metric and that, under certain assumptions, it has a precise geometric interpretation.
Our proposal can be regarded as the generalization of the Erdamar et al. (2014) distance measure, with the two coinciding for a specific parameter setting. However, we show that the proposed distance family has some advantages over the existing one as it is more versatile and performs better in cluster analysis.

Finally, the proposed metric is used to cluster a set of preference-approvals into homogeneous groups, considering the whole 2-dimensional universe of preference-approvals and a real case study.
The chapter is organized as follows. Section 5.2 is devoted to introduce basic notation, preference-approvals and the codifications used throughout the chapter. Section 5.3 includes our proposal for measuring distances between preference-approvals and some results. Section 5.4 offers some applications for the clustering task. Finally, Section 5.5 concludes the chapter with some remarks.

5.2 Preference-approval

Consider that a set of voters $V = \{v_1, \ldots, v_n\}$, with $n \geq 2$, have to express their opinions over Y. We assume that each voter ranks the alternatives in Y by means of a weak order and, additionally, assesses each alternative as either acceptable or unacceptable by partitioning Y into A, the set of *acceptable* alternatives, and $U = Y \setminus A$, the set of *unacceptable* alternatives, where A and U can be empty sets.
We also assume the following consistency condition: given two alternatives y_i and y_j, if y_j is acceptable and y_i is ranked above y_j, then y_i should be acceptable as well.

Definition 1 *A* preference-approval *on Y is a pair $(\pi, A) \in W(Y) \times \mathscr{P}(Y)$ satisfying the following condition:*

$$\forall y_i, y_j \in Y \left((y_i \succ y_j \text{ and } y_j \in A) \Rightarrow y_i \in A \right).$$

With $\mathscr{R}(Y)$ we denote the set of preference-approvals on Y.

Remark 1 If $(\pi, A) \in \mathscr{R}(Y)$, then the following conditions are satisfied:

1. $\forall y_i, y_j \in Y \left((y_i \in A \text{ and } y_j \in U) \Rightarrow y_i \succ y_j \right)$.
2. $\forall y_i, y_j \in Y \left((y_i \, \pi \, y_j \text{ and } y_i \in U) \Rightarrow y_j \in U \right)$.

We now illustrate preference-approval structures through the following example.

Example 1 Let us consider $(\pi, A) \in \mathscr{R}(\{y_1, \ldots, y_8\})$ represented by

$$\begin{array}{c} y_4 \\ y_1 \; y_6 \\ y_2 \\ \hline y_3 \\ y_5 \; y_7 \; y_8. \end{array}$$

It means that alternatives in the same row are indifferent, alternatives in upper rows are preferred to those located in lower rows, alternatives above the line are acceptable, i.e., $A = \{y_1, y_2, y_4, y_6\}$, and those below the line are unacceptable, i.e., $U = \{y_3, y_5, y_7, y_8\}$.

Table 5.1 includes the number of possible approvals, linear orders, weak orders and preference-approvals when the number of alternatives is $m = 2, 3, \ldots, 10$.

m	Approvals	Preferences		Preference-approvals
		Linear orders	*Weak orders*	
2	4	2	3	8
3	8	6	13	44
4	16	24	75	308
5	32	120	541	2 612
6	64	720	4 683	25 988
7	128	5 040	47 293	296 564
8	256	40 320	545 835	3 816 548
9	512	362 880	7 087 261	54 667 412
10	1 024	3 628 800	102 247 563	862 440 068

Table 5.1: Number of approvals, linear orders, weak orders and preference-approvals.

It is well-known that the total number of approvals (subsets of Y) and linear orders are 2^m and $m!$, respectively. The number of weak orders is $m!(\log_2 e)^{m+1}/2$ (see Good 1980). The formula for calculating the number of preference-approvals has never been defined in the literature. For the first time, the exact number of preference-approvals for $m = 2, 3, \ldots, 10$ alternatives is reported herein in Tab. 5.1. The formula to compute the exact number of preference-approvals on a set of m alternatives is

$$\omega(n) = \sum_{r=0}^{m} (r+1)! \, S_m^{(r)}, \tag{5.1}$$

where $S_m^{(r)}$ is a Stirling integer (number) of the second kind defined by David and Barton (1962, p. 294), Abramowitz and Stegun (1964, p. 824) and more thoroughly in Fisher and Yates (1953, p. 78), while r denotes the number of distinct positions in a weak order on m alternatives, also known as *buckets*. For example, considering four alternatives, if two are tied for first place, and the other two are tied for third place, we can say that the number of distinct positions, or buckets, is two.

Table 5.2 shows the quotients between preference-approvals and approvals. In turn, Tab. 5.3 shows the quotients between preference-approvals and weak orders.

It is clear that the expressivity of voters explodes with preference-approvals.

m	Quotients
2	2
3	5.5
4	19.25
5	81.62
6	406.06
7	2 316.91
8	14 908.39
9	106 772.29
10	842 226.63

Table 5.2: Quotients between preference-approvals and approvals.

m	Quotients
2	2.67
3	3.38
4	4.11
5	4.83
6	5.55
7	6.27
8	6.99
9	7.71
10	8.43

Table 5.3: Quotients between preference-approvals and weak orders.

5.2.1 Codifications

Given $A \subseteq Y$, the *indicator function* (or *characteristic function*) of A, $I_A : Y \longrightarrow \{0,1\}$, is defined as

$$I_A(y_i) = \begin{cases} 1, \text{ if } y_i \in A, \\ 0, \text{ if } y_i \in Y \setminus A. \end{cases} \tag{5.2}$$

Remark 2 Every preference-approval $(\pi, A) \in \pi(\{y_1, \ldots, y_m\})$ can be codified in terms of $P_\pi(y_i)$ (Eq. (2.1)) and $I_A(y_i)$ (Eq. (5.2)) as follows:

$$\Big(P_\pi(y_1), P_\pi(y_2), \ldots, P_R(y_m)\Big) \,\&\, \Big(I_A(y_1), I_A(y_2), \ldots, I_A(y_m)\Big). \tag{5.3}$$

Example 2 Consider the preference-approval $(\pi, A) \in \mathscr{R}(\{y_1, y_2, y_3, y_4\})$ represented

by

$$\begin{array}{c} y_4 \\ y_1 \\ y_2 \\ \hline\hline \\ y_3 \end{array}$$

Following Eq. (5.3), (π, A) is codified as $(2,3,4,1)\,\&\,(1,1,0,1)$.

The *sign function*, $\operatorname{sgn}: \mathbb{R} \longrightarrow \{-1,0,1\}$, is defined as

$$\operatorname{sgn}(a) = \begin{cases} 1, & \text{if } a > 0, \\ 0, & \text{if } a = 0, \\ -1, & \text{if } a < 0. \end{cases}$$

5.3 The proposal

Given two preference-approvals $\Big((\pi_1, A_1), (\pi_2, A_2)\Big) \in \mathscr{R}(Y)$ and two generic alternatives $y_i, y_j \in Y$, we now introduce two indices that measure the discordances between these alternatives with respect to preference and approvals, respectively.
The *pairwise preference-discordance* between y_i and y_j is defined as

$$p_{ij} = \frac{1}{2} \cdot |\operatorname{sgn}\big(P_{\pi_1}(y_j) - P_{\pi_1}(y_i)\big) - \operatorname{sgn}\big(P_{\pi_2}(y_j) - P_{\pi_2}(y_i)\big)|. \tag{5.4}$$

Taking into account Eq. (2.2), Eq. (5.4) can be defined in an equivalent and simpler way:

$$p_{ij} = \frac{1}{2} \cdot |O_{\pi_1}(y_i, y_j) - O_{\pi_2}(y_i, y_j)|, \tag{5.5}$$

and therefore, $p_{ij} \in \{0, 0.5, 1\}$.
The *pairwise approval-discordance* between y_i and y_j is defined as

$$a_{ij} = \frac{1}{2} \cdot \Big(|I_{A_1}(y_i) - I_{A_2}(y_i)| + |I_{A_1}(y_j) - I_{A_2}(y_j)|\Big), \tag{5.6}$$

and again $a_{ij} \in \{0, 0.5, 1\}$.
In both cases, the values of 0, 0.5 and 1 indicate a null, moderate and high discordance, respectively. In order to generate a global measure of discordance between two alternatives, we consider an aggregation function (see Beliakov et al. 2007; Grabisch et al. 2009; Ramík and Vlach 2012, Sect. 2, among others).

Definition 2 *Given an aggregation function* $h : [0,1] \times [0,1] \longrightarrow [0,1]$*, the distance associated with h,* $D : \mathscr{R}(Y) \times \mathscr{R}(Y) \longrightarrow [0,1]$*, is defined as*

$$D\Big((\pi_1, A_1), (\pi_2, A_2)\Big) = \frac{2}{m \cdot (m-1)} \cdot \sum_{\substack{i,j=1 \\ i<j}}^{m} h(p_{ij}, a_{ij}). \tag{5.7}$$

Among the huge variety of aggregation functions, in this proposal, we consider a class of weighted quasiarithmetic means[3]: the family of *weighted power means*, $h : [0,1] \times [0,1] \longrightarrow [0,1]$, defined as

$$h(x,y) = \left(\lambda \cdot x^r + (1-\lambda) \cdot y^r\right)^{\frac{1}{r}}, \tag{5.8}$$

where $\lambda \in [0,1]$ and $r > 0$.

Remark 3 Weighted power means, defined in Eq. (5.8), have interesting properties (see, for instance, Beliakov et al. (Beliakov et al., 2007, pp. 45-47)):

1. *Continuity*: h is continuous.
2. *Monotonicity*: $(x \leq x'$ and $y \leq y') \Rightarrow h(x,y) \leq h(x',y')$, for all $x,y,x',y' \in [0,1]$.
3. *Idempotency*: $h(x,x) = x$ for every $x \in [0,1]$.
4. *Compensativeness*: $\min\{x,y\} \leq h(x,y) \leq \max\{x,y\}$ for all $x,y \in [0,1]$.
5. *Comparability*: h is increasing in r.
6. *Symmetry*: $h(x,y) = h(y,x)$ for all $x,y \in [0,1] \Leftrightarrow \lambda = 0.5$.
7. $\lim_{r\to\infty} h(x,y) = \max\{x,y\}$.
8. $\lim_{r\to 0} h(x,y) = x^{\lambda} \cdot y^{1-\lambda}$ (weighted geometric mean).

Notice that the inputs of h in Eq. (5.7) are the pairs of $0, 0.5, 1$. In Tables 5.4 and 5.5 we show the values of h for these pairs and different values of the parameter r for $\lambda = 0.5, 0.75$, respectively.

According to Tables 5.4 and 5.5, the parameter r governs the penalty for each pair of values. Indeed, as r increases, so does the value of $h(p_{ij}, a_{ij})$. As a result, taking an excessively large r value results in very similar penalties and reduces the weight of high discordance compared to moderate discordance.

(x,y)	$h(x,y)$					
	$r=0.5$	$r=1$	$r=1.5$	$r=2$	$r=5$	$r=10$
$(0,0)$	0	0	0	0	0	0
$(0,0.5)$	0.12	0.25	0.31	0.35	0.44	0.47
$(0,1)$	0.25	0.50	0.63	0.71	0.87	0.93
$(0.5,0)$	0.12	0.25	0.31	0.35	0.44	0.47
$(0.5,0.5)$	0.50	0.50	0.50	0.50	0.50	0.50
$(0.5,1)$	0.73	0.75	0.77	0.79	0.88	0.93
$(1,0)$	0.25	0.50	0.63	0.71	0.87	0.93
$(1,0.5)$	0.73	0.75	0.77	0.79	0.88	0.93
$(1,1)$	1	1	1	1	1	1

Table 5.4: Values of h for $\lambda = 0.5$.

(x,y)	$h(x,y)$					
	$r=0.5$	$r=1$	$r=1.5$	$r=2$	$r=5$	$r=10$
$(0,0)$	0	0	0	0	0	0
$(0,0.5)$	0.03	0.12	0.20	0.25	0.38	0.44
$(0,1)$	0.06	0.25	0.40	0.50	0.76	0.87
$(0.5,0)$	0.28	0.38	0.41	0.43	0.47	0.49
$(0.5,0.5)$	0.50	0.50	0.50	0.50	0.50	0.50
$(0.5,1)$	0.61	0.62	0.64	0.66	0.77	0.87
$(1,0)$	0.56	0.75	0.83	0.87	0.94	0.97
$(1,0.5)$	0.86	0.88	0.89	0.90	0.95	0.97
$(1,1)$	1	1	1	1	1	1

Table 5.5: Values of h for $\lambda = 0.75$.

Taking into account Eq. (5.7) with the aggregation function h in Eq. (5.8), we now introduce the family of distances on preference-approvals that we analyze in the present chapter.

Definition 3 *Given $\lambda \in [0,1]$ and $r > 0$, the distance associated with λ and r is the*

[3]They are defined as $h(x,y) = g^{-1}\Big(\lambda \cdot g(x) + (1-\lambda) \cdot g(y)\Big)$, where g is a generating function (see, for instance, Ostasiewicz and Ostasiewicz, 2000 and Beliakov et al., 2007, Section 2.3).

mapping $D^r_\lambda : \mathscr{R}(Y) \times \mathscr{R}(Y) \longrightarrow [0,1]$ *defined as*

$$D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = \frac{2}{m\cdot(m-1)} \cdot \sum_{\substack{i,j=1 \\ i<j}}^{m} \Big(\lambda \cdot p^r_{ij} + (1-\lambda)\cdot a^r_{ij}\Big)^{\frac{1}{r}}. \tag{5.9}$$

Remark 4 When $r = 2$ and $\lambda = 0.5$, the geometric interpretation of $h(p_{ij},a_{ij})$ is related to the Euclidean distance.

Fig. 5.1 reports the *preference-approval plane*, that is a Euclidean plane having on the x-axis the pairwise preference-discordance, p_{ij}, and on the y-axis the pairwise approval-discordance, a_{ij}.

If $r = 2$ and $\lambda = 0.5$, then $h(p_{ij},a_{ij})$ is proportional to the Euclidean distance between (p_{ij},a_{ij}) and the origin, $(0,0)$, $d\Big((p_{ij},a_{ij}),(0,0)\Big)$:

$$h(p_{ij},a_{ij}) = \sqrt{0.5\cdot(p^2_{ij}+a^2_{ij})} = \sqrt{0.5}\cdot d\Big((p_{ij},a_{ij}),(0,0)\Big), \text{ i.e.,}$$
$$h(p_{ij},a_{ij}) \propto d\Big((p_{ij},a_{ij}),(0,0)\Big).$$

This means that the aggregation function h can be interpreted as a proper distance in the preference-approval plane. As a result, the point of greatest discordance, $(1,1)$, will be the farthest from the origin of the axes. Conversely, $(0,0)$ represents the point of greatest agreement. The red segments in Fig. 5.1 are proportional to the values $h(p_{ij},a_{ij})$ for each $p_{ij},a_{ij} \in \{0,0.5,1\}$.

Thus, the aggregated distance $D^2_{0.5}\Big((\pi_1,A_1),(\pi_2,A_2)\Big)$ (see Eq. (5.9)) can be interpreted as the sum of $\frac{m\cdot(m-1)}{2}$ Euclidean distances in the preference-approval plane. That is,

$$D^2_{0.5}\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = \sqrt{0.5}\cdot \sum_{\substack{i,j=1 \\ i<j}}^{m} d\Big((p_{ij},a_{ij}),(0,0)\Big).$$

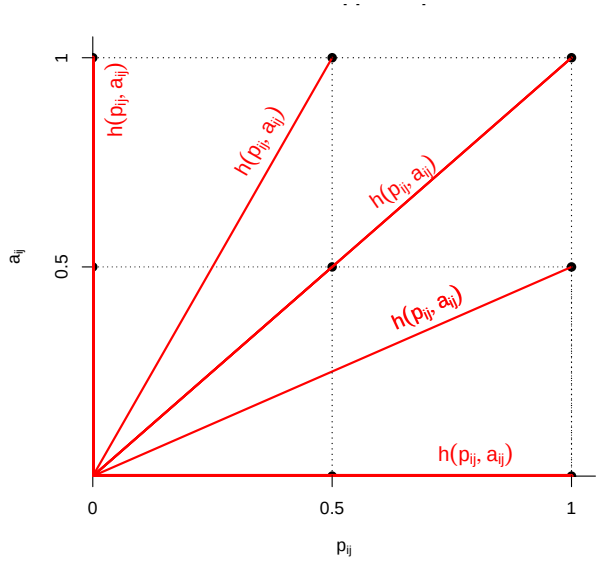


Figure 5.1: Preference-approval plane.

Proposition 1 *D^r_λ is a metric on $\mathscr{R}(Y)$ for all $\lambda \in (0,1)$ and $r \geq 1$. That is, for all $(\pi_1,A_1),(\pi_2,A_2) \in \mathscr{R}(Y)$ the following conditions are satisfied*[4]*:*

1. Positivity*: $D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) \geq 0$.*
2. Symmetry*: $D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = D^r_\lambda\Big((\pi_2,A_2),(\pi_1,A_1)\Big)$.*
3. Identity of indiscernibles*: $D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = 0 \Leftrightarrow (\pi_1,A_1) = (\pi_2,A_2)$.*
4. Triangle inequality*: $D^r_\lambda\Big((\pi_1,A_1),(\pi_3,A_3)\Big) \leq D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) + D^r_\lambda\Big((\pi_2,A_2),(\pi_3,A_3)\Big)$, for every $(\pi_3,A_3) \in \mathscr{R}(Y)$.*

The proof can be found in the Appendix C.1.

Remark 5 If $\lambda \in \{0,1\}$, then D^r_λ is not a metric.
If $\lambda = 0$, let $(\pi_1,A_1),(\pi_2,A_1) \in \mathscr{R}(Y)$ be such that $\pi_1 \neq \pi_2$. Then, we have $D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_1)\Big) = 0$.
If $\lambda = 1$, let $(\pi_1,A_1),(\pi_1,A_2) \in \mathscr{R}(Y)$ be such that $A_1 \neq A_2$. Then, we have $D^r_\lambda\Big((\pi_1,A_1),(\pi_1,A_2)\Big) = 0$.
Consequently, if $\lambda \in \{0,1\}$, then D^r_λ does not verify the identity of indiscernibles, hence it is not a metric.

[4] If $0 < r < 1$, then D^r_λ reduces to a distance since the triangle inequality does not hold

Proposition 2 demonstrate that our proposal can be considered as a generalization of the preference-approval distance proposed by Erdamar et al. (2014).
Given two preference-approvals $\Big((\pi_1,A_1),(\pi_2,A_2)\Big) \in \mathscr{R}(Y)$, its distance, $d_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big)$, is generated from the preference distance and the approval distance marginally, and eventually aggregate them by a convex combination.
The authors measure the disagreement between preferences by using the Kemeny metric (Kemeny, 1959), d_K:

$$d_K(\pi_1,\pi_2) = \sum_{\substack{i,j=1 \\ i<j}}^{m} |\operatorname{sgn}\big(P_{\pi_1}(y_j) - P_{\pi_1}(y_i)\big) - \operatorname{sgn}\big(P_{\pi_2}(y_j) - P_{\pi_2}(y_i)\big)|.$$

Or, equivalently, by considering the *Score matrix* Eq. (2.2), as defined in Chapter 1, Eq.(3.2):

$$d_K(\pi_1,\pi_2) = \sum_{\substack{i,j=1 \\ i<j}}^{m} |O_{\pi_1}(y_i,y_j) - O_{\pi_2}(y_i,y_j)|.$$

Notice that $d_K(\pi_1,\pi_2) \in [0, m\cdot(m-1)]$.
In turn, the approval disagreement is measured through the Hamming metric (Hamming, 1950), d_H:

$$d_H(A_1,A_2) = \sum_{i=1}^{m} |I_{A_1}(y_i) - I_{A_2}(y_i)|. \tag{5.10}$$

Notice that $d_H(A_1,A_2) \in [0,m]$.
In order to aggregate d_K and d_H as a global distance, the two metrics are normalized to the same codomain $[0,1]$ via dividing by their maximum distances.
The mappings $d_R : \mathscr{R}(Y)\times\mathscr{R}(Y) \longrightarrow [0,1]$ and $d_A : \mathscr{R}(Y)\times\mathscr{R}(Y) \longrightarrow [0,1]$ are defined as

$$d_R\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = \frac{d_K(\pi_1,\pi_2)}{m\cdot(m-1)},$$

$$d_A\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = \frac{d_H(A_1,A_2)}{m}.$$

The two normalized distances are eventually aggregated in a final preference-approval distance, $d_\lambda : \mathscr{R}(Y)\times\mathscr{R}(Y) \longrightarrow [0,1]$, defined as

$$\begin{aligned} &d_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = \\ &\lambda\cdot d_R\Big((\pi_1,A_1),(\pi_2,A_2)\Big) + (1-\lambda)\cdot d_A\Big((\pi_1,A_1),(\pi_2,A_2)\Big), \end{aligned} \tag{5.11}$$

where $\lambda \in [0,1]$ is a parameter used to control the relative relevance of the two components.
Taking into account Eqs. (3.2) and (5.10), Eq. (5.11) can be re-written as

$$d_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = \frac{\lambda}{m\cdot(m-1)}\cdot\sum_{\substack{i,j=1\\ i<j}}^{m}|O_{\pi_1}(y_i,y_j)-O_{\pi_2}(y_i,y_j)| + \frac{1-\lambda}{m}\cdot\sum_{i=1}^{m}|I_{A_1}(y_i)-I_{A_2}(y_i)|. \tag{5.12}$$

Proposition 2 *For all* $(\pi_1,A_1),(\pi_2,A_2)\in\mathscr{R}(Y)$ *and* $\lambda\in[0,1]$ *it holds*

$$D^1_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = d_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big).$$

The proof is given in Appendix C.2. Note that Proposition 2 is valid for weighted power means. They are the proper weighted quasiarithmetic means that allow us to generalize the distance between preference-approvals introduced by Erdamar et al. (2014).
In Proposition 2, we have shown that $D^r_\lambda = d_\lambda$ when $r=1$. We now show that is not true if $r\neq 1$.

Proposition 3 *If* $r\neq 1$, $D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = d_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big)$ *for all* $(\pi_1,A_1),(\pi_2,A_2)\in\mathscr{R}(Y)$ *and* $\lambda\in[0,1]$ *is not true.*

PROOF: Let us consider the case of two alternatives. Notice that in Eq. (5.9), when $m=2$, D^r_λ reduces to the h function computed in $i=1$ and $j=2$. That is, $D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = h(p_{12},a_{12}) = (\lambda\cdot p^r_{12}+(1-\lambda)\cdot a^r_{12})^{\frac{1}{r}}$. By Proposition 2, we have $D^1_\lambda\Big(\pi_1,A_1),\pi_2,A_2)\Big) = d_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = \lambda\cdot p_{12}+(1-\lambda)\cdot a_{12}$.
If we force the equality $D^1_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big)$, we have $\lambda\cdot p_{12}+(1-\lambda)\cdot a_{12} = (\lambda\cdot p^r_{12}+(1-\lambda)\cdot a^r_{12})^{\frac{1}{r}}$, i.e.,

$$(\lambda\cdot p_{12}+(1-\lambda)\cdot a_{12})^r = \lambda\cdot p^r_{12}+(1-\lambda)a^r_{12}. \tag{5.13}$$

We have to prove that there exist $p_{12},a_{12}\in\{0,0.5,1\}$ and $\lambda\in[0,1]$ such that Eq. (5.13) is not true for any $r\neq 1$.
If $p_{12}=1$ and $a_{12}=0$, then Eq. (5.13) becomes $\lambda^r=\lambda$, and it is true if and only if $\lambda\in\{0,1\}$. In all the other cases, if $r\neq 1$, then Eq. (5.13) is false. ■

5.4 Clustering tasks

This section shows how the proposed distance can be used to study the universe of preference-approvals and to determine clusters.
Subsection 5.4.1 examines the universe of preference approvals in the case of two alternatives in order to observe how the values of r and λ affect the creation of homogeneous clusters. Afterwards, the influence of the two parameters r and λ when the number of alternatives n varies is investigated.
Subsection 5.4.2 provides an application on real data, to investigate how the countries of the European Union can be clustered into groups, according to their preference-approvals on nine alternatives concerning social values. The dataset used comes from the Eurobarometer website[5].

5.4.1 Universe of preference-approvals

Let us consider the 2-dimensional preference-approval universe where the set of alternatives is $Y = \{y_1, y_2\}$. Following Eq. (5.3), the preference-approvals (π_i, A_i), $i = 1, 2, \ldots, 8$, are represented by two 2-dimensional vectors:

$$(2,1)\,\&\,(1,1) \equiv \begin{array}{c} y_2 \\ y_1 \\ \hline \end{array} \qquad (2,1)\,\&\,(0,1) \equiv \begin{array}{c} y_2 \\ \hline y_1 \end{array}$$

$$(2,1)\,\&\,(0,0) \equiv \begin{array}{c} \\ \hline y_2 \\ y_1 \end{array} \qquad (1,2)\,\&\,(1,1) \equiv \begin{array}{c} y_1 \\ y_2 \\ \hline \end{array}$$

$$(1,2)\,\&\,(1,0) \equiv \begin{array}{c} y_1 \\ \hline y_2 \end{array} \qquad (1,2)\,\&\,(0,0) \equiv \begin{array}{c} \\ \hline y_1 \\ y_2 \end{array}$$

$$(1.5,1.5)\,\&\,(1,1) \equiv \begin{array}{c} y_1\ y_2 \\ \hline \end{array} \qquad (1.5,1.5)\,\&\,(0,0) \equiv \begin{array}{c} \\ \hline y_1\ y_2 \end{array}$$

The distances between preference-approvals on two alternatives for $r = 1$ and $\lambda = 0.5$

[5] https://europa.eu/eurobarometer/screen/home.

(Fig. 5.2) and $\lambda = 0.75$ (Fig. 5.3) are reported in the heatmaps.

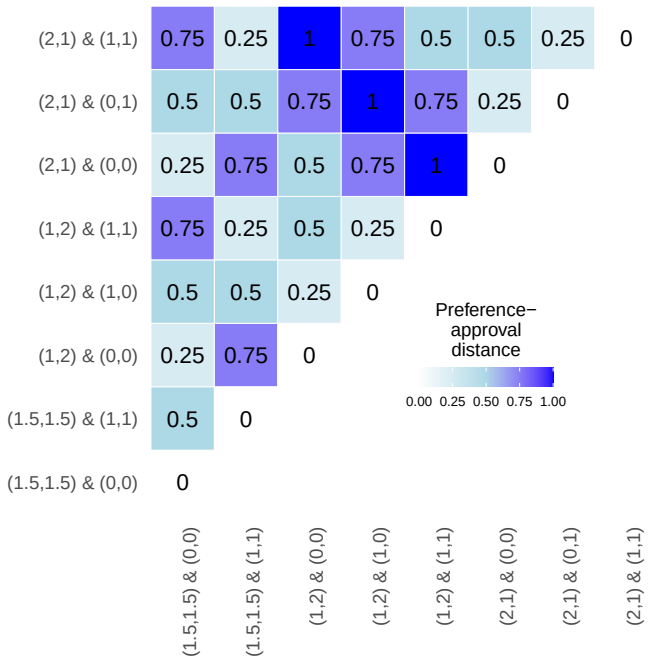


Figure 5.2: Distances between preference-approvals for 2 alternatives, $r = 1$ and $\lambda = 0.5$.

Increasing the value of λ emphasizes the discordance in the preference part, and modifies the relationships between the corresponding preference-approvals. Indeed, when $\lambda = 0.75$, there is an increase in the intensity of the distances at the top-right hand side of the graph, which concerns the triples

$$(2,1)\,\&\,(1,1),\ (2,1)\,\&\,(0,1),\ (2,1)\,\&\,(0,0)$$

and

$$(1,2)\,\&\,(0,0),\ (1,2)\,\&\,(1,1),\ (1,2)\,\&\,(1,1).$$

The hierarchical relationship between objects is reported in Fig. 5.4; the dendrograms show how the hierarchical clustering of the eight preference-approvals changes based on D^r_λ.
Fig. 5.4 shows that the value of λ strongly influences the hierarchical aggregation of preference-approvals. A similar analysis can be carried out by varying the value of r. In Fig. 5.5 the distances between the corresponding preference-approvals, for $r = 2$

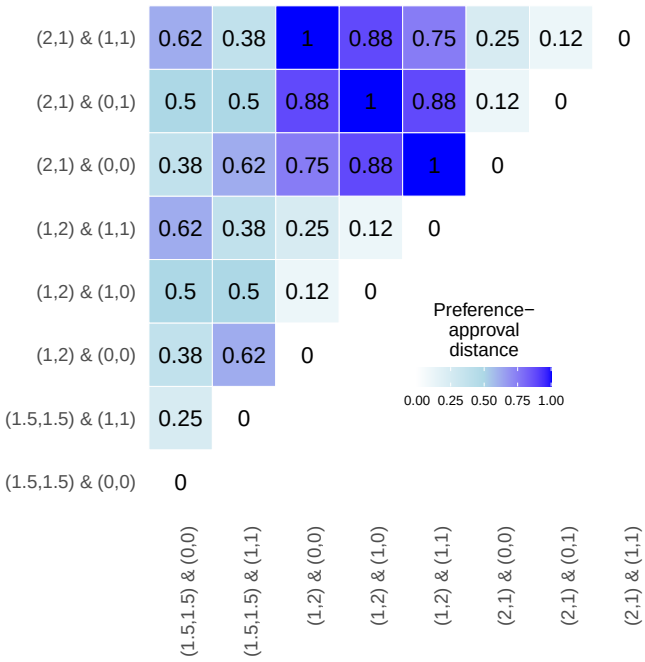


Figure 5.3: Distances between preference-approvals for 2 alternatives, $r = 1$ and $\lambda = 0.75$.

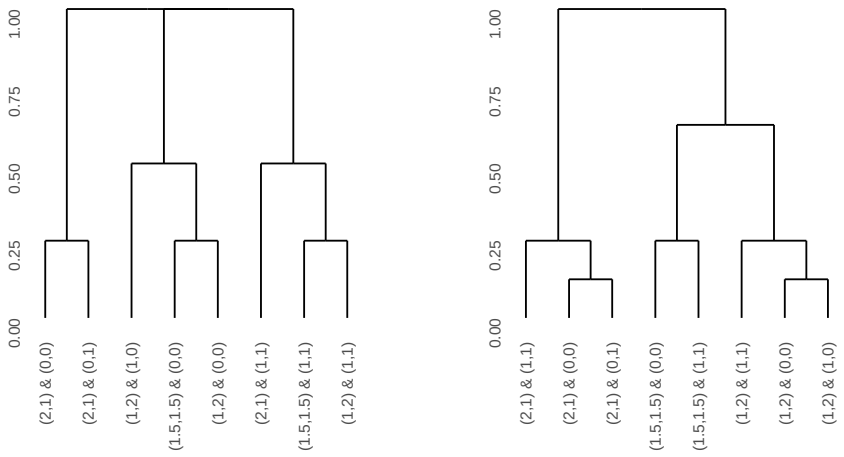


Figure 5.4: Hierarchical clustering dendrogram for 2 alternatives, $r = 1$, $\lambda = 0.5$ (left) and $\lambda = 0.75$ (right).

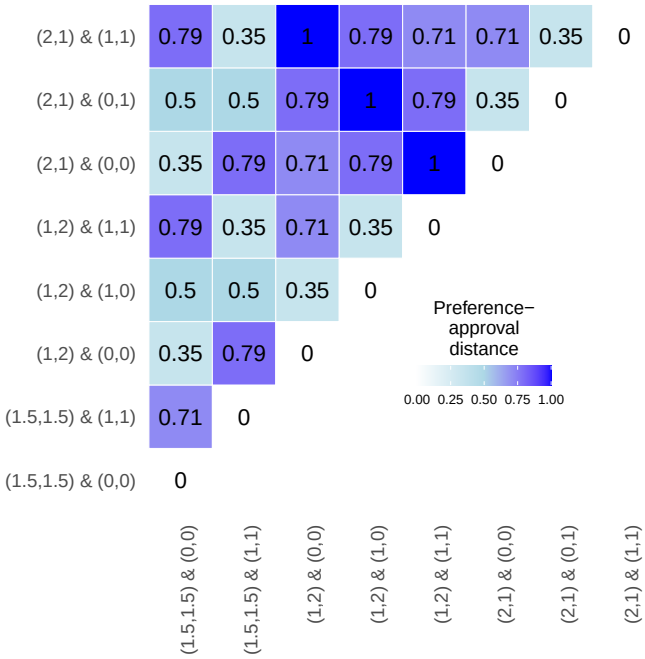


Figure 5.5: Distance between preference-approvals for 2 alternatives, $r = 2$, $\lambda = 0.5$

and $\lambda = 0.5$ are shown.

Compared to Fig. 5.2, Fig. 5.5 shows a general increase of distances determined by the increase of r. In particular,

$$D^2_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) \geq D^1_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big),$$

for all $(\pi_1,A_1),(\pi_2,A_2) \in \mathscr{R}(Y)$. This is due to h being increasing in r.

The dendrograms between preference-approvals objects are reported in Fig. 5.6.

Fig. 5.6 shows that an increase in r contributes differently (with respect to an increase in λ) to the change of the hierarchical aggregation structure. In fact, the two dendrograms merge preference-approvals in the same way. What changes is the "height" at which there is the aggregation or, in other words, the distance to be tolerated to aggregate two preference-approvals. Note that this happens only for two alternatives.

Tables 5.6, 5.7, 5.8 and 5.9 show the cophenetic correlation coefficient[6] (see Sokal and

[6]The cophenetic correlation coefficient is a measure of similarity between dendrograms. It is particularly used in biostatistics to investigate how faithfully a dendrogram preserves the pairwise distances between the original unmodeled data points, or also to study where raw data tend to occur in clumps or clusters. This coefficient has also been proposed as a nested cluster test (see Rohlf and Fisher 1968 and Saraçli et al. 2013).

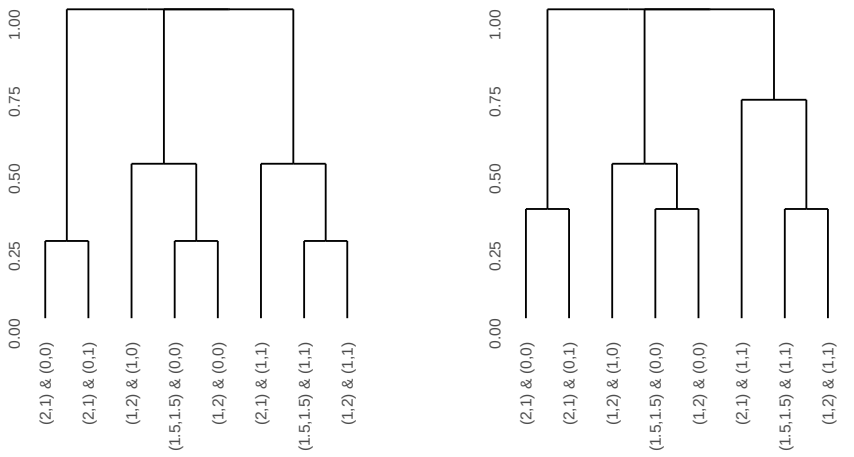


Figure 5.6: Hierarchical clustering dendrogram for 2 alternatives, $r = 1$ (left), $r = 2$ (right) and $\lambda = 0.5$.

Rohlf 1962 and Schlee 1973, pp. 278-284) between dendrograms, for $m = 2, 3, 4, 5$ and $\lambda = 0.5$. The cophenetic coefficient was computed in `R` using the `dendextend` package (Galili, 2015).

$m = 2$	1	1.5	2	5	10
1	1				
1.5	0.99	1			
2	0.98	1	1		
5	0.94	0.97	0.98	1	
10	0.90	0.94	0.97	1	1

Table 5.6: Cophenetic dendrogram correlations for $m = 2$, $r = 1, 1.5, 2, 5, 10$ and $\lambda = 0.5$.

$m = 3$	1	1.5	2	5	10
1	1				
1.5	0.76	1			
2	0.76	1	1		
5	0.63	0.68	0.70	1	
10	0.65	0.71	0.73	0.99	1

Table 5.7: Cophenetic dendrogram correlations for $m = 3$, $r = 1, 1.5, 2, 5, 10$ and $\lambda = 0.5$.

$m=4$	1	1.5	2	5	10
1	1				
1.5	0.85	1			
2	0.80	0.83	1		
5	0.71	0.76	0.88	1	
10	0.70	0.75	0.86	0.99	1

Table 5.8: Cophenetic dendrogram correlations for $m = 4$, $r = 1, 1.5, 2, 5, 10$ and $\lambda = 0.5$.

$m=5$	1	1.5	2	5	10
1	1				
1.5	0.80	1			
2	0.72	0.79	1		
5	0.61	0.73	0.81	1	
10	0.57	0.69	0.79	0.95	1

Table 5.9: Cophenetic dendrogram correlations for $m = 5$, $r = 1, 1.5, 2, 5, 10$ and $\lambda = 0.5$.

Tables 5.6, 5.7, 5.8 and 5.9 show that dendrogram correlations are strictly related to the values of r and m. Overall, the correlations between dendrograms tend to decrease as r increases. This is especially evident when we examine the first column of each table, which reports the correlation between dendrograms obtained with $r = 1$ and dendrograms obtained with $r = 1.5, 2, 5, 10$. In terms of the number of alternatives, it should be noted that as m increases, the dendrogram correlations generally decrease with an oscillatory trend.

In other words, Tables 5.6, 5.7, 5.8 and 5.9 highlight that the parameter r has a considerable influence, not only on the resulting values of the proposed distance D^r_λ, but also on the cluster structure discovered among the observations of the preference-approvals universe. Specifically, as m increases and the expressiveness of the voters explodes (Table 5.1), so does the discriminating power of r, allowing different clustering structures to be highlighted. Indeed, the proposed family of distances D^r_λ is more flexible than the existing one, and it ultimately comes down to a new parameter that can be exploited in various applications, such as maximizing the goodness of a clustering procedure.

To explore further this issue, let us consider a simulation study on the universe of 5 alternatives, which involves three steps:

1. generate four groups of clustered preference-approvals;

2. apply a hierarchical clustering algorithm for different values of r.

3. compute an external validation index, the Adjusted Rand index (Hubert and Arabie, 1985), to investigate which value of r maximises the similarity between the estimated and the theoretical clusters.

Therefore, we aim to find the value of r that provides more reliable clusters, i.e. clusters that are more consistent with the data-generating process.
The number of preference-approvals (on five alternatives) generated within each cluster was determined by randomly drawing four values from a normal distribution $\mathcal{N}(50,4)$ and converting them into integer numbers.
Orderings and approvals were generated individually and merged to produce the final set of preference-approvals. Specifically, orderings within each sub-partition were generated from a Mallows Model (Mallows, 1957), already introduced in Chapter 4. The θ values for our simulation studies are $\{0, 0.5, 1, 1.5, 2\}$. Assuming that π is a generic ranking, the probability for this ranking is function of θ, and it is given by:

$$Pr(\theta) = \frac{\exp\big(-\theta d(\pi, \pi_0)\big)}{\psi(\theta)}, \tag{5.14}$$

where d is a ranking distance measure and $\psi(\theta)$ is a normalization constant.
We generated rankings assuming the Kemeny distance d_K. The cluster central permutations, π_0, used in the analysis are reported in Tab. 5.10.

Cluster k	Central permutation π_0
1	(4.5, 2, 4.5, 3, 1)
2	(1, 3, 4, 2, 5)
3	(4, 3, 1.5, 1.5, 5)
4	(3, 5, 2, 4, 1)

Table 5.10: Cluster central permutations.

Approvals, within each cluster are generated from four multinomial distributions, with probability vectors, p_{ik}, described in Tab. 5.11. Specifically, p_{ik} is the probability to draw i approved alternatives into the k-th cluster.
After deriving clusters, the adjusted Rand index (Hubert and Arabie, 1985) is used to assess their goodness. The adjusted Rand index is a measure of the similarity between two sets of clusterings. It is the corrected-for-chance version of the Rand index (Rand, 1971). The correction uses the predicted similarity of all pair-wise comparisons between clusterings described by a random model to generate a baseline. Although the

Cluster k	Approved alternatives i					
	0	1	2	3	4	5
1	0.10	0.40	0.25	0.15	0.05	0.05
2	0.10	0.10	0.30	0.30	0.15	0.05
3	0	0.05	0.10	0.25	0.35	0.25
4	0.35	0.30	0.15	0.10	0.05	0.05

Table 5.11: Multinomial probability vectors.

Rand Index can only provide values between 0 and +1 (0 when the two data clusterings do not agree on any pair of points, and 1 when data clusterings are exactly the same), the modified Rand Index can return negative values if the index is lower than the expected similarity of all pair-wise comparisons between clusterings specified by a random model.

The results (Table 5.12) are obtained by averaging the adjusted Rand index over ten randomly generated datasets for each value of θ.

		θ				
		0	0.5	1	1.5	2
	1	0.093	0.267	0.591	**0.847**	0.801
	0.5	**0.114**	0.249	0.435	0.692	**0.822**
	2	0.080	**0.320**	**0.611**	0.665	0.602
r	3	0.090	0.274	0.568	0.607	0.559
	4	0.082	0.313	0.569	0.588	0.530
	5	0.081	0.301	0.558	0.553	0.502
	7	0.089	0.273	0.534	0.542	0.502
	10	0.074	0.276	0.528	0.540	0.502

Table 5.12: Average adjusted Rand index over r and θ.

Table 5.12 shows that, except for the case $\theta = 1.5$, our measure D_λ^r with $r \neq 1$ results in higher average adjusted Rand indices. Thus, $r \neq 1$ allows the true clustered structure of data to be found more accurately and provides more accurate clusters.

5.4.2 A real data application

This subsection shows how the proposed metric can be used to perform cluster analysis on real data retrieved from the Eurobarometer website.

Eurobarometer is a collection of cross-country public opinion surveys conducted on behalf of the European Commission and other European Union (EU) institutions since 1973. These polls address a wide range of issues pertaining to the EU and its member countries. The data utilized in these analyses are specifically from question Q5 of the poll titled "Defending Democracy, Empowering citizens. Public Opinion at the legislature's midpoint"[7].
A group of voters, divided by countries, was asked to indicate which of the following values should the European Parliament defend as a matter of priority:

- y_1: Equality between women and men.
- y_2: The fight against discrimination and for the protection of minorities.
- y_3: Tolerance and respect for diversity in society.
- y_4: Solidarity between EU Member States and between its regions.
- y_5: Solidarity between the EU and poor countries in the world.
- y_6 The protection of human rights in the EU and worldwide.
- y_7: Freedom of religion and belief.
- y_8: Freedom of movement.
- y_9: Freedom of speech and thought.

As a result, data are stored in a table (see Tab. 5.14) with 27 rows (one row for each EU member country) and 9 columns (each column representing an alternative of $Y = \{y_1, \ldots, y_9\}$). The total number of votes cast by the i-th country in favour of the j-th alternative is shown in the table's generic cell ij.
In order to transform the original table into a set of preference-approvals, preferences and approvals need to be derived. For each country, the alternatives are ranked in order of popularity, beginning with the one that received the most votes and ending with the one that received the fewest. Furthermore, in order to generate a vector of approvals, those alternatives that received more votes than the national average were deemed acceptable.
For example, in Tab. 5.13 we show the votes expressed in France (the votes of all countries are included in Tab. 5.14).

[7] https://europa.eu/eurobarometer/surveys/detail/2612.

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9
France	37	17	20	12	13	20	12	12	32

Table 5.13: Votes in France.

Since the votes' average is 19.44, the votes in France are transformed into a preference-approval codification (see Eq. (5.3)) as

$$(1,5,3.5,8,6,3.5,8,8,2)\,\&\,(1,0,1,0,0,1,0,0,1)$$

that can be visualized as follows:

$$\begin{array}{c} y_1 \\ y_9 \\ y_3\ y_6 \\ \hline y_2 \\ y_5 \\ y_4\ y_7\ y_8. \end{array}$$

To run the cluster analysis, the distance matrix 27×27 was constructed using Eq. (5.9). All the alternatives seem important in this example, so a distinction between acceptable and unacceptable alternatives should not be interpreted as a distinction between valuable and not valuable, but instead as a distinction between more and less urgent. For this reason, $\lambda = 0.75$ was chosen in order to emphasize preference differences more than approvals.

A cluster-wise measure of cluster stability (Hennig, 2007) is used to jointly discover the optimal value of r and the optimal number of clusters k. Stability refers to the property of a meaningful and valid cluster that does not change easily when the data set is perturbed in a non-essential way. That is, when applied to many datasets collected from the same data distribution, a reliable clustering method should produce similar partitions. The cluster stability method (Hennig, 2007) employs three steps:

1. use various strategies to resample new data sets from the original and apply the hierarchical clustering method to each of them;
2. for every given original cluster, find the most similar cluster using the Jaccard coefficient (Jaccard, 1901) in the new data set and record the similarity value;
3. assess the cluster stability of every single cluster by the mean similarity taken over the resampled data sets.

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9
Belgium	31	15	16	23	12	24	10	14	34
Bulgaria	11	7	14	24	10	23	11	32	21
Czech Republic	12	6	10	22	8	28	4	23	25
Denmark	18	14	17	13	10	30	7	12	27
Germany	15	11	18	21	9	32	6	8	28
Estonia	11	12	14	14	5	20	8	29	24
Ireland	27	19	15	14	9	28	11	27	24
Greece	9	12	8	34	19	31	8	14	35
Spain	35	15	18	15	15	20	4	12	25
France	37	17	20	12	13	20	12	12	32
Croatia	14	14	14	20	16	25	12	28	28
Italy	25	17	14	21	11	20	9	21	29
Cyprus	25	14	5	24	21	37	10	11	24
Latvia	5	14	11	33	5	35	4	20	26
Lithuania	10	12	17	17	8	33	7	18	29
Luxembourg	23	19	17	18	13	19	7	16	22
Hungary	15	17	15	19	11	28	12	20	21
Malta	21	17	15	16	13	28	10	15	18
Netherlands	25	18	25	18	9	34	12	6	31
Austria	24	16	19	19	12	23	9	19	30
Poland	15	15	13	18	11	19	12	24	19
Portugal	32	22	16	30	20	27	7	5	17
Romania	14	12	12	20	16	24	14	28	22
Slovenia	15	8	23	19	9	32	5	24	31
Slovakia	19	10	10	20	9	21	15	35	28
Finland	15	14	15	14	6	30	7	17	26
Sweden	33	12	19	12	11	39	5	12	30

Table 5.14: Votes in the EU.

The average cluster-wise stability is shown in Fig. 5.7 as a function of r (for $k = 2,3,4$ clusters). The procedure suggests that the most stable cluster configuration is $k = 2$ and $r = 2$. It is worth noting that, regardless of the value of k, $r > 1$ always leads to improved cluster stability. Indeed, with two clusters ($k = 2$) the value of r that maximizes stability is $r = 2$. Whereas with three or four clusters, the optimal solution is $r = 4$. In addition, as the number of clusters k increases, the average stability decreases.

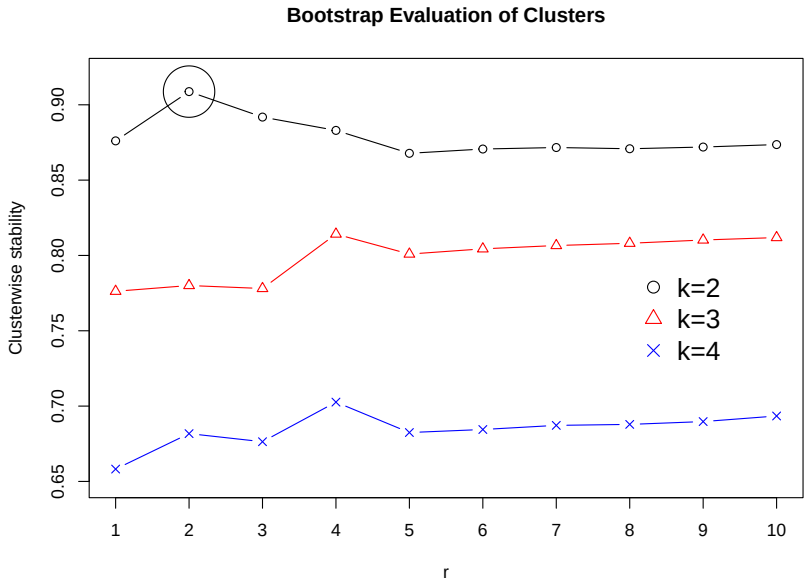


Figure 5.7: Average cluster-wise stability over r.

For several reasons, stability is a particularly relevant cluster validation measure in this example for determining the best value of r. First, it is not possible to use external validation measures in this case as the true clustered structure of the EU countries is not known. At the same time, most internal validation measures employ the distance between observations (D_λ^r) to assess the goodness of clusters. However, this may be an issue in our instance since the distance between observations (D_λ^r) is influenced by r. Therefore, to determine which value of r yields more accurate clusters, a metric that is independent of r is desirable. Furthermore, cluster stability has been examined both theoretically and practically (Hennig, 2007; Von Luxburg, 2010; Ullmann et al., 2022), and it has been shown to be capable of distinguishing between meaningful stable and spurious clusters.

Figures 5.8 and 5.9 show the resulting dendrogram and clusters, respectively, obtained with $k = 2$ and $r = 2$.

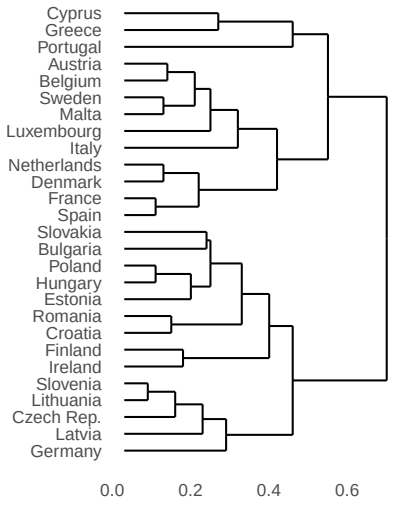


Figure 5.8: EU cluster dendrogram.

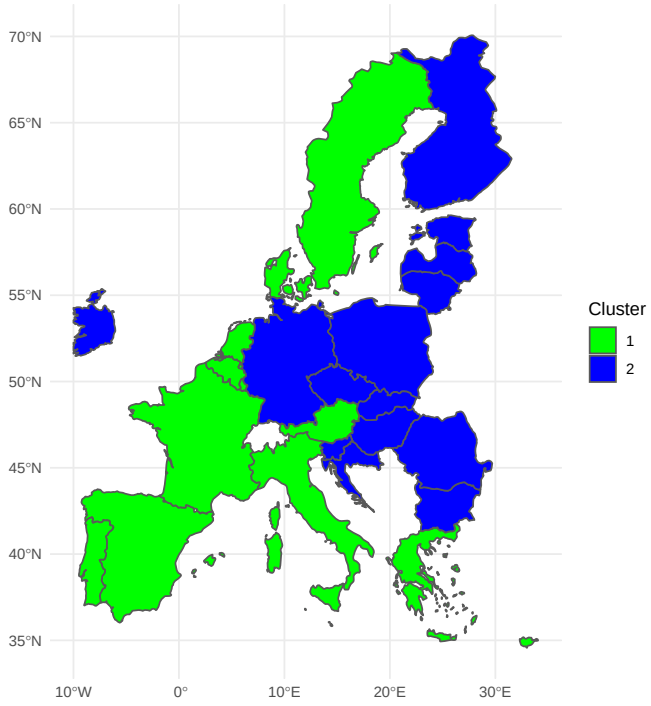


Figure 5.9: Map of EU voters with clusters.

The clustering procedure suggests that the EU countries can be separated into two large groups. Cluster 1 is mainly made up of Western European countries, whereas Cluster 2 of Eastern European countries.

To provide a more in-depth picture of how the EU countries express their views on the nine alternatives proposed, the two preference-approvals that represent the two clusters, that we call *representative preference-approvals*, are shown in Eq. (5.15).

To obtain the representative preference-approvals that summarize each cluster, preferences and approvals need to be aggregated. In each cluster, the set of preferences is combined into a unique weak order by deriving the average position for each alternative and ranking them according to it. Note that this aggregation method is equivalent to the Borda count (Borda, 1781) extended to weak orders (see Smith, 1973, Black, 1976 and Cook and Seiford, 1982).

In our example, the extended Borda count assigns a score to each alternative; for each country, the number of alternatives ranked below plus half of the number of alternatives that are indifferent to it:

$$B_R(y_i) = \#\left\{y_j \in Y \mid y_i \succ y_j\right\} + \frac{1}{2} \cdot \#\left\{y_j \in Y \setminus \{y_i\} \mid y_i \sim y_j\right\}.$$

Similarly, the set of approvals is combined into a unique approval vector by taking the average approval for each alternative and then considering those alternatives whose average approval is greater than 0.5 as approved:

$$B_I(y_i) = I\left(\sum_{l=1}^{m} I_{A_l}(y_i) \geq m/2\right)$$

Cluster 1	Cluster 2
$y_1\, y_9$	y_6
y_6	y_9
———	y_8
y_4	y_4
y_2	———
y_3	$y_1\, y_3$
y_8	y_2
y_5	y_5
y_7	y_7

(5.15)

It is worth noting that y_6 and y_9, namely, "The protection of human rights in the EU and worldwide" and "Freedom of speech and thought", respectively, are above the

approval line in the two representative preference-approvals, indicating that they can be considered very urgent. Regarding y_1, that is "Equality between women and men", it is ranked at the top of the representative preference-approval of Cluster 1, while it is just below the approval line in the Cluster 2 representative preference-approval. Similarly, y_4, that is "Solidarity between the EU Member States and between its regions", is ranked fourth (above the approval line) in Cluster 2. Still, it is the first alternative below the approval line in Cluster 1. Furthermore, Cluster 2 prioritizes y_8, that is "Freedom of movement", which is at the end of the preference-approval of Cluster 1. Finally, in both the two representative preference-approvals, y_7, that is "Freedom of religion and belief", is ranked last.

Table 5.15 reports the $D^2_{0.75}$ distances of each country to the representative cluster preference-approvals.

It should be noted that, except for Greece, each country is closer to the preference-approval of its own cluster than the other. Despite being reasonable, this result is not trivial since the technique for obtaining the cluster preference-approval does not involve D^r_λ.

Some countries can be considered central in their clusters as they are very close to the representative preference-approval, e.g. Belgium (0.092), Austria (0.096), Malta (0.096) for Cluster 1, and the Czech Republic (0.036), Lithuania (0.094), Hungary (0.072) and Slovenia (0.105) for Cluster 2. As a rule of thumb, the greater the distance from the own cluster preference-approval, the more the country disagrees with the other countries in its cluster. Finally, it is worth noting that some countries, such as Ireland, Italy and Greece, are located in the middle of the two clusters, as they have a similar distances to the two cluster preference-approvals.

5.5 Concluding remarks

In social choice theory, preference rankings and approvals are two popular ways to collect the preferences of a group of agents on a set of alternatives. In the preference-approval setting, each agent, in addition to ordering a set of alternatives from best to worst, submits a cut-off line to distinguish between acceptable and unacceptable. Within this framework, in this chapter, we propose a new distance for preference-approvals, following the approach of the Kemeny distance. Given two preference-approvals and two alternatives, we introduce two indices that measure the discordances between these alternatives with respect to preference and approvals, and an aggregation function belonging to the class of weighted power means to define a new distance.

Country	Cluster 1	Cluster 2	Cluster assignment
Belgium	**0.092**	0.246	1
Bulgaria	0.487	**0.192**	2
Czech Rep.	0.342	**0.036**	2
Denmark	**0.176**	0.332	1
Germany	0.286	**0.204**	2
Estonia	0.355	**0.195**	2
Ireland	0.267	**0.265**	2
Greece	0.388	**0.318**	1
Spain	**0.164**	0.419	1
France	**0.219**	0.440	1
Croatia	0.387	**0.156**	2
Italy	**0.200**	0.225	1
Cyprus	**0.281**	0.370	1
Latvia	0.341	**0.120**	2
Lithuania	0.400	**0.094**	2
Luxembourg	**0.144**	0.359	1
Hungary	0.343	**0.072**	2
Malta	**0.096**	0.301	1
Netherlands	**0.243**	0.377	1
Austria	**0.096**	0.262	1
Poland	0.349	**0.132**	2
Portugal	**0.367**	0.537	1
Romania	0.463	**0.204**	2
Slovenia	0.415	**0.105**	2
Slovakia	0.354	**0.219**	2
Finland	0.320	**0.123**	2
Sweden	**0.144**	0.274	1

Table 5.15: Distance between countries and representative cluster preference-approvals.

This new distance depends on two parameters. The effect of these parameters on the distance is analyzed and described through some heatmaps. The proposed distance can be used to study the universe of preference-approvals and to determine clusters of voters: how the two parameters characterizing the distances affect the clustering process is shown with some dendrograms and by the cophenetic correlations among them. We have shown that the new distance family offers some advantages compared to the existing distance function. Specifically, through a simulation study and the adjusted Rand index, we have proved that D_λ^r with $r \neq 1$ allows the true clustered structure of data

to be found more accurately. Similarly, through a cluster-wise stability index, we have shown that D^r_λ with $r \neq 1$ produces more stable clusters on the real data example. The proposed distance will be used in future work to apply fuzzy clustering algorithms to deal with voters who do not have clear cluster assignments.

In future work, axiomatizing the new family of distance functions might prove important. Finally, future research should examine consensus measures based on distances between preference-approvals (see Erdamar et al. 2014), algorithms to determine representative preference-approvals efficiently (see D'Ambrosio et al. 2017b), clustering on alternatives (see González del Pozo et al. 2017), and also reaching consensus processes (see Palomares et al. 2014; García-Lapresta and Pérez-Román 2017; Chao et al. 2021, among others).

Chapter 6

A new pseudometric for clustering alternatives in preference-approvals

6.1 Introduction

Preference-approval structures have been studied recently from different perspectives in Erdamar et al. (2014); Kamwa (2019); Dong et al. (2021); Kruger and Sanver (2021); Long et al. (2021); Barokas and Sprumont (2022). However, little effort has been devoted to developing clustering algorithms that deal with preference-approvals. The clustering task deals with classifying objects in homogeneous clusters, such that objects in a cluster are more similar to each other than they are to an object belonging to a different cluster (see Jain et al. 1999 and Everitt et al. 2011). To the best of our knowledge, the only proposal applying clustering algorithms to preference-approval structures is found in Albano et al. (2022a). They introduced a family of distances between preference-approvals and used a hierarchical clustering algorithm to find homogeneous groups of individuals. The possibility of clustering alternatives in preference-approvals has not yet been addressed. In this paper, we aim to fill this gap since we argue that identifying homogeneous groups of alternatives could be beneficial to reducing the complexity of the preference-approval space and provide a more accessible interpretation of data. In other words, considering a set of voters expressing their preference-approval on a finite set of alternatives, $Y = \{y_1, \ldots, y_m\}$, we aim at devel-

oping a method able to split Y into k groups, solely based on the voters' opinions.

Although the literature on clustering algorithms applied to preference orderings is rich, it is not straightforward to transfer it directly to the preference-approval framework because preference-approvals are more complex structures. Clustering methods for preference rankings can be done over the individuals or over the alternatives.
Most popular classification approaches for clustering individuals, use an algorithm model (e.g., hierarchical clustering, tree construction) or attempt to maximize some badness-of-fit function (e.g., K-means, fuzzy clustering, PCA, MDS). On this, see Heiser and D'Ambrosio (2013, pp. 19-31).
Alternatively, probabilistic methods model the population of rankers assuming homogeneity between them. Paired comparison models (Kendall and Smith, 1940; Mallows, 1957) consider a ranking as the outcome of a paired comparison process. Parsimoniously modelling each paired comparison leads to the famous Mallows model and its generalization to distance-based models (Fligner and Verducci, 1986). In this framework, Jacques and Biernacki (2014) proposed the first model-based clustering algorithm dedicated to multivariate partial ranking data that can take into account potentially missing positions (partial rankings), occurring not necessarily at the end of the rankings.
Despite being less studied, the task of clustering alternatives rather than individuals in preference rankings is undoubtedly relevant. Marden (1996) defined a distance between two alternatives as the squared Euclidean distance of the ranks assigned to them. Thus, objects will be close if the voters give them similar ranks. Finally, they applied a simple hierarchical clustering to find meaningful groups. Sciandra et al. (2020) proposed a projection pursuit-based clustering method to identify simultaneous clusters of both individuals and items in preference rankings.
Similarly to the task of clustering alternatives in preference rankings, González del Pozo et al. (2017) focused on ordered qualitative scales and developed an agglomerative hierarchical clustering procedure based on the concept of ordinal proximity measure. They clustered nine presidential candidates considering a similarity function and a sequential similarity vector based on the degrees of consensus. The consensus is measured through the degrees of proximity between all pairs of individual appraisals over the evaluated alternatives.

In this work, we introduce a new family of pseudometrics on the set of alternatives taking into account voters' opinions on these alternatives through preference-approvals. To obtain clusters, we apply an order-invariant partitioning algorithm, known as Ranked

k-medoids (RKM), see Zadegan et al. (2013), taking as input the similarities among pairs of alternatives based on the proposed pseudometrics. Finally, clusters are represented in 2-dimensional space using non-metric multidimensional scaling.
The chapter is organized as follows. Section 6.2 contains our proposal for clustering alternatives. Section 6.3 includes some case studies. Finally, Section 6.4 concludes the chapter with some remarks.

6.1.1 A pseudometric on preferences

We introduce a pseudometric on the set of alternatives that measures the difference between the positions of two alternatives in a weak order.

Proposition 4 *Given $R \in W(Y)$, the mapping $d_P : Y \times Y \longrightarrow \mathbb{R}$ defined as*

$$d_P(y_i, y_j) = |P_\pi(y_i) - P_\pi(y_j)| \tag{6.1}$$

is a pseudometric on Y, i.e., it satisfies the following conditions for all $y_i, y_j, y_k \in Y$:

1. $d_P(y_i, y_j) \geq 0$.
2. $d_P(y_i, y_i) = 0$.
3. $d_P(y_i, y_j) = d_P(y_j, y_i)$.
4. $d_P(y_i, y_j) \leq d_P(y_i, y_k) + d_P(y_k, y_j)$.

Additionally, it is satisfied $d_P(y_i, y_j) = 0 \Leftrightarrow y_i \sim y_j$, for all $y_i, y_j \in Y$.

Obviously, if $R \in L(Y)$, then d_P is a metric, i.e., $d_P(y_i, y_j) = 0 \Leftrightarrow y_i = y_j$, for all $y_i, y_j \in Y$.
Note that $d_P(y_i, y_j) \in \{0, 1, \ldots, m-1\}$ for all $y_i, y_j \in Y$.

6.1.2 A pseudometric on approvals

Given $A \subseteq Y$, the *indicator function* (or *characteristic function*) of A, $I_A : Y \longrightarrow \{0, 1\}$, is defined as

$$I_A(y_i) = \begin{cases} 1, \text{ if } y_i \in A, \\ 0, \text{ if } y_i \in Y \setminus A. \end{cases} \tag{6.2}$$

From Eq. (6.2), we now introduce a pseudometric on the set of alternatives that measures the difference between the membership of two alternatives in a set.

Proposition 5 *Given $A \subseteq Y$, the mapping $d_A : Y \times Y \longrightarrow \mathbb{R}$ defined as*

$$d_A(y_i, y_j) = |I_A(y_i) - I_A(y_j)| \tag{6.3}$$

is a pseudometric on Y, i.e., it satisfies the following conditions for all $y_i, y_j, y_k \in Y$:

1. $d_A(y_i, y_j) \geq 0$.
2. $d_A(y_i, y_i) = 0$.
3. $d_A(y_i, y_j) = d_A(y_j, y_i)$.
4. $d_A(y_i, y_j) \leq d_A(y_i, y_k) + d_A(y_k, y_j)$.

Additionally, it is satisfied $d_A(y_i, y_j) = 0 \Leftrightarrow \Big(y_i, y_j \in A \text{ or } y_i, y_j \notin A\Big)$, for all $y_i, y_j \in Y$.

Note that $d_A(y_i, y_j) \in \{0, 1\}$ for all $y_i, y_j \in Y$.

6.2 The proposal

Given a profile $\Big((\pi_1, A_1), \ldots, (\pi_m, A_m)\Big) \in \mathscr{R}(Y)^m$ and two alternatives $y_i, y_j \in Y$, we now introduce two indices that measure the discordances between these alternatives with respect to preference and approvals, respectively, for each voter $v_k \in V$. They are based on the pseudometrics introduced in Eqs. (6.1) and (6.3).

6.2.1 Preference discordances

The *individual preference-discordance* between y_i and y_j for the voter $v_k \in V$ is defined as

$$\rho_{ij}^{(k)} = \frac{1}{m-1} \cdot |P_{\pi_k}(y_i) - P_{\pi_k}(y_j)|. \tag{6.4}$$

where $\rho_{ij}^{(k)} \in [0, 1]$.

Remark 6 Note that if a voter expresses a linear order $\pi \in L(Y)$, then: i) there will not be any pair of alternatives whose individual preference-discordance is 0 and ii) there will be only one pair of alternatives whose individual preference-discordance is maximum, equal to 1:

$$\pi \in L(Y) \Rightarrow \begin{cases} \rho_{ij}^{(k)} \neq 0 \text{ for all } y_i, y_j \in Y \\ \exists!\, y_i, y_j \in Y \quad \rho_{ij}^{(k)} = 1. \end{cases}$$

On the contrary, if a voter expresses a weak order that is not a linear order $\pi' \in \Big(W(Y) \setminus L(Y)\Big)$, and indifference happens at the bottom or at the head of the weak order, then: i) there will exist at least a pair of alternatives whose individual preference-discordance is 0; ii) no pair of alternatives produces an individual preference-discordance equal to 1:

$$\pi' \in \Big(W(Y) \setminus L(Y)\Big) \Rightarrow \begin{cases} \exists\, y_i, y_j \in Y \quad \rho_{ij}^{(k)} = 0 \\ \rho_{ij}^{(k)} \neq 1 \ \text{ for all } \ y_i, y_j \in Y. \end{cases}$$

Remark 7 Note that, $\rho_{ij}^{(k)}$ is decreasing as the total number of alternatives, m, increases. Fig. 6.1 plots the individual preference-discordance $\rho_{ij}^{(k)}(y_i, y_k)$ as a function of m, where y_i, y_j are two adjacent alternatives for the k-th voter.

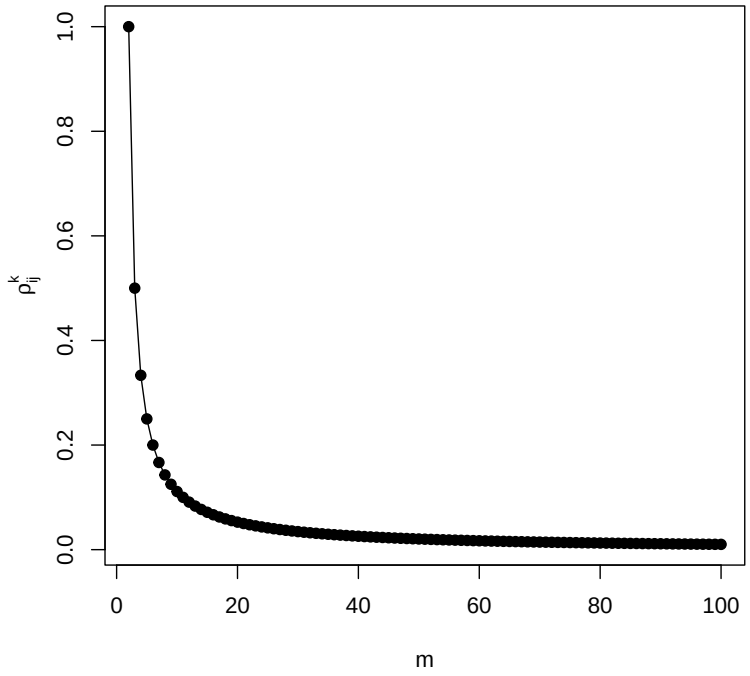


Figure 6.1: Individual preference-discordance of two adjacent alternatives by m.

The total number of different alternatives, m, determines the expressivity of voters. Two

alternatives $y_i, y_j \in Y$ that are adjacent are considered more similar in a large order than in a small one. For example, consider the universe of weak orders for $m = 4$: y_i and y_j are adjacent in 40 out of the 75 possible scenarios, about 53%. On the contrary, when the number of alternatives doubles, $m = 8$, the number of weak orders in which y_i and y_j are adjacent drops to 170 440 out of 545 835, approximately 31%. As m increases, the percentage of scenarios in which y_i and y_j are adjacent decreases so does the average distance between them.

Finally, the *average preference-discordance*, $\bar{\rho}_{ij}$, summarizes the average dissimilarity between two alternatives according to the whole set of voters:

$$\bar{\rho}_{ij} = \frac{1}{n} \sum_{k=1}^{n} \rho_{ij}^{(k)}. \tag{6.5}$$

6.2.2 Approval discordances

The *individual approval-discordance* between y_i and y_j for the voter $v_k \in V$ is defined as

$$\alpha_{ij}^{(k)} = |I_{A_k}(y_i) - I_{A_k}(y_j)|, \tag{6.6}$$

where $\alpha_{ij}^{(k)} \in \{0, 1\}$.

Unlike $\rho_{ij}^{(k)}$, the individual approval-discordance is not influenced by the number of alternatives whose acceptability is established. Considering all possible approvals of m alternatives, the percentage of approvals in which y_i and y_j receive the same rating remains constant as m varies.

Finally, the *average approval-discordance*, $\bar{\alpha}_{ij}$, summarizes the average dissimilarity between two alternatives according to the whole set of approvals:

$$\bar{\alpha}_{ij} = \frac{1}{n} \sum_{k=1}^{n} \alpha_{ij}^{(k)}. \tag{6.7}$$

It is worth noting that the *individual* discordances, $\rho_{ij}^{(k)}$ and $\alpha_{ij}^{(k)}$, are related to the *pair-wise* discordances, p_{ij} and a_{ij}, defined in the previous Chapter. In particular, the pair-wise discordance compares the agreement of two voters on two alternatives. Thus, the final distance between two voters is derived by summing over the possible pairs of items. On the contrary, individual discordances consider the degree to which two alternatives are considered different for each voter. Thus, the final distance between two alternatives can be derived by summing voter-by-voter.

6.2.3 Global discordances

In order to generate a global measure of discordance between each pair of alternatives, we consider the family of *weighted means*, $h:[0,1]\times[0,1]\longrightarrow[0,1]$, defined as

$$h(x,y)=\lambda\cdot x+(1-\lambda)\cdot y, \tag{6.8}$$

where $\lambda\in[0,1]$.

Taking into account the preference and approval discordances introduced in Eqs. (6.4), (6.5), (6.6) and (6.7), respectively, and the family of weighted means defined in Eq. (6.8), we now introduce a global measure of discordance between pairs of alternatives.

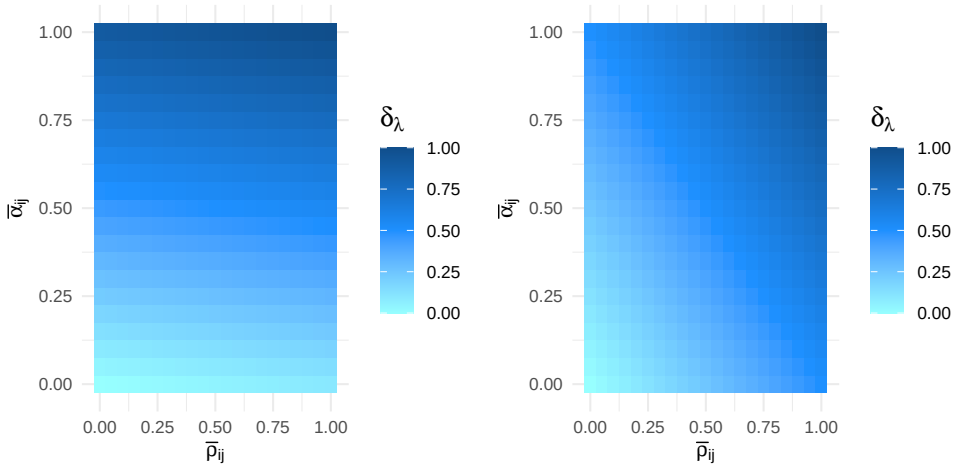
Definition 4 *Given a profile* $\Big((\pi_1,A_1),\ldots,(\pi_m,A_m)\Big)\in\mathscr{R}(Y)^m$ *and* $\lambda\in[0,1]$, *the mapping* $\delta_\lambda:Y\times Y\longrightarrow[0,1]$ *is defined as*

$$\delta_\lambda(y_i,y_j)=\frac{1}{n}\cdot\sum_{k=1}^{n}\Big(\lambda\cdot\rho_{ij}^{(k)}+(1-\lambda)\cdot\alpha_{ij}^{(k)}\Big)=\lambda\cdot\bar{\rho}_{ij}+(1-\lambda)\cdot\bar{\alpha}_{ij}. \tag{6.9}$$

Proposition 6 *Given a profile* $\Big((\pi_1,A_1),\ldots,(\pi_m,A_m)\Big)\in\mathscr{R}(Y)^m$, *the mapping* δ_λ *is a pseudometric on Y for every* $\lambda\in[0,1]$. *We say that* δ_λ *is the pseudometric associated with* λ.

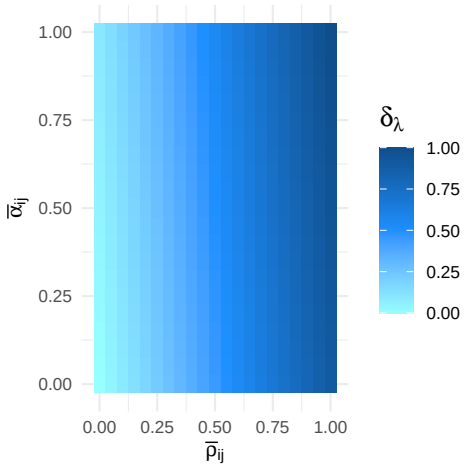
PROOF: Taking into account Propositions 4 and 5, it is obvious that δ_λ satisfies the following conditions for all $y_i,y_j\in Y$: $\delta_\lambda(y_i,y_j)\geq 0$, $\delta_\lambda(y_i,y_i)=0$ and $\delta_\lambda(y_i,y_j)=\delta_\lambda(y_j,y_i)$. Finally, δ_λ satisfies the triangle inequality being a convex combination of two pseudometrics. ■

Fig. 6.2 shows how δ_λ varies as a function of $\bar{\rho}_{ij}$ and $\bar{\alpha}_{ij}$ for $\lambda=0.1,0.5,0.9$.



(a) $\lambda = 0.1$

(b) $\lambda = 0.5$



(c) $\lambda = 0.9$

Figure 6.2: Heatmaps δ_λ.

In Fig. 6.2b, λ is set to 0.5. Thus, $\bar{\rho}_{ij}$ and $\bar{\alpha}_{ij}$ have the same weight in determining the final distance $\delta_\lambda(y_i, y_j)$. As a result, the corresponding heatmap is symmetrical with respect to the secondary diagonal, and δ_λ increases diagonally from bottom to top and from left to right.

On the contrary, $\lambda = 0.1$ (Fig. 6.2a) and $\lambda = 0.9$ (Fig. 6.2c) correspond to two unbalanced settings. Giving much more importance to approvals, $\lambda = 0.1$ (Fig. 6.2a), causes the bottom area of the graph to contain lower distances, δ_λ grows much more

noticeably vertically rather than horizontally. Finally, in Fig. 6.2c, δ_λ is dominated by the preference-discordance. The lesser distances are found on the left side of the graph, and δ_λ expands horizontally significantly more than vertically.

6.2.4 Clustering procedure and visualization

In this chapter, we use the algorithm *Ranked k-medoids* (RKM) (see Zadegan et al. (2013)) to find clusters, but we highlight that our pseudometrics can be used jointly with any distance-based clustering algorithm.

The RKM method introduces a function that ranks alternatives according to their similarities. With this function, the more similar alternative gets a lower rank. In other words, $\text{rank}(y_i, y_j) = l$ shows that y_j is the l-th similar alternative to y_i among m alternatives in the dataset. The ranks of the remaining objects according to an object like y_i can be computed by sorting the similarity values between y_i and other objects in the dataset. The rank function also expresses a rank matrix $K = [k_{ij}]$, where $\text{rank}(y_i, y_j) = k_{ij}$ for all $y_i, y_j \in Y$.

Note that K is not necessarily a symmetric matrix since two objects are not always at the same rank as each other. Thus, K is a $m \times m$ matrix that shows the *hostility* relationship among alternatives in the dataset. In order to find the medoids, the hostility value (hv) of an object in a group of objects is introduced. The hostility value, hv_i, of an object y_i in a set of objects G is defined as follows:

$$\text{hv}_i = \sum_{y_i \in G} k_{ij}. \tag{6.10}$$

Starting from the similarities among pairs of objects based on $\delta_\lambda(y_i, y_j)$, the RKM algorithm firstly calculates K matrix and selects the medoids randomly. Then, for each medoid, select the group of the most similar objects to each medoid, using the sorted index matrix, and calculate the hostility values of every object in those groups using Eq. (6.10). Afterwards, select the object with the highest hostility value as the new medoid and move one of the medoids placed in the same group. Finally, iterate the process and assign each object to the most similar medoid.

The RKM method is particularly suitable in our case since it analyzes a rank ordering of dissimilarities, which makes the results order-invariant, meaning that order-preserving transformations of the data have no effect.

In order to represent the resulting clusters in a 2-dimensional space, multidimensional scaling (MDS) is employed. This class of methods attempts to express an observable proximity or distance matrix by a simple geometrical model or map so that the greater

the perceived distance between two alternatives, the more apart the points representing them in the final geometrical model are.
Such models estimate q-dimensional coordinate values to represent m alternatives of a distance matrix. They optimize a chosen goodness of fit index, which measures how well the fitted distances match the observed proximities. A number of optimization strategies, when combined with a variety of goodness of fit indices, result in various MDS algorithms (Hothorn and Everitt, 2006).
In this chapter, given the nature of the objects, the Non-metric Multidimensional Scaling is employed. This method constructs fitted distances in the same rank order as the original distance, thus preserving the rank order of the proximities. Algorithms for accomplishing this are described in Kruskal (1964). The required coordinates for a given set of disparities are found by minimizing a function of the squared differences between the observed proximities and the derived disparities, known as *Stress*. The procedure then iterates until a properly selected convergence criterion is met.

6.3 Case studies

This section shows how the proposed metric can be used to perform cluster analysis on real data.

6.3.1 Eurobarometer dataset

The data utilized in these analyses come from the EuroBarometer website, specifically, from question QA7 of the survey titled "Public opinion in the European Union"[1]. A group of voters, divided by countries, were asked to indicate which of the values included in Tab. 6.1 the EU mean to them.

[1] https://europa.eu/eurobarometer/surveys/detail/2553.

Alternatives	Names
y_1	Peace
y_2	Economic prosperity
y_3	Democracy
y_4	Social protection
y_5	Freedom to travel, study and work anywhere in the EU
y_6	Cultural diversity
y_7	Stronger say in the world
y_8	Euro
y_9	Unemployment
y_{10}	Bureaucracy
y_{11}	Waste of money
y_{12}	Loss of our cultural identity
y_{13}	More crime
y_{14}	Not enough control at external borders
y_{15}	Quality of life of future generations

Table 6.1: Values in the EU.

As a result, data are stored in Tab. 6.5, with 27 rows (one row for each EU member country) and 15 columns (each column representing an alternative of $Y = \{y_1, \ldots, y_{15}\}$). The total number of votes cast by the i-th country in favour of the j-th alternative is shown in the table's generic cell ij.

In order to transform the original table into a set of preference-approvals, preferences and approvals need to be derived. Following Albano et al. (2022a), the alternatives are ranked in order of popularity, from the most to the least voted, and approvals are derived by considering those alternatives that received more votes than the national average as acceptable. For example, in Tab. 6.2 we show the votes expressed in Italy (the votes of all countries are included in Tab. 6.5).

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	y_{13}	y_{14}	y_{15}
Italy	24	13	19	11	43	15	27	32	13	12	12	10	9	16	22

Table 6.2: Votes in Italy.

Since the votes' average is 18.53, the votes in Italy are transformed into a preference-approval codification (see Eq.(5.3)) as

$$(4, 9.5, 6, 13, 1, 8, 3, 2, 9.5, 11.5, 11.5, 14, 15, 7, 5) \,\&\, (1,0,1,0,1,0,1,1,0,0,0,0,0,0,1)$$

that can be visualized as follows

y_5
y_8
y_7
y_1
y_{15}
y_3

---

y_{14}
y_6
y_2 y_9
y_{10} y_{11}
y_4
y_{12}
y_{13}

In Fig. 6.3, the 15 alternatives are arranged on the preference-approval plane. The location of each alternative in this 2-dimensional space is identified by its expected rank (i.e., the average rank over the whole set of voters) and by its relative approval, i.e., the relative frequency of voters who considered it acceptable.

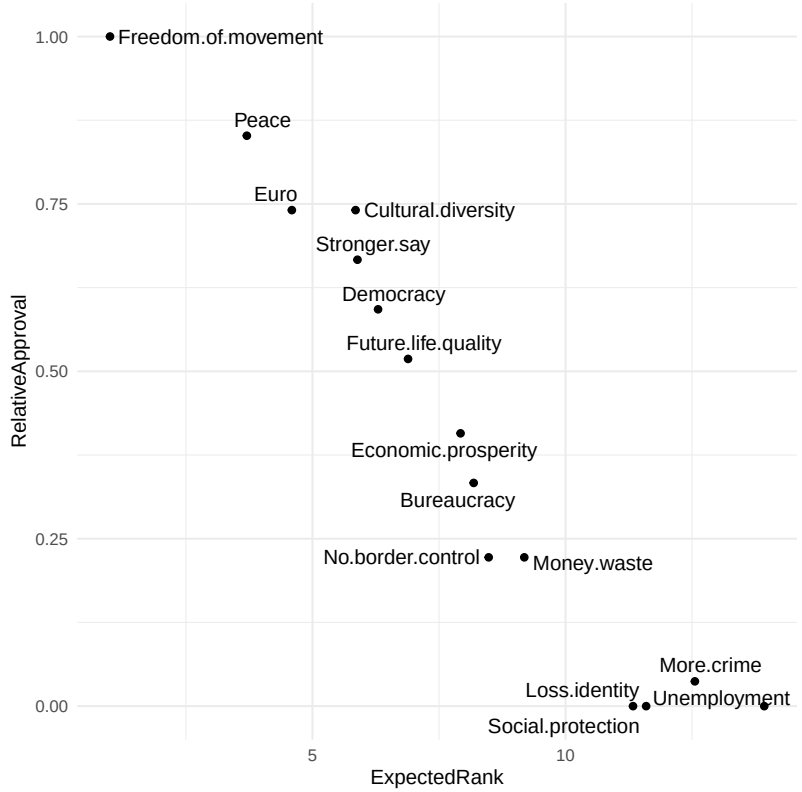


Figure 6.3: Preference-approval plane, Eurobarometer.

The preference-approval plane provides a summary of the evaluations of voters on average. In particular, it reveals that all voters consider "Freedom of movement" the best alternative: it is unanimously approved and always placed first in the preference-approvals; its RelativeApproval and its ExpectedRank are both equal to 1. The other alternatives tend to lie on a straight line with a negative angular coefficient. The further we move away from the point $(1,1)$, the worse the corresponding alternatives obtained average ratings.

Note that the preference-approval plane aids the interpretation of clusters once they have been estimated. However, it should not be considered a tool to identify clusters since the distance between points in the preference-approval plane does not necessarily reflect the pseudometric in Eq.(6.9). Alternatives having similar average ranking positions and approvals may show discordance over the voters.

Example 3 To further clarify this concept, let us consider $(\pi_1, A_1), (\pi_2, A_2) \in \mathscr{R}(\{y_1, y_2, y_3, y_4\})$

the following preference-approvals:

$$(\pi_1, A_1) \equiv \begin{array}{c} y_1 \\ y_2 \\ \hline y_3 \\ y_4 \end{array} \qquad (\pi_2, A_2) \equiv \begin{array}{c} y_4 \\ y_2 \\ \hline y_3 \\ y_1 \end{array}$$

For each alternative $y_i \in Y$, the expected rank, expected approval and $\delta_{0.5}$ distance matrix are reported in Tables 6.3 and 6.4.

Alternative	ExpectedRank	RelativeApproval
y_1	**2.5**	**0.5**
y_2	2	1
y_3	3	0
y_4	**2.5**	**0.5**

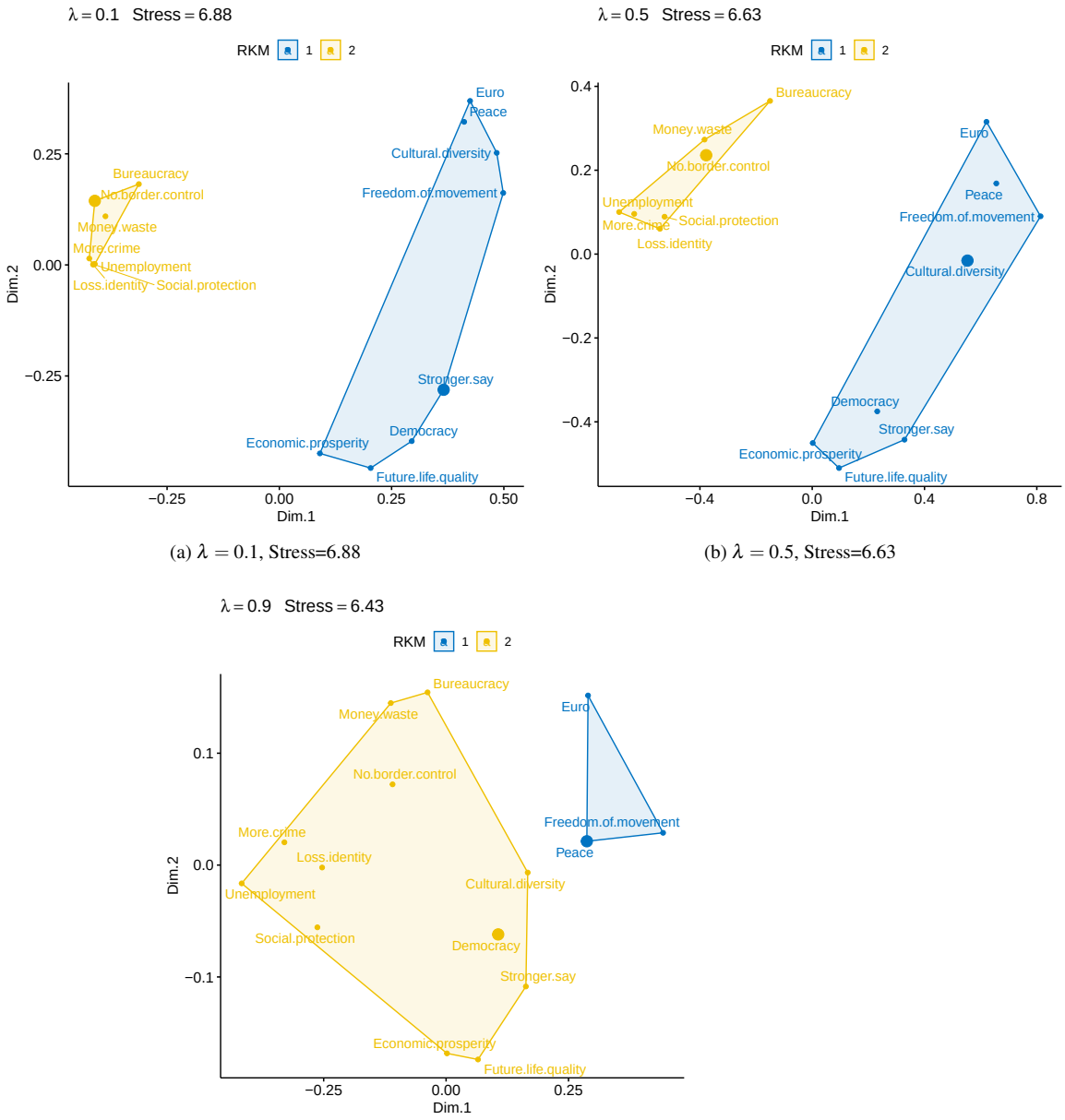
Table 6.3: ExpectedRank and RelativeApproval.

	y_1	y_2	y_3	y_4
y_1	0			
y_2	0.5	0		
y_3	0.5	0.67	0	
y_4	**1**	0.5	0.5	0

Table 6.4: Distances $\delta_{0.5}$.

Note that y_1 and y_4 have the same relative approval and expected rank, thus identical coordinates in the preference-approval plane, but show maximum discordance over the voters, i.e. $\delta_{0.5}(y_1, y_4) = 1$. In fact, they are placed at the opposite extremes in both preference-approvals. Therefore, the preference-approval plane is intended to be an interpretative tool to visualize average judgments and interpret clusters once they have been estimated. At the same time, it is not appropriate to identify clusters since it does not reflect similarities among elements.

Fig. 6.4 shows the clusters estimated by the RKM algorithm, where the central medoid for each cluster is highlighted through the dimension of the point. We investigate the effect of the λ parameter on the output, by setting $\lambda = 0.1, 0.5, 0.9$. In this way, we are able to study three scenarios: $\lambda = 0.5$, which corresponds to giving the same importance to approvals and preferences, and $\lambda = 0.1, 0.9$, which corresponds to the opposite unbalanced situations.
We also show the Stress values in each scenario to assess the goodness of the graphical representation obtained with the MDS. Note that the position of the points in the new space found by MDS depends on the value of λ. If the parameter, λ varies, the graphical representation does as well.



(a) $\lambda = 0.1$, Stress=6.88

(b) $\lambda = 0.5$, Stress=6.63

(c) $\lambda = 0.9$, Stress=6.43

Figure 6.4: Graphical representation of RKM clusters.

In general, the Stress coefficient varies between 6.88 and 6.43, showing a good adaptation that tends to improve slightly as λ increases. The optimal number of clusters,

chosen through the Silhouette criterion, turns out to be two independently from the λ value.

When $\lambda = 0.1, 0.5$, the clusters found are the same, but the degree of separation between them clearly changes. In fact, the two clusters exhibit a higher separation index[2] under $\lambda = 0.1$, i.e. assigning much more weight to the approvals than under $\lambda = 0.5$. Thus, a clear division is obtained between frequently accepted alternatives and frequently not accepted alternatives. The two clusters become closer as λ reaches 0.5. In this example, there are clearly two different types of alternatives: those referring to negative aspects ("Bureaucracy", "Unemployment", "Money waste", etc.) and those referring to positive aspects ("Freedom", "Democracy", etc.). For this reason, a voter with a bad opinion about the EU will prefer the former and vice versa. Indeed, the two clusters are robust and remain unchanged for small and moderate values of λ.

Note that the proximity between points in the two-dimensional space discovered by the MDS (Figures 6.4a, 6.4b and 6.4c) reflects the similarities based on δ_λ, between the alternatives over the voters. Thus, the position of the elements in this new space addresses the cluster interpretation.

Indeed, although in the preference-approval plane (see Fig. 6.3), "Money waste" is closer to the alternatives belonging to Cluster 1, its position in the MDS space reveals that actually, it is part of Cluster 2.

Fig. 6.4c displays clusters under $\lambda = 0.9$, i.e., unbalanced towards preferences. In this case, Cluster 1 isolates the three alternatives frequently placed in the first positions (see the preference-approval plane Fig. 6.3), namely: "Freedom", "Peace" and "Euro".

[2] Based on the distances for every point to the closest point not in the same cluster.

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}	y_{12}	y_{13}	y_{14}	y_{15}
Belgium	32	22	23	14	52	24	27	42	7	18	25	12	11	19	20
Bulgaria	17	15	17	10	57	20	18	11	7	14	14	17	9	12	24
Czech Republic	33	30	31	8	66	18	32	17	2	37	28	18	7	20	30
Denmark	50	31	35	15	56	27	32	15	4	32	13	11	10	20	22
Germany	51	25	37	12	61	36	31	46	7	33	29	11	20	26	21
Estonia	30	19	28	13	76	29	18	50	2	34	19	19	4	19	22
Ireland	30	28	22	15	57	25	26	35	4	12	5	7	5	8	25
Greece	45	14	25	18	66	35	45	48	30	17	20	28	23	36	19
Spain	16	22	18	14	47	24	20	27	6	16	11	3	3	9	15
France	37	10	19	12	46	31	20	37	9	18	30	14	8	24	15
Croatia	26	27	23	20	55	24	26	23	4	11	12	18	9	13	39
Italy	24	13	19	11	43	15	27	32	13	12	12	10	9	16	22
Cyprus	38	20	26	24	67	36	26	45	31	21	22	24	35	36	25
Latvia	24	22	16	15	60	20	10	31	6	19	17	14	4	12	25
Lithuania	33	17	20	18	75	32	25	18	3	18	20	13	4	17	31
Luxembourg	46	23	34	17	61	36	22	51	6	26	27	12	18	26	22
Hungary	21	18	23	13	48	24	20	15	6	15	8	10	10	20	26
Malta	22	29	27	21	56	24	34	31	3	16	10	16	5	17	32
Netherlands	53	41	28	10	67	30	37	47	3	31	17	11	9	17	33
Austria	39	25	30	25	58	26	32	50	24	34	37	25	31	38	26
Poland	26	19	29	10	47	16	23	10	5	12	12	11	7	11	29
Portugal	19	27	24	18	57	27	32	44	6	6	7	7	6	16	22
Romania	23	19	21	14	45	16	16	23	9	15	14	18	15	16	21
Slovenia	36	25	27	16	54	27	20	47	5	18	15	13	15	16	24
Slovakia	28	20	16	10	64	21	28	44	12	28	33	20	22	38	19
Finland	32	18	24	6	67	27	21	54	3	35	24	11	11	22	18
Sweden	46	19	34	12	67	25	44	14	4	40	29	10	17	23	19

Table 6.5: Votes in the EU.

6.3.2 Pew Research Center dataset

Pew Research Center is a research institute that conducts public opinion polling, demographic research, content analysis and other data-driven social science research.
In this analysis, the survey "American Trends Panel Wave 33"[3] is considered. Data in this report is drawn from the panel wave conducted from March 27 to April 9, 2018, to collect the opinions of United States citizens regarding the space agency NASA.
In this analysis, we focus specifically on a query in which a total of 2541 respondents were asked to assess how much priority NASA should give to a list of nine lines of

[3] https://www.pewresearch.org/science/dataset/american-trends-panel-wave-33/

action, listed in Tab. 6.6. Individuals employed the linguistic terms from the qualitative scale in Tab. 6.7 to accomplish this.

Alternatives	Names
y_1	Searching for life and planets that could support life
y_2	Searching for raw materials and natural resources that could be used on Earth
y_3	Conducting basic scientific research to increase knowledge and understanding of space
y_4	Developing technologies that could be adapted for uses other than space exploration
y_5	Monitoring asteroids and other objects that could potentially hit the Earth
y_6	Monitoring key parts of the Earth's climate system
y_7	Sending human astronauts to explore the moon
y_8	Sending human astronauts to explore Mars
y_9	Conducting scientific research on how space travel affects human health

Table 6.6: Lines of action.

	Linguistic term
l_1	Top priority
l_2	Important but lower priority
l_3	Not too important
l_4	Should not be done
l_5	No answer

Table 6.7: Linguistic terms.

In order to remove neutral answers, the respondents giving at least a "No answer" response were excluded, i.e., about 3% of the total sample size. Furthermore, for each respondent, alternatives were arranged into a preference-approval. The two linguistic terms l_1 and l_2 were used to indicate an acceptable alternative. An example is provided in Tab. 6.8.

Respondent	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9
v_{10}	l_4	l_2	l_1	l_1	l_4	l_3	l_4	l_4	l_2

Table 6.8: Pew Research Center example.

The respondent v_{10} preference-approval (see Eq.(5.3)) is

$$(7.5, 3.5, 1.5, 1.5, 7.5, 5, 7.5, 7.5, 3.5)\,\&\,(0,1,1,1,0,0,0,0,1)$$

that can be visualized as follows

$$\begin{array}{c} y_3\ y_4 \\ y_2\ y_9 \\ \hline\hline y_6 \\ y_1\ y_5\ y_7\ y_8 \end{array}$$

In contrast to the previous example, approvals are generated directly by the voters through linguistic terms so that each voter can define all items as acceptable or vice versa.
Fig. 6.5 shows the nine alternatives on the preference-approval plane.

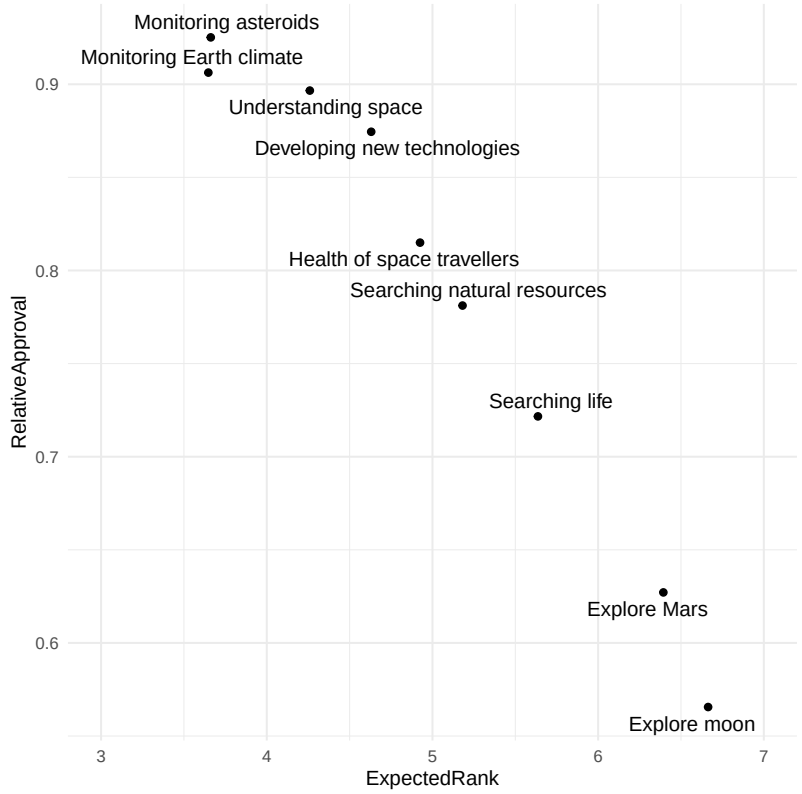
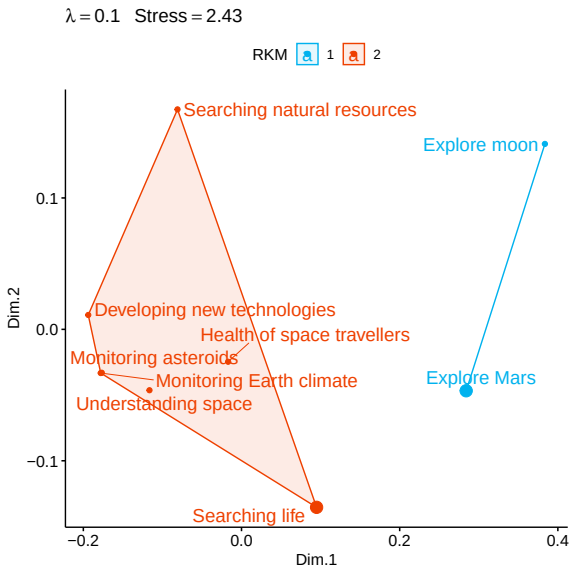


Figure 6.5: Preference-approval plane, Pew Research Center.

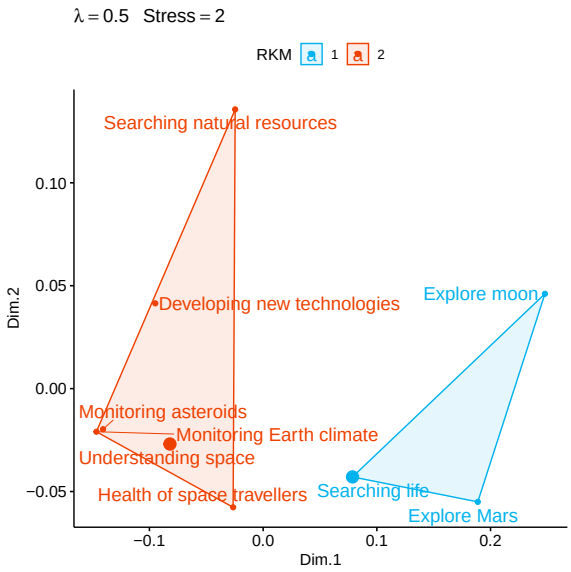
In this example, the relative approval of each item ranges between 55% and 95%, meaning that each alternative has been considered acceptable by more than half of

the individuals. Therefore, although the alternatives may be regarded as acceptable by voters on average, less urgent alternatives, such as exploration of other planets and satellites (Moon and Mars), and more urgent alternatives, such as earth monitoring (Climate and Asteroids) can be identified.

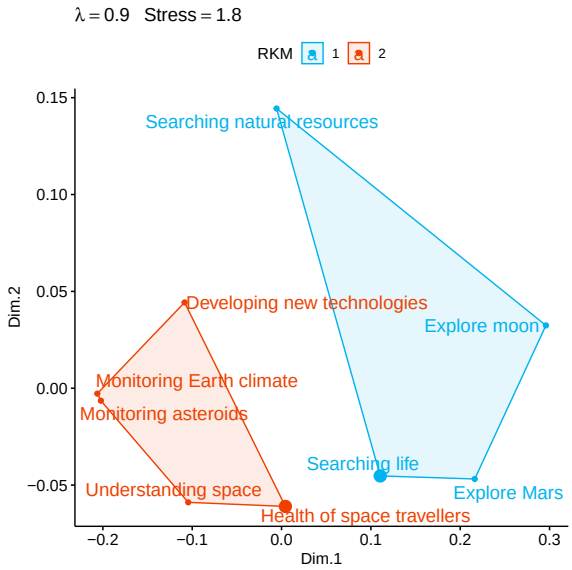
The clusters estimated by the RKM algorithm are shown in Fig 6.6. As in the previous example, three different values of $\lambda = 0.1, 0.5, 0.9$ are used. For each scenario, the clusters, medoids and Stress values reached by the MDS are illustrated for graphical representation.



(a) $\lambda = 0.1$, Stress=2.43



(b) $\lambda = 0.5$, Stress=2



(c) $\lambda = 0.9$, Stress=1.8

Figure 6.6: Graphical representation of RKM clusters.

In general, the stress coefficient varies between 2.43 and 1.8, showing an excellent adaptation that tends to improve as λ increases. The optimal number of clusters, chosen

through the Silhouette criterion, turns out to be two independently from the λ value.
The effect of the parameter λ on cluster building is visible. Setting $\lambda = 0.1$ results in Cluster 1, including only the two alternatives related to space exploration (Moon and Mars), which have the lowest relative approval (see Fig. 6.5). Voters strongly tend to attribute the same approvals to these two alternatives.
Increasing the value of $\lambda = 0.5$ causes Cluster 1 to enlarge by including the alternative "Searching life". Finally, giving much more weight to the rankings, i.e. $\lambda = 0.9$, results in Cluster 1 also including the alternative Searching natural resources. In this way, Cluster 1 contains the four alternatives most frequently placed in the last positions in voters' preference-approvals.

6.4 Concluding remarks

Preference-approvals structures are gaining increasing attention in social choice as they allow decision makers to describe their preferences using more flexible and intuitive ordinal information. In this chapter, we propose a new method for clustering alternatives in preference-approvals. First, we introduce a family of pseudometrics, δ_λ, able to quantify the distance between alternatives based on two main components: the *individual preference-discordance* ρ_{ij} and the *individual approval-discordance* α_{ij}, and on the λ parameter, which regulates the weight to give to each component.
To obtain clusters, we apply the Ranked k-medoids partitioning algorithm, taking as input the similarities among pairs of alternatives based on the proposed pseudometrics. Finally, clusters are represented in 2-dimensional space using Non-Metric Multidimensional Scaling.
Through two applications to real data, we demonstrate how our algorithm allows dividing a heterogeneous population of alternatives into homogeneous groups, reducing the complexity of the preference-approval space and providing a more accessible interpretation of data. We also show the effect of the λ parameter on cluster identification and visualization.
Future research should consider using the proposed clustering method to collapse categories in the context of multiple-choice models. Moreover, it will be important that future research investigate a method to identify simultaneous clusters of both individuals and alternatives in the preference-approval framework, extracting helpful information in a low-dimensional subspace.

Chapter 7

Ranking coherence in Topic Models using Statistically Validated Networks

7.1 Introduction

The task of ranking alternatives is useful in a variety of scientific fields. Indeed, ranking methods are especially well suited to solving textual analysis problems. The scientific interest in automatic textual analysis has grown dramatically over the last decade due to the increase in available digital textual data. Indeed, researchers from several disciplines have become increasingly interested in incorporating textual data in their works. Text Mining or Knowledge Discovery from Text (KDT) was first introduced by Feldman and Dagan (1995) and refers to the process of extracting high-quality information from text. One of the most critical goals of text mining is the clustering task (Allahyari et al., 2017), studied in different research domains such as data mining (Berkhin, 2006), machine learning (McGregor et al., 2004), and information retrieval (Wu et al., 2003). Topic modeling (Blei et al., 2003) is one of the most popular probabilistic clustering algorithms, since it aims to process extensive collections of texts that are useful for tasks such as classification, novelty detection, summarisation, similarity and relevance judgments.

These models learn topics automatically, from unlabeled documents in an unsupervised way. These topics are called hidden thematic structures or latent topics and are typically

represented as sets of essential words. Documents are considered as a mixture of topics, where each topic is represented by a probability distribution of words (Blei, 2012). Thus, these models build latent topics as multinomial distributions of words, and the models assume that each document can be described as a mixture of these topics. Each topic's essential words frequently tend to appear together and (hopefully) are related to the same common theme. Once the models are trained, they provide a framework for humans to understand document collections both directly by "reading" models or indirectly by using topics as input variables for further analysis (Boyd-Graber et al., 2017). The Latent Dirichlet Allocation (LDA) is one of the most popular topic models and the state-of-the-art unsupervised machine learning technique for extracting thematic information (topics) from a collection of documents.

Indeed as highlighted by Boyd-Graber et al. (2017), LDA plays an essential role in the analysis of historical documents, scientific documents, fiction, poetry and literature. The main obstacle in topic detection models is that not all the estimated topics are equally important, and not all correspond to genuine domain themes. Some of the topics can be a collection of irrelevant words or unchained words representing insignificant themes.

Often, in qualitative studies, the goal is to find meaningful and interpretable topics. Researchers usually use top-N words with the highest probability given a topic (Lau et al., 2014; Newman et al., 2010; Aletras and Stevenson, 2013; Ramrakhiyani et al., 2017), and employ humans to obtain an interpretability score. Indeed, topic discovering algorithms do not automatically provide a way to interpret their output. For instance, Chang et al. (2009) state that "Although there appears to be a longstanding assumption that the latent space discovered by topic models is meaningful and useful, evaluating such assumptions is difficult because discovering topics is an unsupervised process". Moreover, Hoyle et al. (2021) highlight that automated evaluation metrics often suffer from inconsistency. Therefore, it would be desirable to fully automize the process by introducing a metric that automatically ranks learned topics closely matching human judgments. This challenge motivated recent research on topic quality metrics that closely match human judgement. Within this framework, quantifying the coherence of a set of words plays a central role (AlSumait et al., 2009; Lau et al., 2014; Newman et al., 2010; Aletras and Stevenson, 2013; Nikolenko et al., 2017; Ramrakhiyani et al., 2017; Röder et al., 2015) .

In topic models, a topic can be viewed as a set of words that frequently co-occur in the same documents, which is very similar to latent word groups (or communities) (Zuo et al., 2016) in the word network. Since words that frequently co-occur in the same sentences are closely connected in the semantic space, they tend to appear in the same

document.

This chapter proposes a new ranking method to explore topic coherence based on the construction and analysis of Statistically Validated Networks (SVNs) of words (Tumminello et al., 2011). Specifically, the method builds a co-occurrence network for each topic whose most probable words are the nodes. We set a link between two nodes (words) in each network if their co-occurrence in sentences is statistically significant. We claim that these links carry relevant information about the structure of the topic, i.e., the more connected the network, the more semantically coherent the corresponding topic. Therefore, we propose to use connectivity measures on the SVN of words to build a metric of topic coherence.

The main contributions of this chapter are: i) to define a new coherence measure (Coh_{SVN}) based on a rigorous statistical model that approximates human ratings better than state-of-the-art methods; ii) to filter out marginal associations of words and to facilitate the graphical representation and interpretation of the obtained topics through Statistically Validated Networks (SVNs) (Tumminello et al., 2011).

The chapter is organized as follows: Section 7.2 describes the background and reviews related works. In Section 7.3, we describe the proposed coherence model, while we report a real-world application of the method in Section 7.4. Finally, in Section 7.5, we draw our conclusions and propose ideas for future development.

7.2 Background and related works

The main idea of topic modeling is to create a probabilistic generative model for a corpus of text documents. A probabilistic topic model is a type of generative model that aims to learn the latent semantic structure of a corpus. Probabilistic topic models reduce the complex process of document generation to a small number of probabilistic steps by assuming exchangeability, because only word occurrence information (i.e., frequencies) is considered.

The first probabilistic topic model was the Probabilistic Latent Semantic Analysis (pLSA), introduced by Hofmann (1999); unfortunately, the model does not provide any probabilistic model at the document level. Then, Blei et al. (2003) proposed the The Latent Dirichlet Allocation (LDA) model as an extension of the pLSA, introducing a Dirichlet prior on mixture weights of topics per document. The name of the model incorporates its main features. Specifically, the term *Latent* indicates that the model involves probabilistic inferences for extrapolating missing probabilistic pieces of the generative story from texts. The term *Dirichlet* recalls that the model uses Dirichlet

parameters to encode sparsity. Finally, the name includes the word *Allocation* since the Dirichlet distribution encodes the prior probability for each document's allocation of the topics (Boyd-Graber et al., 2017). In these models, documents are described as random mixtures over latent topics, where a distribution of words characterizes each topic (Blei et al., 2003). The words of the documents are the observed variables, whereas the topic structures are the hidden variables. The problem of inferring the hidden topic structure from the documents consists in computing the posterior distribution of topic structures, that is, the conditional distribution of the hidden variables given the documents (Blei, 2012).

Recently, many other probabilistic topic models that consider topic correlations were proposed, such as the correlated topic model (CTM–see Blei and Lafferty (2006)), the Pachinko allocation model (see Li and McCallum (2006)). Other works extend probabilistic topic models focusing on the evolution of topics over time, such as the dynamic topic model (DTM) (Dieng et al., 2019), or introducing word embedding representation–the embedded topic model (ETM) by Dieng et al. (2020).
Finally, neural topic models represent a broader set of related models. These mainly focus on improving topic modeling inference through deep neural networks (see Srivastava and Sutton (2017)).
Finally, Blei (2012) and Boyd-Graber et al. (2017) provide comprehensive reviews of probabilistic topic models.
Among these models, we applied our coherence measure to the LDA model, since it represents a benchmark in the topic modelling community, for comparison with its various extensions. However, it is worth highlighting that the proposed measure applies to any topic model.

7.2.1 Literature review

Evaluating the quality of the latent spaces provided by topic models is a difficult challenge because discovering topics is an unsupervised process that gives no guarantees on the interpretability of its output. In text mining, the problem of semantic evaluation has attracted much interest breaking down the research into coherence measures (Röder et al., 2015). There is no gold-standard list of topics to compare against for every corpus. Thus, a technique for evaluating the outputs of topic models could be employed to gather exogenous data. In this section, we discuss previous work on the topic evaluation.
For many years, the primary way to evaluate the quality of a topic model was to mea-

sure the log-likelihood of a held-out test set (Blei et al., 2003; Wallach et al., 2009). The held-out likelihood consists of density estimation on a collection of unseen documents given a training set. The most commonly used measure based on the held-out method is the perplexity, a monotonically decreasing function of likelihood:

$$perplexity(D) = exp\left\{ - \frac{\sum_{d=1}^{M} \log p(\mathbf{w}_d)}{\sum_{d=1}^{M} N_d} \right\},$$

where D is the collection of documents, N_d is the number of words in document d, and $p(\mathbf{w}_d)$ is the marginal distribution of document d, following the notation used in previous section. A lower perplexity score indicates better generalization performance. However, Chang et al. (2009) showed that the perplexity on the held-out test set emphasizes *complexity* rather than *interpretability*, which is the property users are mostly interested in. In their work, they fit three different topic models to two corpora and demonstrated that the perplexity scores are negatively correlated with human ratings. In other words, such measure is useful for evaluating the predictive performance of the model, but it does not address the more explanatory goals of topic modeling. Indeed, topic models are mainly used to organize, summarize and help users to explore large corpora, while evaluating the predictive performance of the model is a completely different task. Therefore, there is no technical reason to suppose that held-out accuracy corresponds to a better organization or easier interpretation. In recent years, many methods have been proposed for assessing topic coherence. The approaches can be split into two categories: qualitative methods and quantitative methods. Qualitative methods are less common than quantitative since they require the use of human resources for topic assessment, and are time-consuming. Quantitative approaches, on the other hand, seek to automate the whole evaluation process by trying to replicate human judgment.

7.2.2 Qualitative methods

Chang et al. (2009) proposed the task of *word intrusion* to create a formal setting where humans can evaluate the latent space of a topic model. This task allows for an evaluation of whether a topic has human-identifiable semantic coherence or not. In the *word intrusion* task, the subject is presented with six randomly ordered words, and the task of the user is to find the word which is out of place or which does not belong with the others, i.e., the *intruder*.

Later, Morstatter and Liu (2018) proposed a modified version of the word intrusion task, named *Model Precision Choose Two*. As in the word intrusion task, they propose

to form a list with the top (most likely) five words from a topic and to inject one low-probability word from the same topic into the list. The critical difference with word intrusion is that they ask the annotators to select *two* intruded words from the six. The intuition behind this experiment is that the annotators' first choice will be the intruded word, just as in Chang et al. (2009). However, their second choice is what makes the topic's quality clear. In a coherent topic, the annotator will not be able to distinguish a second word as all of the words will appear similarly coherent.

7.2.3 Quantitative methods

The qualitative methods are time-consuming since they require the manual annotations of humans. In the last decade, researchers have proposed fully automating the process by introducing a metric that automatically ranks learned topics. One of the first automated measures was proposed by AlSumait et al. (2009). They introduced an approach to **automatically** rank the LDA topics based on their semantic importance and, eventually, to identify junk and insignificant topics. Their idea is to measure the amount of "*insignificance*" that an inferred topic carries in its distribution by measuring how "different" the topic distribution is from a "*junk*" distribution. In the same work, AlSumait et al. proposed three definitions of Junk and Insignificant (J/I) topic distribution, namely: i) the Uniform Distribution Over Words *(W-Uniform)*, ii) the Vacuous Semantic Distribution *(W-Vacuous)* and iii) the Background Distribution *(D-BGround)*. Finally, to quantify the difference between an estimated topic and a J/I distribution, three different distance measures are employed, namely: Kullback-Leibler (KL) Divergence; Cosine Dissimilarity; and Correlation Coefficient.

Later, Wang et al. (2011) proposed a re-ranking algorithm to select "significant" topics by topic similarity calculation. Specifically, each topic is represented as a probability distribution $p(w_i|z_j)$ over words. To compute the distance between word-topic distributions they employed the Jensen-Shannon distance (a symmetrised extension of the KL divergence) :

$$Dist(z_i, z_j) = \frac{1}{2}[KL(z_i||z_j) + KL(z_j||z_i)].$$

Finally, for each topic i, they computed the average distance between i and all the other topics, and they sorted the average distance for each topic in a queue. The last element in the queue is ranked the highest.

In the framework of topic quality evaluation, many relevant works make use of the top-N most probable words (rather than using the entire word-topic distribution), and they

assess pairwise semantic cohesion among them through their co-occurrences provided by the dataset or external sources.
The general idea is to compute the mean of the sum of the pairwise scores of the top-N words that most contribute to describing the topic:

$$Coherence = \frac{2}{N \cdot (N-1)} \sum_{i=2}^{N} \sum_{j=1}^{i-1} score(w_i, w_j).$$

One of the best-known topic quality measures based on the top-N words was proposed by Newman et al. (2009). They introduced for the first time, a model that uses external text data sources, such as Wikipedia and Google hits, to predict human judgements.
Specifically, Newman et al. (2009) measured co-occurrence of word pairs, taken from the list of the ten most probable words in a given topic, using two huge external text datasets: all articles from English Wikipedia and the Google n-grams data set. Specifically, they identify a co-occurrence of words w_i and w_j if they occurred together in a 10-word window of any Wikipedia article. Similarly, they identify a co-occurrence of the two words according to Google n-grams if they both appear in any of the existing 5-grams. Finally, they measure the score of association between word pairs through the Pointwise Mutual Information (PMI) (Bouma, 2009):

$$PMI(w_i, w_j) = \log \frac{p(w_i, w_j) + \varepsilon}{p(w_i)p(w_j)}, \tag{7.1}$$

where $p(\cdot)$ is the relative frequency of a word and $p(\cdot,\cdot)$ is the relative frequency of the co-occurrence of two words, while ε is a smoothing term. This measure is also called *UCI*.
Mimno et al. (2011) pointed out that "bad" topics can be categorized into three definitions:

- *Chained*: every word is connected to every other word through some pairwise word chain, but not all word pairs make sense.
- *Intruded*: either two or more unrelated sets of related words, joined arbitrarily, or an otherwise good topic with a few "intruder" words.
- *Random*: no clear, reasonable connections between more than a few pairs of words.

In their work, the authors suggest that these poor-quality topics could be detected using metrics based on word co-occurrences within the documents.

They proposed to use an asymmetrical confirmation measure, $UMass$, between top word pairs (smoothed conditional probability), where the estimations of word probabilities are based on their frequencies in the original documents used to train the algorithm on the topics:

$$UMass(w_i, w_j) = \log \frac{D(w_i, w_j) + 1}{D(w_j)}, \tag{7.2}$$

where $D(w_i)$ is the *document frequency of word*, (i.e., the number of documents that contains w_i, and $D(w_i, w_j)$ is *co-document frequency* (i.e., the number of documents containing both words). Note that Eq. 7.2 is equal to the empirical conditional log-probability $\log p(w_i|w_j) = \log \frac{p(w_i,w_j)}{p(w_j)}$ smoothed by adding one to $D(w_i, w_j)$, where $p(w_i) = \frac{D(w_i)}{M}$. Therefore, the score function is not symmetric as it is an increasing function of the empirical probability $p(w_j|w_i)$, where the probability of w_i is higher than the word w_j, given a topic. Therefore, this score measures how much (within the words used to describe a topic) a common word, w_i, is, on average a good predictor for a less common word, w_j.

Another important contribution was given by Lau et al. (2014) who proposed to use the Normalized Pointwise Mutual Information (NPMI) (Bouma, 2009) of word pairs in the automated methods of word intrusion and observed coherence:

$$\text{NPMI}(w_i, w_j) = \frac{\text{PMI}(w_i, w_j)}{-\log\left[p(w_i, w_j) + \varepsilon\right]}, \tag{7.3}$$

where $p(\cdot)$ and $p(\cdot,\cdot)$ are defined as for PMI. The NPMI ranges between (-1,+1) resulting in -1 (in the limit) for never occurring together, 0 when they are distributed as expected under independence, and +1 (in the limit) for complete co-occurrence.

Aletras and Stevenson (2013) proposed a method for determining topic coherence using the distributional similarity between the n most likely words of the topic. Representing each word as a vector, let $\overrightarrow{w_1}$,$\overrightarrow{w_2}$,..., $\overrightarrow{w}_n$ denote the vectors of the top n most probable words in a topic. The authors also assume that each vector consists of N elements (the size of the Vocabulary) and $\overrightarrow{w}_{ij}$ is the jth element of vector $\overrightarrow{w_i}$. The semantic space was created using Wikipedia as a reference corpus and a window of ± 5 words. Then they compute the similarity between words using three measures:

- Cosine similarity:

$$Cos(\overrightarrow{w_i}, \overrightarrow{w_j}) = \frac{\overrightarrow{w_i} \cdot \overrightarrow{w_j}}{\|\overrightarrow{w_i}\| \|\overrightarrow{w_j}\|}$$

- Dice coefficient:

$$Dice(\overrightarrow{w_i}, \overrightarrow{w_j}) = \frac{2\sum_{k=1}^{N} \min(\overrightarrow{w_{ik}}, \overrightarrow{w_{jk}})}{\sum_{k=1}^{N} (\overrightarrow{w_{ik}} + \overrightarrow{w_{jk}})}$$

- Jaccard coefficient:

$$Jaccard(\overrightarrow{w_i}, \overrightarrow{w_j}) = \frac{\sum_{k=1}^{N} \min(\overrightarrow{w_{ik}}, \overrightarrow{w_{jk}})}{\sum_{k=1}^{N} \max(\overrightarrow{w_{ik}} + \overrightarrow{w_{jk}})}.$$

Then, the coherence of topics is constructed by the mean of all pairwise scores. Each of these measures estimates the distance between a pair of words in a topic and produce a topic cohesion measure based on distributional semantics. Röder et al. (2015) proposed a framework that allows for the construction of existing word-based coherence measures as well as new ones, by combining elementary components. They conducted a systematic search of the space of coherence measures for the evaluation and they identified a complex combinations (named *CV*) as the best performers on their test corpora.

Omar et al. (2015) quantitatively describe topics via normalized mean values of pairwise word similarities. They used two types of word similarities, namely, thesaurus and local corpus-based as the descriptive features of a topic, and performed topic classification by using the represented topics as input and a binary 0-1 human ratings.

Some of the latest work in the field was produced by Nikolenko et al. (2017): they highlighted that the topic coherence defined by Mimno et al. (2011) is able to consistently identify bad topics (i.e., topics with poor coherence) but does not perform well in identifying good ones (i.e., topics with a high degree of coherence). To cope with this problem, Nikolenko et al. (2017) proposed tf-idf (term frequency - inverse document frequency) coherence as a modification of Mimno's coherence metric that accounts for the informative content of the topics.

Their idea is to introduce tf-idf scores instead of the number of co-occurrences in order to construct their measure. The tf-idf value, as defined by Salton and Buckley (1988), increases proportionally to the number of times a word appears in a document and is inversely proportional to the number of documents in the corpus that contain that word. This measure privileges the words that not only frequently occur in a given text, but that also occur rarely in other texts. Thus, a coherence metric with tf-idf scores penalizes co-occurrence of common words that have low discriminative power. The

measure for a given topic is defined as follow:

$$C_{tf\text{-}idf}(w_i, w_j) = log \frac{\sum\limits_{d:w_i,w_j \in d} tf\text{-}idf(w_i,d) tf\text{-}idf(w_j,d) + \varepsilon}{\sum\limits_{d:w_i \in d} tf\text{-}idf(w_i,d)},$$

where ε is a smoothing count usually set to either 1 or 0.01, while the $tf\text{-}idf$ metric is computed with augmented frequency:

$$tf\text{-}idf = tf(w,d) \cdot idf(w,d),$$

where

$$tf(w,d) = \left(\frac{1}{2} + \frac{f(w,d)}{\max_{w^* \in d} f(w^*,d)} \right),$$

$$idf(w,d) = log \frac{|D|}{|\{d^* \in D : w \in d^*\}|}.$$

7.3 Methods

In this section, we propose a new coherence measure to evaluate the interpretability of the top words of a topic. Our method consists in building a co-occurrence network for each topic whose most probable words (according to the estimated topic model) are the nodes. The weights of links are calculated as the number of sentences in which the connected words co-occur. In each network, we identify the links whose weight is statistically significant, i.e., those that cannot be explained in terms of random co-occurrences of words in the sentences. Although several measures in the literature have already considered co-occurrence between words as a measure of association, none has undertaken a statistical approach based on hypotheses testing to assess whether the co-occurrence obtained between two words can be attributed to chance or whether these links carry relevant information about the structure of topics. To do this, we exploit Statistically Validated Networks.

7.3.1 Statistically Validated Networks

In recent years, many complex systems have been represented by bipartite networks (Genova et al., 2019; Puccio et al., 2019; Kaya, 2020). The Statistically Validated Network, introduced by Tumminello et al. (2011), is an unsupervised method to statistically test the significance of each link of a projected weighted network as obtained from a multipartite network. It is an unsupervised method that introduces a system of hypotheses for link testing when a multipartite network is projected into a set of nodes. The idea is to represent text data as a bipartite network, Fig7.1, in which the set of nodes S is made by the sentences of corpus and the other set of nodes W is made by a list of words associated with a given topic. A link is set between a word and a sentence if the word belongs to that sentence. Therefore, projecting the set of words, the resulting network is a word-co-occurrence network (Zuo et al., 2016; Paranyushkin, 2011).

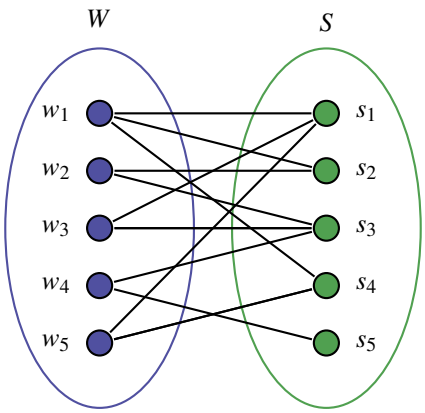


Figure 7.1: Bipartite network where S is the set of corpus sentences and W is the set of topic words.

To take into account the heterogeneity of the set of sentences, a suitable system of hypotheses is introduced. The hypothesis test is constructed as follows. Let us consider a corpus made of N sentences, then consider two words, say, w_i and w_j, and indicate with X_{ij} the times they appear in the same sentences. We are interested in validating the co-occurrences of the words w_i and w_j statistically against a null hypothesis of random co-occurrence that accounts for the heterogeneity of the considered words, that is, the total number of times they appear individually in the text, N_i and N_j, respectively. The probability distribution that describes the random co-occurrence is the hypergeometric distribution, according to which, the probability of observing X_{ij} co-occurrences is

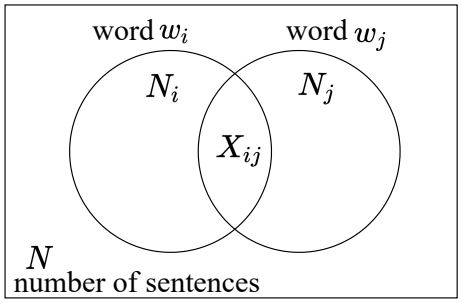


Figure 7.2: Venn Diagram showing the overlap of two words

given by

$$\text{pmf}_H(X_{ij}|N,N_i,N_j) = \frac{\binom{N_i}{X_{ij}}\binom{N-N_i}{N_j-X_{ij}}}{\binom{N}{N_j}}$$

where parameters N_i and N_j naturally allow for the incorporation of the aforementioned heterogeneity of words in the null hypothesis.

The Hypergeometric distribution describes the probability mass function under the null hypothesis in which the probability of co-occurrence between words is conditioned by their marginals, i.e., their individual occurrences.

The distribution introduced can be used to test the presence of an excess of co-occurrence between any pair of words, w_i and w_j. Indeed, assuming that the actual co-occurrences of these words is N_{ij}, then the probability that a value larger than or equal to N_{ij} is observed by chance, according to the null hypothesis, is:

$$p_v(N_{ij}|N_i,N_j,N) = \sum_{X=N_{ij}}^{min(N_i,N_j)} \frac{\binom{N_i}{X}\binom{N-N_i}{N_j-X}}{\binom{N}{N_j}}. \tag{7.4}$$

To claim that the number of co-occurrences, N_{ij}, between words is too large to be consistent with the null hypothesis of random co-occurrences, we shall set a threshold α of statistical significance. However, since we are facing multiple and dependent comparisons, errors of the first kind are a real issue. Therefore, we use the conservative Bonferroni correction (Miller, 1981) for multiple hypothesis testing. The correction states that given a univariate threshold of statistical significance, α, then the threshold corrected for multiple hypothesis testing is $\alpha_T = \frac{\alpha}{T}$, where T is the total number of performed tests, be they dependent or otherwise. The advantage of the Bonferroni correction is that it provides a very strict control of the Family Wise Error Rate even when tests are dependent, as in this case since the same word appears in many tests.

7.3.2 Coherence based on SVNs

In this section, we describe how to construct the new coherence measure, Coh_{SVN}, which makes use of Statistically Validated Networks as combined with different word similarity indices. Specifically, our algorithm can be summarised in the following 5 steps, also sketched in the diagram reported in Fig. 7.3:

(A) Estimate a topic model, and extract the top-m words from each estimated topic;

(B) Represent each topic as a Statistically Validated Network of words;

(C) Evaluate each link's importance, $Imp(w_i, w_j|z_k)$ by considering the strength of the association between word pairs and the relative relevance of each word in the topic;

(D) Compute a global measure of coherence, Coh_{SVN}, for each topic network;

(E) Produce the final ranked list of topics, by sorting them in decreasing order of coherence.

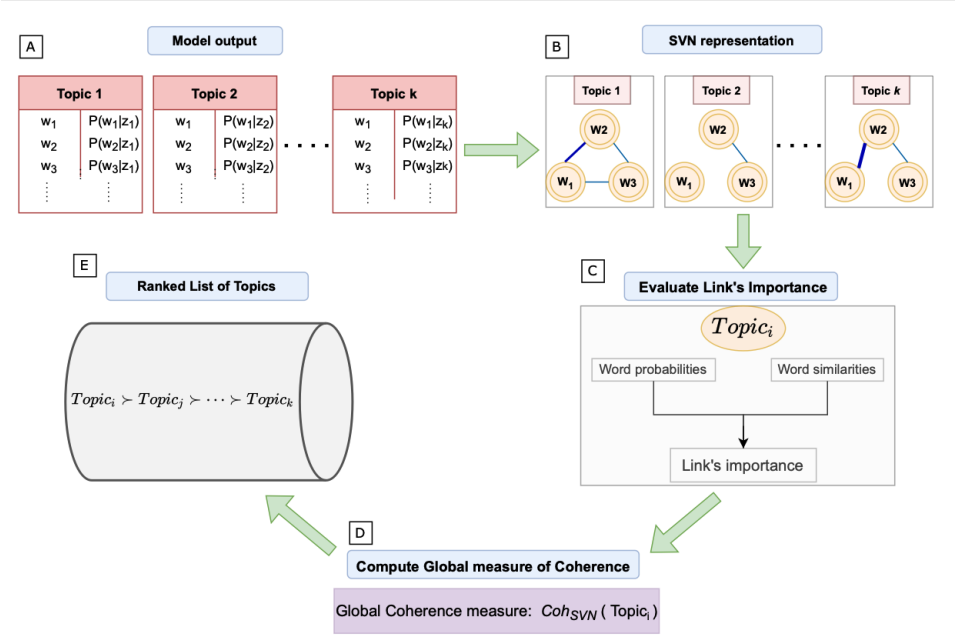


Figure 7.3: Diagram describing the 5 steps of the algorithm.

Regarding the first step, the specific topic model used, the parameter tuning and the choice of the optimal number of topics lay outside the scope of this chapter. Relevant insights on these subjects can be found in references (Arun et al., 2010; Krasnov and Sen, 2019; Sbalchiero and Eder, 2020; Chuang et al., 2013). The estimation of the LDA model provides a list of K latent topics, each one described by an ordered list of words. So, to conclude the first step, we select the m most probable words[1].
To build the SVN of a given topic, $\frac{m(m-1)}{2}$ statistical tests (against the null hypothesis of random co-occurrence) are performed, one for each pair of words.
The results are K weighted Statistically Validated Networks with m nodes and a number of links equal to the number tests that rejects the null hypothesis of random co-occurrence at a given level, α, of statistical significance, after the Bonferroni correction for multiple hypothesis testing. An example is shown in Figure 7.4.

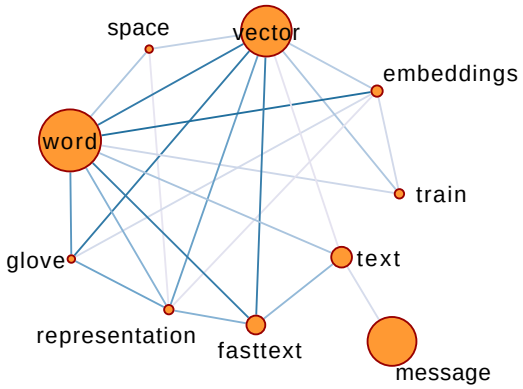


Figure 7.4: Statistically Validated Network of an artificial topic.

The size of each node i in Figure 7.4 is proportional to the probability $P(w_i|z_k)$ that the corresponding word w_i appears in the topic z_k, while the opacity of each link is proportional to the strength of the association between the linked words.
To compute the strength of each validated link, we use corpus-based word similarities within distributional contexts. Specifically, let N denote the total number of sentences in the corpus, N_i and N_j the occurrences of words w_i and w_j, respectively, in the sentences of the corpus, and N_{ij} their co-occurrence. To calculate word similarities we use four metrics already used in other studies. Specifically:

[1] In the present application, we follow the standard approach of setting $m = 10$.

- S_1: Jaccard similarity index (Real and Vargas, 1996)

$$J(w_i, w_j) = \frac{N_{ij}}{N_i + N_j - N_{ij}} \tag{7.5}$$

- S_2: Dice-Sorensen coefficient (Dice, 1945) [2]

$$Dc(w_i, w_j) = \frac{2N_{ij}}{N_i + N_j} \tag{7.6}$$

- S_3: Sokal and Sneath coefficient (Sokal et al., 1963)

$$SS(w_i, w_j) = \frac{N_{ij}}{2N_i + 2N_j - 3N_{ij}} \tag{7.7}$$

- S_4: Fowlkes–Mallows index (Fowlkes and Mallows, 1983)

$$FM(w_i, w_j) = \sqrt{\frac{N_{i,j}^2}{N_i \cdot N_j}}. \tag{7.8}$$

Furthermore, we also consider three metrics that are tightly related to the SVN method. These metrics are:

- S_5: Similarity based on the Pearson's correlation coefficient $\rho(w_i, w_j)$:

$$D_\rho(w_i, w_j) = \frac{1}{2}\left[1 + \rho(w_i, w_j)\right] \tag{7.9}$$

  where

$$\rho(w_i, w_j) = \frac{N_{ij} - \frac{N_i N_j}{N}}{\sqrt{N_i(1 - \frac{N_i}{N})N_j(1 - \frac{N_j}{N})}}.$$

  Since the expected value of the Hypergeometric distribution $H(X|N, N_i, N_j)$ is $\frac{N_i N_j}{N}$ and the variance $\mathbb{V}[X] = \sigma_H^2 = \frac{N_i N_j}{N}\frac{N-N_i}{N}\frac{N-N_j}{N}$, it turns out that $\rho(w_i, w_j)$ is proportional to the Z-score of N_{ij} under the null hypothesis[3].

- S_6: Normalized logarithmic robustness $\tilde{R}$

$$\tilde{R}(w_i, w_j) = \frac{\log_{10}(N) - \log_{10}(N^*|w_i, w_j)}{\log_{10}(N) - \log_{10}(n^*|w_i, w_j)}, \tag{7.10}$$

[2] Notice that it is equivalent to F1 score.

[3] The constant of proportionality is $N^{-\frac{1}{2}}$.

where

$$N^* = \min\{N : p_v(N_{ij}) < \frac{\alpha}{T}\},$$

is defined as the minimum number of sentences needed in the corpus to validate the co-occurrence between w_i and w_j. While,

$$n^* = \min\{N : p_v(N^*_{ij}) < \frac{\alpha}{T}\}$$

is the minimum value of sentences needed to validate the co-occurrence between w_i and w_j assuming a perfect co-occurrence, $N^*_{ij} = \min(N_i, N_j)$.

- S_7: Similarity based on the normalized p-value $\tilde{p}_v$

$$\tilde{p}_v(w_i, w_j) = 1 - \frac{p_v(N_{ij}|N_i, N_j, N)}{\alpha/T}, \tag{7.11}$$

where $p_v(N_{ij}|N_i, N_j, N)$ is computed following Eq.7.4.

All of the proposed similarity measures, $\{S_1, \ldots, S_7\}$, take values in the range $[0, 1]$ where 0 indicates two totally unrelated words, while 1 indicates two perfectly associated words.
Given a validated link between two words, say w_i and w_j, belonging to the topic z_k, we define the link's importance $Imp(w_i, w_j|z_k)$:

$$Imp(w_i, w_j|z_k) = \sqrt{P(w_i|z_k)P(w_j|z_k)}\, S_h(w_i, w_j), \tag{7.12}$$

where S_h is one of the similarity function described above: $\{D_\rho, \tilde{R}, \tilde{p}_v, J, Dc, SS, FM\}$. The importance of a validated link (Eq. 7.12), between w_i and w_j give a topic z_k, takes into account two components:

- the relative relevance of w_i and w_j within z_k:

$$\sqrt{P(w_i|z_k)P(w_j|z_k)};$$

- the strength of the association between w_i and w_j:

$$S_h(w_i, w_j),\ h = 1, \ldots, 7.$$

The conditional probabilities $P(w_i|z_k)$ and $P(w_j|z_k)$ reflect the relevance of words w_i

and w_j, respectively, within the topic z_k. That is, words with a higher probability are more relevant within a topic. Therefore, the more relevant the two terms, the more important the validated link between them. We decided to use the geometric mean of $P(w_i|z_k)$ and $P(w_j|z_k)$ as aggregating function to reduce the impact of the distribution's tails. As regards to $S_h(w_i,w_j)$, it measures the association between w_i and w_j. Intuitively, the higher the association between two words, the greater the importance of the link between them.

Note that, if w_i and w_j exhibit a "perfect" co-occurence, i.e., $N_i = N_j = N_{ij}$, then $S_h(w_i,w_j) = 1$ and the link's importance reduces to $Imp(w_i,w_j|z_k) = \sqrt{P(w_i|z_k)P(w_j|z_k)}$, that is, the geometric mean of the words probabilities, given the topic, provided by the model.

Finally, we define the **global coherence** measure of a topic, z_k, as:

$$Coh_{SVN}(z_k) = \frac{\sum\limits_{w_i \neq w_j, \in \mathscr{L}} Imp(w_i,w_j|z_k)}{\sum\limits_{w_i \neq w_j, \in \Omega_k} \sqrt{P(w_i|z_k)P(w_j|z_k)}}, \tag{7.13}$$

where $\mathscr{L}$ is the set of word pairs linked in the SVN, while Ω_k is the set of all possible $m \cdot (m-1)/2$ word pairs for topic z_k.

In Eq.7.13, the denominator represents the coherence of a perfectly coherent topic, that is, a fully connected network where all the pairwise word similarities are maximized, i.e. $S_h(w_i,w_j) = 1 \; \forall w_i,w_j \in \Omega_k$. Thus, $Coh_{SVN}(z_k)$ ranges in the set $[0,1]$, where the minimum value indicates a totally incoherent and unintelligible topic, while a value of 1 represents a perfectly coherent topic.

Measure $Coh_{SVN}(z_k)$ allows us to rank topics in decreasing order of coherence, which completes the fifth (and final) step of the procedure presented in this section.

7.4 Experimental evaluation

7.4.1 Dataset and pre-processing

We evaluated our estimator of topic quality on a dataset of articles extracted from the *New York Times*, which was already analysed by Xing et al. (2019). The dataset (NYTd from now on) consists of 8,764 articles of the *New York Times*, which appeared between April and July 2016[4].

In particular, we decided to consider a reduced version of this dataset, obtained by

[4] https://www.kaggle.com/nzalake52/new-york-times-articles

removing all the articles with fewer than 20 total words (Hong and Davison (2010) discuss how short documents can confuse topic modeling algorithms), and taking a random sample of size 1,000 out of those remaining.

The following step is to perform data preprocessing in order to reduce noise from the data. The preprocessing usually consists of tasks such as tokenization, filtering, and either lemmatization or stemming. Tokenization means transforming sentences in a list of words, called token, and the filtering step implies removing all punctuation and numbers. Lemmatization and stemming are two text normalization techniques for Natural Language Processing. The first one is the process of finding the base or dictionary form of a word, called *lemma*, with the aim to remove only inflectional endings considering morphological analysis as meaning and context. Instead, stemming is a method to convert words into their root form by cutting the suffix or prefix from the word. Comparing the lemmatization and stemming methods, we opted for the lemmatization. Stemmed words, in general, are very complicated to interpret, since roots of words were insufficient to discriminate among alternative meanings (Schütze et al., 2008). For instance, the word *better* has *good* as its lemma, but this link is missed by stemming. We removed urls, mails, punctuation and numbers from the texts through the `Python` `regex` function. Then, we transformed uppercase letters into lowercase letters and removed accents. Furthermore, we used the `gensim` library to construct compound words, such as *United_States* or *North_Korea*, and `spaCY`, an open-source natural language processing library for Python, to split up sentences. Finally, we removed i) infrequently used words (i.e. appearing only once per document); and ii) redundant words (a rule of thumb is to remove terms appearing in more than 80% of the documents). As a matter of facts, infrequently used terms will not contribute much information about topics, while discovery and removing them may greatly reduce the size of the vocabulary (Denny and Spirling, 2018). Equally, it has been shown that redundant words appearing frequently do not convey any meaningful message for topic modeling (Bastani et al., 2019).

The original corpus dictionary, as directly obtained from the 1,000 articles, consisted of $28,104$ tokens, whereas the final corpus (after data preprocessing) included $8,770$ tokens.

The LDA model was trained in `R` setting 50 topics (Waldherr et al., 2015), then we randomly extracted 30 of them for human judgment evaluation.

We have chosen to use only part of the group of estimated topics due to time constraints. Indeed, we structured the questionnaire so that each annotator took, on average, 15 minutes to complete their task, assuming an average response time of about 30 seconds per topic. This issue is crucial for maximising the quality of the answers

obtained; in fact, a questionnaire which takes too long to be completed entails the risk of receiving unreliable answers as the respondent's focus drops.
Finally, we prepared graphical representations of the networks of topics using `Cytoscape` software.[5]

7.4.2 Coherence-based topic annotations

To obtain high-quality ratings, the survey was structured in two steps. During the first step, which we call "pilot", 23 PhD students from the Department of Economics, Business and Statistics at the University of Palermo, Italy, were brought in. We provided them with 32 topics (consisting of 10 words each) to be evaluated on a 5-point scale where 5="coherent" and 1="not coherent". Among topics, 30 were genuine topics according to the LDA model applied to the New York Times dataset, and the remaining two were synthetic (control) topics. The first synthetic topic included a group of unrelated words that formed a meaningless and incoherent topic, $z_{31} = \{$Lasagna; Finance; Jeans; Buddhist; Pokemon; Drive; Molecule; Sound; Chess; Revolver$\}$. Instead, the second synthetic topic included perfectly coherent words that formed a strongly coherent topic, $z_{32} = \{$Black; White; Red; Green; Pink; Purple; Brown; Yellow; Grey; Blue$\}$.
We also provided textual guidelines on how to judge whether a topic was coherent or incoherent. In addition to showing several examples of such topics we provided the following preliminary instructions to the respondents.

Guidelines

Topic modeling consists of the automatic extraction of groups of words, called *topics*, from a collection of texts. For a topic to be "coherent", it must make sense and be interpretable. This means that the topic's words must:

1. be related to each other
2. belong to the same theme

An automatic procedure for the identification and evaluation of topics is reliable if the topics identified are coherent and interpretable for humans. This is why we are asking you to be part of a benchmark sample of individuals to test the effectiveness of a new topic modeling algorithm we are working on. Therefore, we ask you to rate the coherence of specific topics on a scale of 1 to 5. For ex-

[5] https://cytoscape.org

ample, you can give a topic a low mark if you find few links between the words in it, the mark increases as the number of linked words increases.
It is not always easy to evaluate a list of words, especially if some of them are unfamiliar or belong to a language other than yours (in this case, English). We ask you, *PLEASE*, we ask you to translate any words or nouns you do not know to give as informed a mark as possible.
You will notice that some topics share one or more words; this is not a problem! The topics are not related to each other, so each topic must be evaluated individually. There is no right or wrong answer since we aim to collect your subjective opinion.

The role of the pilot was to assess the topic annotators' ability in understanding their assigned task. We also investigated which improvements were necessary in letting annotators deepen their comprehension of the meaning of "coherence".
The most critical issue in the pilot was to investigate whether an odd scale was appropriate. Thus, we studied the relationship between the percentage of neutral answers given by an annotator (i.e. providing a grade of 3) and their probability of failing at least one control topic evaluation.

Table 7.1: Relationship between giving neutral answers and failing at least one control topic evaluation

	Fail control		
Neutral responses	No	Yes	Total
$\leq 30\%$	14	1	**15**
$> 30\%$	2	6	**8**
Total	**16**	**7**	**23**

Table (7.1) shows that these two features are strongly related since the odds ratio (Schmidt and Kohlmann, 2008) is equal to $\frac{14\times6}{2\times1} = 42$. As a matter of fact, many studies (Pornel and Saldaña, 2013; Taherdoost, 2019) showed that some respondents quickly select the midpoint on the 5-point scale as a dumping ground (Chyung et al., 2017). Such attitude can be explained in psychological terms: *"choosing a minimally acceptable response as soon as it is found, instead of putting effort to find an optimal response"* (Chyung et al., 2017). Therefore, we could easily identify "unreliable annotators" that do not produce reliable judgments, by looking at the respondents who fail the control topics. The results of the pilot survey informed our decision to provide the final survey annotators with the same guidelines, but we asked them to evaluate the

coherence of topics on a scale from 1 to 4 to discourage annotators from expressing neutral responses.

The final survey was designed to obtain human judgments to be used as ground truth for comparing our method with state-of-the-art coherence measures.

The annotators of the final survey were 222 PhD students from various departments of the University of Palermo; in this way, we employed highly educated judges with heterogeneous knowledge within the sample.

The 222 judges were asked to assess the coherence of 32 topics (30 genuine and 2 artificial topics) on a Google Form[6].

Table 7.2 reports the control topics' scores manual assigned by the 222 annotators. Overall, about 90% of the total (202 out of 222 annotators) succeeded in evaluating both control topics. In the case of the highly coherent topic z_{32}, we considered the ratings equal to 4 to "be successful" since a group of words containing only colours should receive the maximum rating. At the same time, we regarded ratings of 1 or 2 as a success for the incoherent coherent topic z_{31}.

Table 7.2: Control topics' scores assigned by annotators, reliable annotators are highlighted in red.

		Topic z_{32} scores				
		1	**2**	**3**	**4**	Tot
Topic z_{31} scores	**1**	1	1	2	192	196
	2	0	1	4	10	15
	3	0	0	2	4	6
	4	0	2	0	3	5
	Tot	1	4	8	209	222

Fig 7.5 reports the frequency distributions of the scores assigned by the annotators to the 30 genuine topics, removing the annotators who failed at least one control topic evaluation.

[6] https://docs.google.com/forms/d/e/1FAIpQLSdoWQsO3MLMcQZDatkCkrSWaThuuj2D-Wm7sR18cy3x8XiRhw/viewform

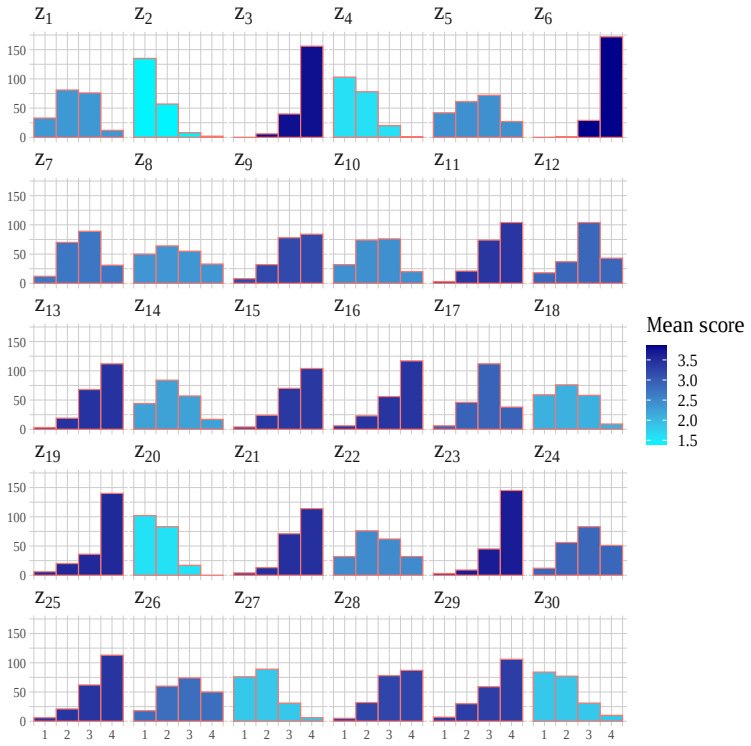


Figure 7.5: Annotators' coherence evaluations

The final dataset contains: i) the list of the most probable words, ii) the coherence ratings given by evaluators, and iii) the document term matrix used in our study. It is available upon request from the authors.

7.4.3 Data analysis and results

To compare the effectiveness of the proposed method in replicating human judgment with respect to the other coherence measures proposed in the literature, we collected the results of the survey and re-arranged them in the form of rankings. Thus, conditioning to a specific coherence metric, the topic with the highest coherence score will be ranked 1 and the topic with the lowest coherence score will be ranked 30.
Therefore, we build two matrices:

- the matrix of scores $\mathbf{S}_{30\times 13}$, where the generic s_{ij} element represent the coherence score of the z_j-topic assigned by the i^{th} metric. As regards the last column,

i.e. human judgment, the z_j-topic is given the average coherence score assigned by human evaluators. (see Tab. 7.3 for a reduced version of the matrix, and Tab. D.2 for the full matrix);

- the matrix of rankings $\mathbf{R}_{30\times13}$, where the generic r_{ij} element represent the relative rank of the z_j-topic assigned by the i^{th} metric. In this matrix, the estimated topic coherences are compared with each other, in order to establish a preference ordering: from the most coherent to the least coherent topic. (see Tab. 7.4 for a reduced version of the matrix, and Tab. D.3 for the full matrix).

Table 7.3: Coherence scores: the **S** matrix

Topic	D_ρ	$\tilde{p}_v$	$\cdots$	HumanJ
z_1	0.076	0.133	$\cdots$	2.332
z_2	0.049	0.084	$\cdots$	1.391
z_3	0.159	0.265	$\cdots$	3.743
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
z_{30}	0.100	0.150	$\cdots$	1.837

Table 7.4: Ranking coherence scores: the **R** matrix

Topic	D_ρ	$\tilde{p}_v$	$\cdots$	HumanJ
z_1	26	26	25	23
z_2	29	29	30	30
z_3	18	18	20	2
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
z_{30}	22	23	28	26

To evaluate the correlation between human judgments and the topic quality scores predicted by all the automatic metrics, we use the Emond and Mason's rank correlation coefficient, τ_x (Emond and Mason, 2002). The higher the τ_x, the better the metric is at measuring topic quality.

In addition, to conforming our comparison procedure to the literature standard, we also computed the Pearson's linear correlation coefficient (Lau et al., 2014; Röder et al., 2015) and the Spearman's rank correlation coefficient (Newman et al., 2010; Aletras and Stevenson, 2013; Morstatter and Liu, 2018), see Tab. D.1 in the appendix. Although these two measures have been frequently used in the literature, we argue that they are not particularly suitable in this framework. On the one hand, the Pearson's correlation coefficient only considers the linear correlation between two vectors, which is undoubtedly restrictive for our purpose, and its value may be seriously affected by only

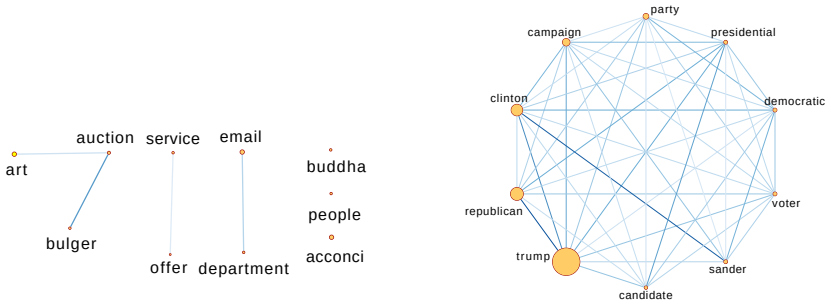
one outlier (Croux and Dehon, 2010). On the other hand, as highlighted in Chapter 1, the Spearman rank correlation suffers from the *sensitivity to irrelevant alternatives*. Moreover, Croux and Dehon (2010) highlighted that the Spearman rank correlation has a smaller gross error sensitivity (GES) (low robustness) and a greater asymptotic variance (AV) (low efficiency) compared to the Kendall τ_b and τ_x. These features make Spearman coefficient a less preferable measure from both perspectives.
Table 7.5 reports the τ_x rank correlation between human judgments and all the considered metrics. We compared the correlations obtained either by keeping ("with noise" column of Tab 7.5) or removing ("without noise" column of Tab 7.5) the unreliable annotators. The results show that the proposed SVN Coherence measure, based on D_ρ, outperforms all the baselines.

Table 7.5: Emond and Mason τ_x rank correlation coefficient with human judgments for metrics.

Method	*Correlation with human judgement* τ_x with noise	τ_x without noise
Coh_{SVN}		
J	0.621	0.632
Dc	0.616	0.627
SS	0.616	0.627
FM	0.708	0.714
D_ρ	**0.721**	**0.728**
$\tilde{R}$	0.579	0.586
$\tilde{p}_v$	0.698	0.705
State-of-the-art		
PMI (Newman et al., 2009)	0.616	0.618
$UMass$ (Mimno et al., 2011)	0.565	0.563
NPMI (Lau et al., 2014)	**0.685**	**0.687**
CV (Röder et al., 2015)	0.570	0.572
$tf\text{-}idf$ (Nikolenko et al., 2017)	0.629	0.636

7.4.4 Interpretation of the resulting topics

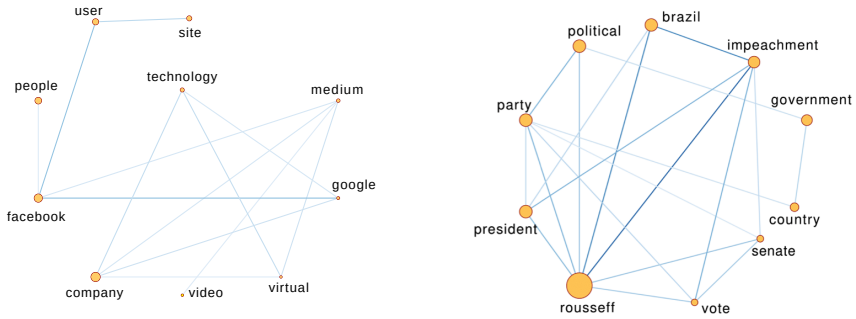
In this section, we report a comparison between Coh_{SVN} and human judgment in evaluating the coherence of some estimated topics. In Fig. 7.6, topics for which there is high concordance between human judgement and Coh_{SVN} are reported.



(a) *Topic* z_2 (b) *Topic* z_6

Figure 7.6: SVN representation of Topic z_2 and Topic z_6

In particular, Fig.7.6(a) represents topic z_6, which is the most coherent topic. It has been assigned an average score equal to 3.84 (first in the rank) by the annotators. Likewise, Coh_{SVN} scores it 0.545, which make it the most coherent in the final ranking. As a matter of fact, topic z_6 can be considered a genuine theme of the domain, i.e., a politically themed topic where all the top words can be associated with US politics. Therefore, the annotators quickly recognized that the words are strongly related, and the co-occurrences in the corpus reflect their solid semantic association.
Topic z_2, in fig. 7.6(b), is one of the least coherent topics. Annotators rated it with an average score of 1.37 (last position in the ranking). Besides, the topic's Coh_{SVN} score is equal to 0.049, which corresponds to the second-to-last position in the ranking.
The SVN constructed on topic z_2 reveals that the words composing it are mostly unrelated; therefore, there are few statistically validated links.
Fig. 7.7 report topics whose scores (and, consequently, the ranking) assigned by the annotators are not consistent with our coherence measure.



(a) *Topic* z_3 (b) *Topic* z_{28}

Figure 7.7: SVN representation of Topic z_3 and Topic z_{28}

Topic z_3 (fig. 7.7(a)) has been positively evaluated by the annotators; the average score is equal to 3.74 (second in the ranking). Instead, Coh_{SVN} places it 18th in the ranking, with a score equal to 0.159

The annotators considered the words in topic z_3 to be related to each other, but the semantic associations detected by humans are not reflected by the co-occurrences in the reference corpus. For example, the words *Facebook* and *company* are not linked in the resulting Statistically Validated Network. This issue could be due to the structure of the corpus used in the analysis. As a matter of fact, the statistical significance of word pairs' co-occurrences can also be validated including external text data sources, such as Wikipedia or Google hits, rather than using only the corpus sentences. Alternatively, one could use paragraphs instead of sentences to count co-occurrences, but if the text is not properly formatted it might prove difficult to identify the paragraphs.

Finally, topic z_{28} is reported in fig. 7.7(b). The corresponding Coh_{SVN} score is equal to 0.264, the 7th in ranking. While, according to the survey, it has an average score equal to 3.22, and it is 12th in ranking. In this case, the topic is considered to be more coherent by Coh_{SVN} than by humans; however, the discrepancy between the automatic measure and the human judgement is less relevant than in the previous case. Overall, about 20% of the annotators did not recognise a central theme and rated it with a low score (1 or 2). This issue could be since topic z_{28} refers to a specific political event that took place in Brazil between 2015 and 2016. Moreover, it contains "hard-to-interpret" terms such as *Rousseff*, a little-known proper name, and *impeachment*, a technical term referring to the political sphere. Indeed, the evaluation of the topic is more complex than the other ones and requires respondents to carry out in-depth research.

7.4.5 Summary of main findings

In summary, according to the presented analysis, Coh_{SVN} represents a new topic coherence measure that:

- follows a rigorous statistical model of co-occurrence based on multiple hypotheses testing, while state-of-the-art measures pass over the randomness of co-occurrence;
- ranges between $[0, 1]$, providing a more readable framework for evaluating the coherence of the topics;
- approximates human ratings better than state-of-the-art methods (see Tab. 7.5);
- allows the graphical representation and interpretation of the obtained topics through Statistically Validated Networks (SVNs)(Tumminello et al., 2011);
- is less sensitive to the text preparation since it considers co-occurrences of word pairs in sentences. Instead, most of the measures proposed in the literature, as summarised in the chapter by Röder et al. (2015), use a sliding window to calculate the co-occurrences, which makes these methods very sensitive to the preprocessing steps.

7.5 Concluding remarks

One of the fundamental challenges in topic detection models is assessing the semantic *coherence* of estimated topics in terms of human interpretability. State-of-the-art coherence measures focus on the marginal probabilities of words and their co-occurrence. However, none of them takes into account the randomness of co-occurrences. In this work, we undertake a rigorous statistical approach based on hypotheses testing to develop a new topic-coherence measure, Coh_{SVN}.
To automatically evaluate how semantically close the top words of the topics are, we represent each topic as a weighted network of its most probable words. The presence of a link between two words indicates that their co-occurrence in sentences is statistically significant against the null hypothesis of random co-occurrence.
The proposed global measure of coherence, Coh_{SVN}, is derived by considering the number of statistically validated links, the strength of the association between word pairs, and the relative relevance of each word in the topic. To prove the effectiveness of our method, we administered a survey on 222 PhD students from University of Palermo,

Italy, and construct a benchmark dataset of human judgements. These judgments were taken as ground truth, and it was shown that the proposed measure reproduces human judgment more closely than the state-of-the-art (Table 7.5). As for future research, the results reported in this chapter suggest to explore the possibility to develop a topic similarity index based on Statistically Validated Networks and including NLP tools, e.g., entity recognition and part-of-speech tagging. Finally, the development of a rigorous statistical method for validating the similarity between two topics could prove beneficial, following the theory of recommendation systems (Zhou et al., 2010), to promote *diversity* in the final ranking of topics. Indeed, the ordered list of topics could be determined by considering both the point-wise quality score (Coh_{SVN}) and the correlations between topics.

Chapter 8

Conclusions

The thesis has focused on developing distance-based methods for preference rankings and preference approvals and on the definition of a new ranking method for textual analysis. After a brief review of rankings, the following two chapters addressed two important issues concerning preference rankings, such as aggregation and prediction, by considering the weight of items in a ranking and the similarity between them. Specifically, Chapter 3 has provided an element weighted rank correlation coefficient $\tau_{x,e}$ for linear, weak, and incomplete orderings. The correspondence between $\tau_{x,e}$ and the corresponding weighted Kemeny distance $d_{K,e}$ was analytically proved. Additionally, we have demonstrated that, when all items are given equal weights, the weighted rank distance, denoted by $d_{K,e}$, is proportional to the well-known Kemeny distance, denoted by d_K, while the correlation coefficient, denoted by $\tau_{x,e}$, is equal to the Emond and Mason's τ_x. Then, we have built an algorithm to perform weighted aggregation of preferences using the proposed weighted measures. From the simulation study and the real data examples, we have demonstrated that the algorithm allows us to find the true consensus and to show how the weighting vector affects the representativity of the median ranking. The weighted consensus algorithm's computational effort was investigated by considering some simulations.

Chapter 4 has proposed an item-weighted algorithm to perform weighted ranking prediction. Specifically, three item-weighted versions (AdaBoost.R.M1, AdaBoost.R.M2, and AdaBoost.R.M3) of the boosting algorithm AdaBoost for Label Ranking have been defined. The algorithms combine many weighted distance-based trees for ranking data to obtain a flexible, strong learner. The three methods were compared, investigating their performance through real and simulated data applications. In particular, we have demonstrated that AdaBoost.R.M3 performs best in many scenarios, having the lowest

prediction errors even at a high noise level. Chapters 5 and 6 have shifted the attention on preference-approvals, an extension of the classical preference model. Preference-approvals are complex structures that combine preference rankings and approval voting for declaring opinions over a set of alternatives. Chapter 5 has introduced a new distance for preference-approvals, D^r_λ, following the Kemeny approach. In order to define a new distance given two preference-approvals, we introduced two indices that quantify the disagreement between two voters for each pair of alternatives as well as an aggregation function belonging to the class of weighted power means. The new distance depends on two parameters. The effect of these parameters on the distance was analyzed and described through some heatmaps. The proposed distance was used to study the universe of preference-approvals and determine clusters of voters. Some dendrograms and cophenetic correlations were used to demonstrate how the two parameters characterizing the distance affect the clustering process. In addition, we have shown that the new distance family offers some advantages compared to the existing distance function. Specifically, through a simulation study and the adjusted Rand index, we have proved that D^r_λ with $r \neq 1$ allows the true clustered structure of data to be found more accurately. Similarly, through a cluster-wise stability index, we have shown that D^r_λ with $r \neq 1$ produces more stable clusters on the real data example.

Chapter 6 has proposed a new method for clustering alternatives in preference-approvals. First, we have introduced a family of pseudometrics, δ_λ, able to quantify the distance between alternatives based on two main components: the *individual preference-discordance* ρ_{ij} and the *individual approval-discordance* α_{ij}, and on the λ parameter, which regulates the weight to give to each component. To obtain clusters, we have applied the Ranked k-medoids partitioning algorithm, taking the similarities among pairs of alternatives as input based on the proposed pseudometrics. Finally, clusters were represented in 2-dimensional space using Non-Metric Multidimensional Scaling. Through two applications to real data, we have demonstrated how our algorithm allows dividing a heterogeneous population of alternatives into homogeneous groups, reducing the complexity of the preference-approval space and providing a more accessible interpretation of data. The impact of the λ parameter on cluster identification and visualization has also been demonstrated.

To conclude the thesis, Chapter 7 has developed a ranking method that can be applied in the framework of textual analysis to rank learned topics closely matching human judgments automatically. The ranking method proposed was based on a new topic-coherence measure, Coh_{SVN}, employing Statistically Validated Networks. To prove the effectiveness of our approach, we have administered a survey among 222 PhD students from the University of Palermo, Italy, and constructed a benchmark dataset of human

judgements. These judgments were taken as ground truth, and it was shown that the proposed measure reproduces human judgment more closely than the state-of-the-art.

Bibliography

Abramowitz, M. and Stegun, I. A. (1964). *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, volume 55. US Government printing office.

Aiguzhinov, A., Soares, C., and Serra, A. P. (2010). A similarity-based adaptation of naive bayes for label ranking: Application to the metalearning problem of algorithm recommendation. In *International Conference on Discovery Science*, pages 16–26. Springer.

Albano, A., García-Lapresta, J. L., Plaia, A., and Sciandra, M. (2022a). A family of distances between preference-approvals. *Annals of Operations Research*, pages 1–29.

Albano, A. and Plaia, A. (2021). Element weighted Kemeny distance for ranking data. *Electronic Journal Of Applied Statistical Analysis, 14(1)*, pages 117–145.s.

Albano, A., Sciandra, M., and Plaia, A. (2022b). A weighted distance-based approach with boosted decision trees for label ranking. *Expert Systems with Applications*, 213:119000.

Aledo, J. A., Gámez, J. A., and Molina, D. (2017). Tackling the supervised label ranking problem by bagging weak learners. *Information Fusion*, 35:38–50.

Aletras, N. and Stevenson, M. (2013). Evaluating topic coherence using distributional semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013)–Long Papers*, pages 13–22.

Alfaro, E., Gámez, M., and Garcia, N. (2013). adabag: An r package for classification with boosting and bagging. *Journal of Statistical Software*, 54(2):1–35.

Ali, A. and Meilă, M. (2012). Experiments with Kemeny ranking: What works when? *Mathematical Social Sciences*, 64(1):28–40.

Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E. D., Gutierrez, J. B., and Kochut, K. (2017). A brief survey of text mining: Classification, clustering and extraction techniques. *arXiv preprint arXiv:1707.02919*.

AlSumait, L., Barbará, D., Gentle, J., and Domeniconi, C. (2009). Topic significance ranking of LDA generative models. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 67–82. Springer.

Amodio, S., D'Ambrosio, A., and Siciliano, R. (2016). Accurate algorithms for identifying the median ranking when dealing with weak and partial rankings under the Kemeny axiomatic approach. *European Journal of Operational Research*, 249(2):667–676.

Arrow, K. J. (1951). *Social choice and individual values*. John Wiley & Sons.

Arun, R., Suresh, V., Madhavan, C. V., and Murthy, M. N. (2010). On finding the natural number of topics with latent dirichlet allocation: Some observations. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 391–402. Springer.

Barokas, G. and Sprumont, Y. (2022). The broken Borda rule and other refinements of approval ranking. *Social Choice and Welfare*, 58(1):187–199.

Bartholdi, J., Tovey, C. A., and Trick, M. A. (1989). Voting schemes for which it can be difficult to tell who won the election. *Social Choice and welfare*, 6(2):157–165.

Bastani, K., Namavari, H., and Shaffer, J. (2019). Latent dirichlet allocation (lda) for topic modeling of the cfpb consumer complaints. *Expert Systems with Applications*, 127:256–271.

Beliakov, G., Pradera, A., and Calvo, T. (2007). *Aggregation Functions: A guide for practitioners*. Springer.

Berkhin, P. (2006). A survey of clustering data mining techniques. In *Grouping multidimensional data*, pages 25–71. Springer.

Black, D. (1976). Partial justification of the Borda count. *Public Choice*, 28(1):1–15.

Blei, D. and Lafferty, J. (2006). Correlated topic models. *Advances in neural information processing systems*, 18:147.

Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, 55(4):77–84.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Borda, J.-C. d. (1781). Mémoire sur les élections au scrutin: Histoire de l'académie royale des sciences. *Paris, France*, 12.

Bouma, G. (2009). Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, pages 31–40.

Boyd-Graber, J. L., Hu, Y., Mimno, D., et al. (2017). *Applications of topic models*, volume 11. now Publishers Incorporated.

Brams, S. J. (2008). Mathematics and democracy: Designing better voting and fair-division procedures. *Mathematical and Computer Modelling*, 48(9):1666–1670. Mathematical Modeling of Voting Systems and Elections: Theory and Applications.

Brams, S. J. and Fishburn, P. C. (1978). Approval voting. *American Political Science Review*, 72(3):831–847.

Brams, S. J. and Sanver, M. R. (2009). *Voting Systems that Combine Approval and Preference*, pages 215–237. Springer Berlin Heidelberg, Berlin, Heidelberg.

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

Breiman, L. (1998). Arcing classifiers. *The Annals of Statistics*, 26(3):801–824.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

Can, B. (2014). Weighted distances between preferences. *Journal of Mathematical Economics*, 51:109–115.

Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., and Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems*, pages 288–296.

Chao, X., Dong, Y., Kou, G., and Peng, Y. (2021). How to determine the consensus threshold in group decision making: a method based on efficiency benchmark using benefit and cost insight. *Annals of Operations Research*, pages 1–35.

Cheng, W., Dembczynski, K., and Hüllermeier, E. (2010). Label ranking methods based on the plackett-luce model. In *ICML*.

Cheng, W., Hühn, J., and Hüllermeier, E. (2009). Decision tree and instance-based learning for label ranking. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 161–168.

Cheng, W. and Hüllermeier, E. (2008). Instance-based label ranking using the mallows model. In *ECCBR Workshops*, pages 143–157.

Cheng, W. and Hüllermeier, E. (2009). A new instance-based label ranking approach using the mallows model. In *International Symposium on Neural Networks*, pages 707–716. Springer.

Chuang, J., Gupta, S., Manning, C., and Heer, J. (2013). Topic model diagnostics: Assessing domain relevance via topical alignment. In *International conference on machine learning*, pages 612–620. PMLR.

Chyung, S. Y., Roberts, K., Swanson, I., and Hankinson, A. (2017). Evidence-based survey design: The use of a midpoint on the likert scale. *Performance Improvement*, 56(10):15–23.

Cohen, W. W., Schapire, R. E., and Singer, Y. (1999). Learning to order things. *Journal of artificial intelligence research*, 10:243–270.

Condorcet, J. A. N. (1785). *Essai Sur L'Application de L'Analyse a la Probabilite Des Decisions Rendues a la Pluralite Des Voix*. Kessinger Publishing.

Cook, W. D. (2006). Distance-based and ad hoc consensus models in ordinal preference ranking. *European Journal of operational research*, 172(2):369–385.

Cook, W. D., Golany, B., Penn, M., and Raviv, T. (2007). Creating a consensus ranking of proposals from reviewers' partial ordinal rankings. *Computers & Operations Research*, 34(4):954–965.

Cook, W. D., Kress, M., and Seiford, L. M. (1986). An axiomatic approach to distance on partial orderings. *RAIRO-Operations Research*, 20(2):115–122.

Cook, W. D. and Seiford, L. M. (1978). Priority ranking and consensus formation. *Management Science*, 24(16):1721–1732.

Cook, W. D. and Seiford, L. M. (1982). On the Borda-Kendall consensus method for priority ranking problems. *Management Science*, 28(6):621–637.

Cook, Wade D., Kress, Moshe, and Seiford, Lawrence M. (1986). An axiomatic approach to distance on partial orderings. *RAIRO-Oper. Res.*, 20(2):115–122.

Croux, C. and Dehon, C. (2010). Influence functions of the spearman and kendall correlation measures. *Statistical methods & applications*, 19(4):497–515.

D'Ambrosio, A. (2021). *ConsRank: Compute the Median Ranking(s) According to the Kemeny's Axiomatic Approach*. R package version 2.1.2.

D'Ambrosio, A., Amodio, S., and Iorio, C. (2015). Two algorithms for finding optimal solutions of the Kemeny rank aggregation problem for full rankings. *Electronic Journal of Applied Statistical Analysis*, 8(2):198–213.

D'Ambrosio, A. and Heiser, W. J. (2016). A recursive partitioning method for the prediction of preference rankings based upon Kemeny distances. *psychometrika*, 81(3):774–794.

D'Ambrosio, A., Mazzeo, G., Iorio, C., and Siciliano, R. (2017a). A differential evolution algorithm for finding the median ranking under the Kemeny axiomatic approach. *Computers & Operations Research*, 82:126–138.

D'Ambrosio, A., Mazzeo, G., Iorio, C., and Siciliano, R. (2017b). A differential evolution algorithm for finding the median ranking under the Kemeny axiomatic approach. *Computers & Operations Research*, 82:126–138.

David, F. N. and Barton, D. E. (1962). *Combinatorial Chance*. Hafner, New York.

de Sá, C. R., Duivesteijn, W., Azevedo, P., Jorge, A. M., Soares, C., and Knobbe, A. (2018). Discovering a taste for the unusual: exceptional models for preference mining. *Machine Learning*, 107(11):1775–1807.

De Sá, C. R., Soares, C., Jorge, A. M., Azevedo, P., and Costa, J. (2011). Mining association rules for label ranking. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 432–443. Springer.

de Sá, C. R., Soares, C., Knobbe, A., and Cortez, P. (2017). Label ranking forests. *Expert systems*, 34(1):e12166.

Dekel, O., Singer, Y., and Manning, C. D. (2003). Log-linear models for label ranking. *Advances in neural information processing systems*, 16:497–504.

Denny, M. J. and Spirling, A. (2018). Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it. *Political Analysis*, 26(2):168–189.

Dery, L. and Shmueli, E. (2020). Boostlr: a boosting-based learning ensemble for label ranking tasks. *IEEE Access*, 8:176023–176032.

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302.

Dieng, A. B., Ruiz, F. J., and Blei, D. M. (2019). The dynamic embedded topic model. *arXiv preprint arXiv:1907.05545*.

Dieng, A. B., Ruiz, F. J., and Blei, D. M. (2020). Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Dong, Y., Li, Y., He, Y., and Chen, X. (2021). Preference–approval structures in group decision making: Axiomatic distance and aggregation. *Decision Analysis*, 18(4):273–295.

Drucker, H. (1997). Improving regressors using boosting techniques. In *ICML*, volume 97, pages 107–115. Citeseer.

Dummett, M. (1984). *Voting Procedures*. Oxford University Press, Oxford.

Emond, E. J. and Mason, D. W. (2000). *A new technique for high level decision support*. Department of National Defence Canada, Operational Research Division.

Emond, E. J. and Mason, D. W. (2002). A new rank correlation coefficient with application to the consensus ranking problem. *Journal of Multi-Criteria Decision Analysis*, 11(1):17–28.

Erdamar, B., García-Lapresta, J. L., Pérez-Román, D., and Sanver, M. R. (2014). Measuring consensus in a preference-approval context. *Information Fusion*, 17:14–21.

Everitt, B. S., Landau, S., Leese, M., and Stahl, D. (2011). *Cluster Analysis 5th ed.* Wiley.

Fagot, R. F. (1994). An ordinal coefficient of relational agreement for multiple judges. *Psychometrika*, 59(2):241–251.

Feldman, R. and Dagan, I. (1995). Knowledge discovery in textual databases (kdt). In *KDD*, volume 95, pages 112–117.

Fisher, R. A. and Yates, F. (1953). *Statistical tables for biological, agricultural and medical research*. Hafner Publishing Company.

Fligner, M. A. and Verducci, J. S. (1986). Distance based ranking models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 48(3):359–369.

Fligner, M. A. and Verducci, J. S. (1990). Posterior probabilities for a consensus ordering. *Psychometrika*, 55(1):53–63.

Fowlkes, E. B. and Mallows, C. L. (1983). A method for comparing two hierarchical clusterings. *Journal of the American statistical association*, 78(383):553–569.

Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*, ICML'96, page 148–156, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139.

Fürnkranz, J. and Hüllermeier, E. (2010). Preference learning and ranking by pairwise comparison. In *Preference learning*, pages 65–82. Springer.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., and Herrera, F. (2011). A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):463–484.

Galili, T. (2015). Dendextend: an `R` package for visualizing, adjusting and comparing trees of hierarchical clustering. *Bioinformatics*, 31(22):3718–3720.

García-Lapresta, J. L. and Pérez-Román, D. (2010). Consensus measures generated by weighted Kemeny distances on weak orders. In *2010 10th International Conference on Intelligent Systems Design and Applications*, pages 463–468. IEEE.

García-Lapresta, J. L. and Pérez-Román, D. (2011). Measuring consensus in weak orders. In Herrera-Viedma, E., García-Lapresta, J. L., Kacprzyk, J., Fedrizzi, M., Nurmi, H., and Zadrożny, S., editors, *Consensual Processes*, pages 213–234. Springer.

García-Lapresta, J. L. and Pérez-Román, D. (2017). A consensus reaching process in the context of non-uniform ordered qualitative scales. *Fuzzy Optimization and Decision Making*, 16(4):449–461.

Genova, V. G., Tumminello, M., Enea, M., Aiello, F., and Attanasio, M. (2019). Student mobility in higher education: Sicilian outflow network and chain migrations. *Electronic Journal of Applied Statistical Analysis*, 12(4):774–800.

González del Pozo, R., García-Lapresta, J. L., and Pérez-Román, D. (2017). Clustering us 2016 presidential candidates through linguistic appraisals. In *Advances in Fuzzy Logic and Technology 2017*, pages 143–153. Springer.

Good, I. (1980). The number of orderings of n-candidates when ties and omissions are both allowed. *Journal of Statistical Computation and Simulation*, 10(2):159–159.

Grabisch, M., Marichal, J.-L., Mesiar, R., and Pap, E. (2009). *Aggregation Functions*, volume 127. Cambridge University Press.

Grbovic, M., Djuric, N., and Vucetic, S. (2012). Learning from pairwise preference data using gaussian mixture model. *Preference Learning: Problems and Applications in AI*, 33.

Hamming, R. W. (1950). Error detecting and error correcting codes. *The Bell System Technical Journal*, 29(2):147–160.

Har-Peled, S., Roth, D., and Zimak, D. (2003). Constraint classification for multiclass classification and ranking. *Advances in neural information processing systems*, pages 809–816.

Hardy, G. H., Littlewood, J. E., Pólya, G., and Pólya, G. (1952). *Inequalities*. Cambridge university press.

Harper, F. M. and Konstan, J. A. (2015). The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media.

Heiser, W. J. (2004). Geometric representation of association between categories. *Psychometrika*, 69(4):513–545.

Heiser, W. J. and D'Ambrosio, A. (2013). Clustering and prediction of rankings within a Kemeny distance framework. In Lausen, B., Van den Poel, D., and Ultsch, A., editors, *Algorithms from and for Nature and Life*, pages 19–31, Cham. Springer International Publishing.

Hennig, C. (2007). Cluster-wise assessment of cluster stability. *Computational Statistics & Data Analysis*, 52(1):258–271.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57.

Hong, L. and Davison, B. D. (2010). Empirical study of topic modeling in twitter. In *Proceedings of the first workshop on social media analytics*, pages 80–88.

Hothorn, T. and Everitt, B. S. (2006). *A Handbook of Statistical Analyses Using R*. Chapman and Hall/CRC.

Hoyle, A., Goel, P., Hian-Cheong, A., Peskov, D., Boyd-Graber, J., and Resnik, P. (2021). Is automated topic model evaluation broken? the incoherence of coherence. *Advances in Neural Information Processing Systems*, 34:2018–2033.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2(1):193–218.

Hüllermeier, E., Fürnkranz, J., Cheng, W., and Brinker, K. (2008). Label ranking by learning pairwise preferences. *Artificial Intelligence*, 172(16-17):1897–1916.

Jaccard, P. (1901). Distribution de la flore alpine dans le bassin des dranses et dans quelques regions voisines. *Bulletin de la Société Vaudoise des Sciences Naturelles*, pages 241–272.

Jacques, J. and Biernacki, C. (2014). Model-based clustering for multivariate partial ranking data. *Journal of Statistical Planning and Inference*, 149:201–217.

Jacques, J., Grimonprez, Q., and Biernacki, C. (2014). Rankcluster: An R Package for Clustering Multivariate Partial Rankings. *The R Journal*, 6(1):101–110.

Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM Computing Surveys (CSUR)*, 31(3):264–323.

Kamwa, E. (2019). Condorcet efficiency of the preference approval voting and the probability of selecting the condorcet loser. *Theory and Decision*, 87(3):299–320.

Kaya, B. (2020). Hotel recommendation system by bipartite networks and link prediction. *Journal of Information Science*, 46(1):53–63.

Kearns, M. and Valiant, L. (1994). Cryptographic limitations on learning boolean formulae and finite automata. *Journal of the ACM (JACM)*, 41(1):67–95.

Kemeny, J. G. (1959). Mathematics without numbers. *Daedalus*, 88(4):577–591.

Kemeny, J. G. and Snell, J. (1962a). *Mathematical Models in the Social Sciences*. Blaisdall Publishing Company.

Kemeny, J. G. and Snell, L. (1962b). Preference ranking: an axiomatic approach. *Mathematical models in the social sciences*, pages 9–23.

Kendall, M. G. (1948). *Rank Correlation Methods.* Griffin.

Kendall, M. G. and Smith, B. B. (1940). On the method of paired comparisons. *Biometrika*, 31(3/4):324–345.

Krasnov, F. and Sen, A. (2019). The number of topics optimization: Clustering approach. *Machine Learning and Knowledge Extraction*, 1(1):416–426.

Kruger, J. and Sanver, M. R. (2021). An Arrovian impossibility in combining ranking and evaluation. *Social Choice and Welfare*, 57(3):535–555.

Kruskal, J. B. (1964). Nonmetric multidimensional scaling: a numerical method. *Psychometrika*, 29(2):115–129.

Kumar, R. and Vassilvitskii, S. (2010). Generalized distances between rankings. In *Proceedings of the 19th international conference on World wide web*, pages 571–580.

Lau, J. H., Newman, D., and Baldwin, T. (2014). Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539.

Lee, P. H. and Philip, L. (2010). Distance-based tree models for ranking data. *Computational Statistics & Data Analysis*, 54(6):1672–1682.

Li, W. and McCallum, A. (2006). Pachinko allocation: Dag-structured mixture models of topic correlations. In *Proceedings of the 23rd international conference on Machine learning*, pages 577–584.

Liu, T.-Y. (2011). Learning to rank for information retrieval. *Information Retrieval*.

Long, J., Liang, H., Gao, L., Guo, Z., and Dong, Y. (2021). Consensus reaching with two-stage minimum adjustments in multi-attribute group decision making: A method based on preference-approval structure and prospect theory. *Computers & Industrial Engineering*, 158:107349.

Mallows, C. L. (1957). Non-null ranking models. I. *Biometrika*, 44(1/2):114–130.

Marden, J. I. (1996). *Analyzing and modeling rank data*. CRC Press.

McGregor, A., Hall, M., Lorier, P., and Brunskill, J. (2004). Flow clustering using machine learning techniques. In *International workshop on passive and active network measurement*, pages 205–214. Springer.

Miller, J. (1981). Simultaneous statistical inference.

Mimno, D., Wallach, H., Talley, E., Leenders, M., and McCallum, A. (2011). Optimizing semantic coherence in topic models. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 262–272.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2018). *Foundations of machine learning*. MIT press.

Morstatter, F. and Liu, H. (2018). In search of coherence and consensus: Measuring the interpretability of statistical topics. *Journal of Machine Learning Research*, 18(169):1–32.

Newman, D., Karimi, S., and Cavedon, L. (2009). External evaluation of topic models. In *in Australasian Doc. Comp. Symp., 2009*. Citeseer.

Newman, D., Lau, J. H., Grieser, K., and Baldwin, T. (2010). Automatic evaluation of topic coherence. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics*, pages 100–108.

Nikolenko, S. I., Koltcov, S., and Koltsova, O. (2017). Topic modelling for qualitative studies. *Journal of Information Science*, 43(1):88–102.

Omar, M., On, B.-W., Lee, I., and Choi, G. S. (2015). Lda topics: Representation and evaluation. *Journal of Information Science*, 41(5):662–675.

Ostasiewicz, S. and Ostasiewicz, W. (2000). Means and their applications. *Annals of Operations Research*, 97(1):337–355. Copyright - Copyright Kluwer Academic Publishers 2000; Ultimo aggiornamento - 2021-09-09.

Palomares, I., Estrella, F. J., Martínez, L., and Herrera, F. (2014). Consensus under a fuzzy context: Taxonomy, analysis framework AFRYCA and experimental case of study. *Information Fusion*, 20:252–271.

Paranyushkin, D. (2011). Identifying the pathways for meaning circulation using text network analysis. *Nodus Labs*, 26.

Philip, L., Wan, W. M., and Lee, P. H. (2010). Decision tree modeling for ranking data. In *Preference learning*, pages 83–106. Springer.

Piccarreta, R. (2010). Binary trees for dissimilarity data. *Computational Statistics & Data Analysis*, 54(6):1516–1524.

Plaia, A., Buscemi, S., Fürnkranz, J., and Mencía, E. L. (2021a). Comparing boosting and bagging for decision trees of rankings. *Journal of Classification*, pages 1–22.

Plaia, A., Buscemi, S., and Sciandra, M. (2019). A new position weight correlation coefficient for consensus ranking process without ties. *Stat*, 8(1):e236.

Plaia, A., Buscemi, S., and Sciandra, M. (2021b). Consensus among preference rankings: a new weighted correlation coefficient for linear and weak orderings. *Advances in Data Analysis and Classification*, 15(4):1015–1037.

Plaia, A. and Sciandra, M. (2019). Weighted distance-based trees for ranking data. *Advances in data analysis and classification*, 13(2):427–444.

Plaia, A., Sciandra, M., and Buscemi, S. (2018). Consensus measures for preference rankings with ties: an approach based on position weighted Kemeny distance. *Advances in Statistical Modelling of Ordinal Data*, page 171.

Plaia, A., Sciandra, M., and Muro, R. (2017). Ensemble methods for ranking data. In *CLADAG 2017*, pages 1–6. Universitas Studiorum Srl Casa Editrice.

Pornel, J. B. and Saldaña, G. A. (2013). Four common misuses of the likert scale. *Philippine Journal of Social Sciences and Humanities University of the Philippines Visayas*, 18(2):12–19.

Puccio, E., Vassallo, P., Piilo, J., and Tumminello, M. (2019). Covariance and correlation estimators in bipartite complex systems with a double heterogeneity. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(5):053404.

Ramík, J. and Vlach, M. (2012). Aggregation functions and generalized convexity in fuzzy optimization and decision making. *Annals of Operations Research*, 195(1):261–276.

Ramrakhiyani, N., Pawar, S., Hingmire, S., and Palshikar, G. (2017). Measuring topic coherence through optimal word buckets. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 437–442.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850.

Real, R. and Vargas, J. M. (1996). The probabilistic basis of jaccard's index of similarity. *Systematic biology*, 45(3):380–385.

Ribeiro, G., Duivesteijn, W., Soares, C., and Knobbe, A. (2012). Multilayer perceptron for label ranking. In *International Conference on Artificial Neural Networks*, pages 25–32. Springer.

Röder, M., Both, A., and Hinneburg, A. (2015). Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining*, pages 399–408.

Rodrigo, E. G., Alfaro, J. C., Aledo, J. A., and Gámez, J. A. (2021). Mixture-based probabilistic graphical models for the label ranking problem. *Entropy*, 23(4):420.

Rohlf, F. J. and Fisher, D. R. (1968). Tests for hierarchical structure in random data sets. *Systematic Biology*, 17(4):407–412.

Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523.

Sanver, M. R. (2010). Approval as an Intrinsic Part of Preference. In Laslier, J.-F. and Sanver, M. R., editors, *Handbook on Approval Voting*, Studies in Choice and Welfare, pages 469–481. Springer.

Saraçli, S., Doğan, N., and Doğan, İ. (2013). Comparison of hierarchical cluster analysis methods by cophenetic correlation. *Journal of Inequalities and Applications*, 2013(1):1–8.

Sbalchiero, S. and Eder, M. (2020). Topic modeling, long texts and the best number of topics. some problems and solutions. *Quality & Quantity*, pages 1–14.

Schapire, R. E. (2013). Explaining adaboost. In *Empirical inference*, pages 37–52. Springer.

Schapire, R. E. and Freund, Y. (2013). Boosting: Foundations and algorithms. *Kybernetes*.

Schapire, R. E. and Singer, Y. (2000). Boostexter: A boosting-based system for text categorization. *Machine learning*, 39(2):135–168.

Schlee, D. (1973). *Numerical Taxonomy. The Principles and Practice of Numerical Classification*. Freeman, San Francisco.

Schmidt, C. O. and Kohlmann, T. (2008). When to use the odds ratio or the relative risk? *International journal of public health*, 53(3):165.

Schütze, H., Manning, C. D., and Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.

Sciandra, M., D'Ambrosio, A., and Plaia, A. (2020). Projection clustering unfolding: A new algorithm for clustering individuals or items in a preference matrix. *Data Analysis and Applications 3: Computational, Classification, Financial, Statistical and Stochastic Methods*, 5:215–230.

Sciandra, M., Plaia, A., and Capursi, V. (2017). Classification trees for multivariate ordinal response: an application to student evaluation teaching. *Quality & Quantity*, 51(2):641–655.

Smith, J. H. (1973). Aggregation of preferences with variable electorate. *Econometrica*, 41(6):1027–1041.

Sokal, R. R. and Rohlf, F. J. (1962). The comparison of dendrograms by objective methods. *Taxon*, 11(2):33–40.

Sokal, R. R., Sneath, P. H. A., et al. (1963). Principles of numerical taxonomy. *Principles of numerical taxonomy.*

Solomatine, D. P. and Shrestha, D. L. (2004). Adaboost. rt: a boosting algorithm for regression problems. In *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)*, volume 2, pages 1163–1168. IEEE.

Spearman, C. (1987). The proof and measurement of association between two things. *The American Journal of Psychology*, 100(3/4):441–471.

Srivastava, A. and Sutton, C. (2017). Autoencoding variational inference for topic models. *arXiv preprint arXiv:1703.01488*.

Taherdoost, H. (2019). What is the best response scale for survey and questionnaire design; review of different lengths of rating scale/attitude scale/likert scale. *Hamed Taherdoost*, pages 1–10.

Therneau, T., Atkinson, B., Ripley, B., and Ripley, M. B. (2015). Package 'rpart'. *Available online: cran. ma. ic. ac. uk/web/packages/rpart/rpart. pdf (accessed on 20 April 2016)*.

Thompson, G. (1993). Generalized permutation polytopes and exploratory graphical methods for ranked data. *The Annals of Statistics*, pages 1401–1430.

Tumminello, M., Micciche, S., Lillo, F., Piilo, J., and Mantegna, R. N. (2011). Statistically validated networks in bipartite complex systems. *PloS one*, 6(3):e17994.

Ullmann, T., Hennig, C., and Boulesteix, A.-L. (2022). Validation of cluster analysis results on validation data: A systematic framework. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, page e1444.

Von Luxburg, U. (2010). Clustering stability: an overview. *Foundations and Trends in Machine Learning*, 2(3):235–274.

Waldherr, A., Heyer, G., Jähnichen, P., Niekler, A., and Wiedemann, G. (2015). Mining big data with computational methods. In *Political Communication in the Online World*, pages 201–217. Routledge.

Wallach, H. M., Murray, I., Salakhutdinov, R., and Mimno, D. (2009). Evaluation methods for topic models. In *Proceedings of the 26th annual international conference on machine learning*, pages 1105–1112.

Wang, L., Wei, B., and Yuan, J. (2011). Topic discovery based on lda_col model and topic significance re-ranking. *JCP*, 6(8):1639–1647.

Werbin-Ofir, H., Dery, L., and Shmueli, E. (2019). Beyond majority: Label ranking ensembles based on voting rules. *Expert Systems with Applications*, 136:50–61.

Wu, Q., Burges, C. J., Svore, K. M., and Gao, J. (2010). Adapting boosting for information retrieval measures. *Information Retrieval*, 13(3):254–270.

Wu, W., Xiong, H., and Shekhar, S. (2003). *Clustering and information retrieval*, volume 11. Springer Science & Business Media.

Xing, L., Paul, M. J., and Carenini, G. (2019). Evaluating topic quality with posterior variability. *arXiv preprint arXiv:1909.03524*.

Xu, J. and Li, H. (2007). Adarank: a boosting algorithm for information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 391–398.

Young, H. P. and Levenglick, A. (1978). A consistent extension of condorcet's election principle. *SIAM Journal on applied Mathematics*, 35(2):285–300.

Zadegan, S. M. R., Mirzaie, M., and Sadoughi, F. (2013). Ranked k-medoids: A fast and accurate rank-based partitioning algorithm for clustering large datasets. *Knowledge-Based Systems*, 39:133–143.

Zhou, T., Kuscsik, Z., Liu, J.-G., Medo, M., Wakeling, J. R., and Zhang, Y.-C. (2010). Solving the apparent diversity-accuracy dilemma of recommender systems. *Proceedings of the National Academy of Sciences*, 107(10):4511–4515.

Zhou, Y., Liu, Y., Yang, J., He, X., and Liu, L. (2014). A taxonomy of label ranking algorithms. *JCP*, 9(3):557–565.

Zhou, Y. and Qiu, G. (2018). Random forest for label ranking. *Expert Systems with Applications*, 112:99–109.

Zuo, Y., Zhao, J., and Xu, K. (2016). Word network topic model: a simple but general solution for short and imbalanced texts. *Knowledge and Information Systems*, 48(2):379–398.

Appendix A

Additional material Chapter 2

A.1 A comparison of the weighted and unweighted QuickCons algorithms' computation times

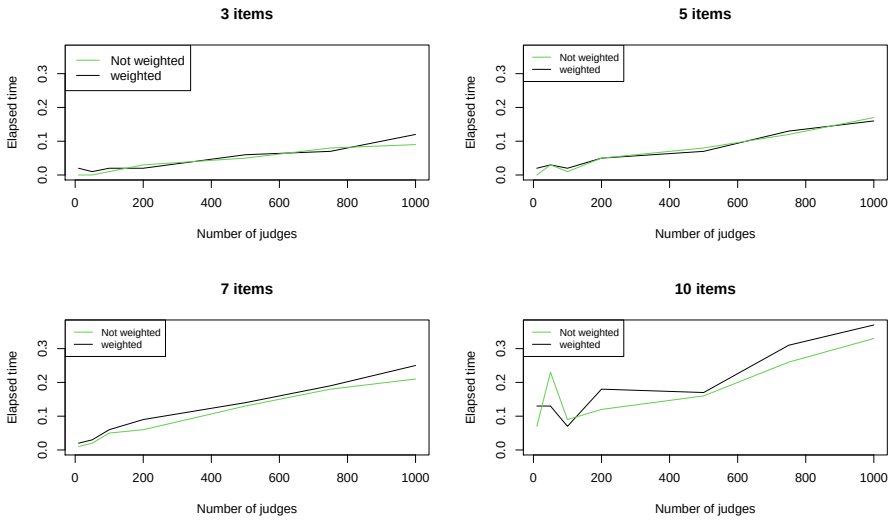


Figure A.1: Computation times comparison: item-weighted QuickCons vs unweighted QuickCons.

Appendix B

Additional material Chapter 3

B.1 Variable importance in the boosting procedure, datasets: German2005, German2009 and Top7Movies

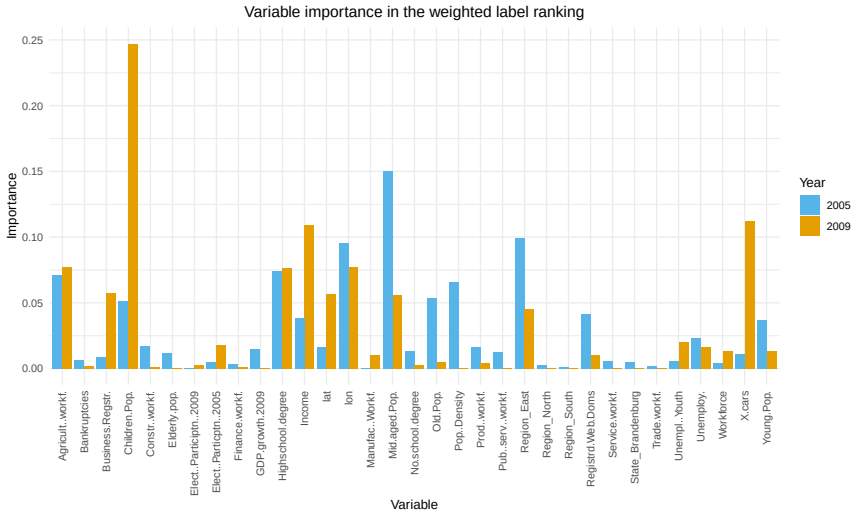


Figure B.1: Variable Importance at final step for 2005-2009 German Elections dataset

Variable	Importance
Longitude	0.53
Latitude	0.46
Age	0.01

Table B.1: Variable Importance at final step for Top7Movies dataset

Appendix C

Proofs of formulas in Chapter 4

C.1 Proof of Proposition 1

1. Positivity holds since $h(p_{ij}, a_{ij}) \geq 0$ for all $i, j \in \{1, \ldots, m\}$.

2. Symmetry holds since $p_{ij} = p_{ji}$ (see Eq. (5.5)) and $a_{ij} = a_{ji}$ (see Eq. (5.6)) for all $i, j \in \{1, \ldots, m\}$.

3. Identity of indiscernibles: Obviously, $D_\lambda^r\Big((\pi_1, A_1), (\pi_1, A_1)\Big) = 0$. If $D_\lambda^r\Big((\pi_1, A_1), (\pi_2, A_2)\Big) = 0$, then $(\lambda \cdot p_{ij}^r + (1-\lambda) \cdot a_{ij}^r)^{\frac{1}{r}} = 0$ for all $i, j \in \{1, \ldots, m\}$. Since $p_{ij}, a_{ij} \geq 0$ and $\lambda \in (0,1)$, we have $p_{ij} = a_{ij} = 0$ for all $i, j \in \{1, \ldots, m\}$. Then, $O_{\pi_1}(y_i, y_j) = O_{\pi_2}(y_i, y_j)$, $I_{A_1}(y_i) = I_{A_1}(y_i)$ and $I_{A_1}(y_j) = I_{A_2}(y_j)$ for all $i, j \in \{1, \ldots, m\}$. Consequently, $(\pi_1, A_1) = (\pi_2, A_2)$.

4. Triangle inequality: If we define

$$p'_{ij} = \frac{1}{2} \cdot |O_{\pi_1}(y_i, y_j) - O_{\pi_2}(y_i, y_j)|,$$

$$p''_{ij} = \frac{1}{2} \cdot |O_{\pi_2}(y_i, y_j) - O_{\pi_3}(y_i, y_j)|,$$

$$p'''_{ij} = \frac{1}{2} \cdot |O_{\pi_1}(y_i, y_j) - O_{\pi_3}(y_i, y_j)|,$$

then, we have

$$p'''_{ij} \leq p'_{ij} + p''_{ij}. \tag{C.1}$$

Similarly, if we define

$$a'_{ij} = \frac{1}{2} \cdot \Big(|I_{A_1}(y_i) - I_{A_2}(y_i)| + |I_{A_1}(y_j) - I_{A_2}(y_j)| \Big),$$
$$a''_{ij} = \frac{1}{2} \cdot \Big(|I_{A_2}(y_i) - I_{A_3}(y_i)| + |I_{A_2}(y_j) - I_{A_3}(y_j)| \Big),$$
$$a'''_{ij} = \frac{1}{2} \cdot \Big(|I_{A_1}(y_i) - I_{A_3}(y_i)| + |I_{A_1}(y_j) - I_{A_3}(y_j)| \Big),$$

then, we have

$$a'''_{ij} \leq a'_{ij} + a''_{ij}. \tag{C.2}$$

From Eqs. (C.1) and (C.2) it follows

$$p'''_{ij} + a'''_{ij} \leq p'_{ij} + p''_{ij} + a'_{ij} + a''_{ij}. \tag{C.3}$$

To prove the triangle inequality we need to show

$$h(p'''_{ij}, a'''_{ij}) \leq h(p'_{ij}, a'_{ij}) + h(p''_{ij}, a''_{ij}),$$

i.e.,

$$\Big(\lambda \cdot (p'''_{ij})^r + (1-\lambda) \cdot (a'''_{ij})^r \Big)^{\frac{1}{r}} \leq$$
$$\Big(\lambda \cdot (p'_{ij})^r + (1-\lambda) \cdot (a'_{ij})^r \Big)^{\frac{1}{r}} + \Big(\lambda \cdot (p''_{ij})^r + (1-\lambda) \cdot (a''_{ij})^r \Big)^{\frac{1}{r}} \tag{C.4}$$

Raising the two members of the inequality by r, Eq. (C.4) is equivalent to

$$\lambda \cdot (p'''_{ij})^r + (1-\lambda) \cdot (a'''_{ij})^r \leq \tag{C.5}$$
$$\left(\Big(\lambda \cdot (p'_{ij})^r + (1-\lambda) \cdot (a'_{ij})^r \Big)^{\frac{1}{r}} + \Big(\lambda \cdot (p''_{ij})^r + (1-\lambda) \cdot (a''_{ij})^r \Big)^{\frac{1}{r}} \right)^r.$$

Taking into account that for all $a, b \geq 0$ and $r \geq 1$, (see Hardy et al. (1952, p. 32) for more details) it holds:

$$(a+b)^r \geq a^r + b^r,$$

we have

$$\left(\left(\lambda\cdot(p'_{ij})^r+(1-\lambda)\cdot(a'_{ij})^r\right)^{\frac{1}{r}}+\left(\lambda\cdot(p''_{ij})^r+(1-\lambda)\cdot(a''_{ij})^r\right)^{\frac{1}{r}}\right)^r\geq$$
$$\left(\lambda\cdot(p'_{ij})^r+(1-\lambda)\cdot(a'_{ij})^r\right)+\left(\lambda\cdot(p''_{ij})^r+(1-\lambda)\cdot(a''_{ij})^r\right)=$$
$$\lambda\cdot\left((p'_{ij})^r+(p''_{ij})^r\right)+(1-\lambda)\cdot\left((a'_{ij})^r+(a''_{ij})^r\right). \tag{C.6}$$

Because of Eqs. (C.1) and (C.2), we have

$$\lambda\cdot\left((p'_{ij})^r+(p''_{ij})^r\right)+(1-\lambda)\cdot\left((a'_{ij})^r+(a''_{ij})^r\right)\geq$$
$$\lambda\cdot(p'''_{ij})^r+(1-\lambda)\cdot(a'''_{ij})^r. \tag{C.7}$$

Therefore, following Eqs. (C.6) and (C.7), we can write:

$$\lambda\cdot(p'''_{ij})^r+(1-\lambda)\cdot(a'''_{ij})^r\leq$$
$$\lambda\cdot\left((p'_{ij})^r+(p''_{ij})^r\right)+(1-\lambda)\cdot\left((a'_{ij})^r+(a''_{ij})^r\right)\leq$$
$$\left(\left(\lambda\cdot(p'_{ij})^r+(1-\lambda)\cdot(a'_{ij})^r\right)^{\frac{1}{r}}+\left(\lambda\cdot(p''_{ij})^r+(1-\lambda)\cdot(a''_{ij})^r\right)^{\frac{1}{r}}\right)^r.$$

for all $i,j\in\{1,\ldots,n\}$.

Hence,

$$D^r_\lambda\Big((\pi_1,A_1),(\pi_3,A_3)\Big)\leq D^r_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big)+D^r_\lambda\Big((\pi_2,A_2),(\pi_3,A_3)\Big)$$

for all $(\pi_1,A_1),(\pi_2,A_2),(\pi_3,A_3)\in\mathscr{R}(X)$.

C.2 Proof of Proposition 2

The first distance can be expressed in the following way:

$$D^1_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = \frac{2}{n\cdot(n-1)}\cdot\sum_{\substack{i,j=1\\ i<j}}^{n} h(p_{ij},a_{ij}) =$$

$$\frac{2}{n\cdot(n-1)}\cdot\sum_{\substack{i,j=1\\ i<j}}^{n}\Big(\lambda\cdot p^1_{ij}+(1-\lambda)\cdot a^1_{ij}\Big) =$$

$$\frac{2\cdot\lambda}{n\cdot(n-1)}\cdot\sum_{\substack{i,j=1\\ i<j}}^{n}\frac{|O_{\pi_1}(y_i,y_j)-O_{\pi_2}(y_i,y_j)|}{2}+$$

$$\frac{2\cdot(1-\lambda)}{n\cdot(n-1)}\cdot\sum_{\substack{i,j=1\\ i<j}}^{n}\frac{|I_{A_1}(y_i)-I_{A_2}(y_i)|+|I_{A_1}(y_j)-I_{A_2}(y_j)|}{2}.$$

Taking into account Eq. (5.12), the equality between $D^1_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big)$ and $d_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big)$ holds if and only if

$$\frac{1-\lambda}{n\cdot(n-1)}\cdot\sum_{\substack{i,j=1\\ i<j}}^{n}|I_{A_1}(y_i)-I_{A_2}(y_i)|+|I_{A_1}(y_j)-I_{A_2}(y_j)| =$$

$$\frac{1-\lambda}{n}\cdot\sum_{i=1}^{n}|I_{A_1}(y_i)-I_{A_2}(y_i)|. \tag{C.8}$$

Let us define $I_i = |I_{A_1}(y_i)-I_{A_2}(y_i)|$. Then,

- $\displaystyle\sum_{\substack{i,j=1\\ i<j}}^{n}\Big(|I_{A_1}(y_i)-I_{A_2}(y_i)|+|I_{A_1}(y_j)-I_{A_2}(y_j)|\Big) = \sum_{\substack{i,j=1\\ i<j}}^{n}(I_i+I_j),$
- $\displaystyle\sum_{i=1}^{n}|I_{A_1}(y_i)-I_{A_2}(y_i)| = \sum_{i=1}^{n}I_i.$

Therefore, the equality in Eq. (C.8) can be re-written as:

$$D^1_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) = d_\lambda\Big((\pi_1,A_1),(\pi_2,A_2)\Big) \quad \Leftrightarrow$$

$$\frac{1-\lambda}{n\cdot(n-1)}\cdot\sum_{\substack{i,j=1\\ i<j}}^{n}(I_i+I_j) = \frac{1-\lambda}{n}\cdot\sum_{i=1}^{n}I_i. \tag{C.9}$$

To prove Eq. (C.9):

$$\frac{1-\lambda}{n\cdot(n-1)}\cdot\sum_{\substack{i,j=1\\ i<j}}^{n}(I_i+I_j)=$$

$$\frac{1-\lambda}{n\cdot(n-1)}\cdot(I_1+I_2+\ I_1+I_3+\cdots+I_2+I_3+\cdots+I_{n-1}+I_n)=$$

$$\frac{1-\lambda}{n\cdot(n-1)}(n-1)\cdot(I_1+I_2+\cdots+I_n)=\frac{1-\lambda}{n}\cdot\sum_{i=1}^{n}I_i.$$

The equality Eq. (C.9) is a necessary and sufficient condition to show that $D^1_\lambda = d_\lambda$, for every $\lambda\in[0,1]$.

Appendix D

Additional material Chapter 6

Table D.1: Spearman rank correlation coefficient and Pearson correlation coefficient with human judgments for metrics without noise

	Correlation coefficient without noise	
Method	Spearman	Pearson
J	0.81	0.67
Dc	0.81	0.68
SS	0.81	0.67
FM	0.86	**0.78**
D_ρ	**0.87**	0.77
$\tilde{R}$	0.79	0.66
$\tilde{p}_v$	0.86	0.77
PMI Newman et al. (2009)	0.80	0.84
$UMass$ Mimno et al. (2011)	0.75	0.81
NPMI Lau et al. (2014)	**0.88**	**0.87**
CV Röder et al. (2015)	0.77	0.76
$tf\text{-}idf$ Nikolenko et al. (2017)	0.81	0.85

Table D.2: Coherence scores

	Coh_{SVN}						state-of-the-art						
Topic	J	Dc	SS	FM	D_p	$\bar{R}$	$\tilde{p}_v$	PMI Newman et al. (2009)	$UMass$ Mimno et al. (2011)	NPMI Lau et al. (2014)	CV Röder et al. (2015)	$tf\text{-}idf$ Nikolenko et al. (2017)	HumanJ
z_1	0.006	0.012	0.003	0.137	0.076	0.037	0.133	-2.619	-5.988	-0.119	0.326	-296.42	2.332
z_2	0.004	0.008	0.002	0.089	0.049	0.022	0.084	-5.979	-9.360	-0.272	0.309	-492.41	1.391
z_3	0.010	0.018	0.005	0.291	0.159	0.060	0.265	0.926	-1.965	0.084	0.663	-87.84	3.743
z_4	0.007	0.014	0.004	0.156	0.086	0.042	0.144	-6.258	-9.391	-0.208	0.392	-498.77	1.599
z_5	0.006	0.011	0.003	0.140	0.078	0.037	0.131	-3.533	-6.818	-0.148	0.321	-344.72	2.416
z_6	0.070	0.128	0.037	0.937	0.545	0.422	0.966	1.632	-0.900	0.257	0.899	-5.06	3.847
z_7	0.021	0.039	0.011	0.345	0.194	0.118	0.317	0.864	-1.365	0.127	0.627	-62.09	2.688
z_8	0.009	0.017	0.005	0.228	0.125	0.058	0.219	0.846	-1.826	0.004	0.562	-98.04	2.351
z_9	0.024	0.043	0.013	0.356	0.206	0.148	0.354	-1.721	-5.016	-0.067	0.293	-247.09	3.178
z_{10}	0.018	0.033	0.009	0.277	0.159	0.127	0.281	-1.674	-4.393	-0.102	0.303	-237.55	2.416
z_{11}	0.019	0.037	0.010	0.397	0.223	0.140	0.394	0.977	-1.738	0.063	0.622	-93.95	3.381
z_{12}	0.017	0.032	0.009	0.359	0.201	0.094	0.348	0.804	-1.437	0.046	0.587	-80.95	2.851
z_{13}	0.061	0.111	0.033	0.886	0.506	0.341	0.848	2.437	-1.716	0.365	0.911	-10.61	3.431
z_{14}	0.007	0.013	0.004	0.178	0.098	0.051	0.178	-0.579	-3.907	-0.013	0.484	-190.96	2.233
z_{15}	0.015	0.028	0.007	0.425	0.234	0.085	0.408	0.586	-1.306	0.093	0.590	-79.56	3.356
z_{16}	0.041	0.078	0.021	0.819	0.460	0.271	0.815	1.351	-1.030	0.229	0.845	-34.27	3.406
z_{17}	0.155	0.014	0.505	0.500	0.247	0.048	0.026	-2.040	-4.797	0.007	0.384	-258.17	2.901
z_{18}	0.047	0.002	0.081	0.080	0.008	0.009	0.005	-2.305	-4.582	-0.117	0.234	-277.77	2.084
z_{19}	0.028	0.051	0.015	0.623	0.343	0.152	0.603	1.654	-0.990	0.230	0.867	-32.36	3.535
z_{20}	0.006	0.012	0.003	0.103	0.059	0.050	0.105	-7.400	-11.466	-0.304	0.378	-575.41	1.579
z_{21}	0.023	0.042	0.012	0.419	0.235	0.142	0.396	-0.762	-4.153	0.089	0.579	-191.81	3.460
z_{22}	0.014	0.026	0.007	0.292	0.163	0.085	0.261	-3.339	-6.392	-0.090	0.328	-323.97	2.465
z_{23}	0.023	0.043	0.012	0.445	0.247	0.130	0.418	0.818	-1.308	0.125	0.661	-75.73	3.644
z_{24}	0.012	0.022	0.006	0.224	0.126	0.048	0.184	0.298	-1.236	0.023	0.413	-97.63	2.856
z_{25}	0.028	0.053	0.014	0.600	0.333	0.165	0.564	1.183	-1.350	0.169	0.781	-47.16	3.396
z_{26}	0.021	0.039	0.011	0.356	0.201	0.111	0.351	0.515	-1.683	0.061	0.544	-97.11	2.772
z_{27}	0.007	0.012	0.003	0.132	0.075	0.043	0.135	-7.618	-12.673	-0.285	0.379	-589.28	1.837
z_{28}	0.026	0.047	0.014	0.467	0.264	0.139	0.447	1.287	-1.086	0.155	0.740	-58.45	3.223
z_{29}	0.018	0.034	0.010	0.396	0.219	0.104	0.370	0.825	-1.457	0.127	0.674	-72.54	3.307
z_{30}	0.005	0.010	0.003	0.181	0.100	0.048	0.150	-3.518	-7.015	-0.108	0.312	-346.57	1.837

Table D.3: Ranking coherence scores

	Coh_{SVN}							state-of-the-art					
Topic	J	Dc	SS	FM	D_ρ	$\tilde{R}$	$\tilde{p}_v$	PMI Newman et al. (2009)	$UMass$ Mimno et al. (2011)	NPMI Lau et al. (2014)	CV Röder et al. (2015)	tf-idf Nikolenko et al. (2017)	HumanJ
z_1	25	25	25	26	26	28	26	23	23	25	24	23	23
z_2	30	30	30	29	29	30	29	27	27	28	27	27	30
z_3	20	20	20	18	18	19	18	8	16	12	8	12	2
z_4	22	22	22	24	24	27	24	28	28	27	19	28	28
z_5	27	27	27	25	25	29	27	26	25	26	25	25	20
z_6	1	1	1	1	1	1	1	3	1	2	2	1	1
z_7	11	12	11	16	16	13	16	9	9	8	10	7	18
z_8	21	21	21	20	21	20	20	10	15	18	15	16	22
z_9	8	8	8	15	13	7	13	20	22	20	29	20	13
z_{10}	15	15	15	19	19	12	17	19	19	22	28	19	20
z_{11}	13	13	13	11	11	9	11	7	14	13	11	13	9
z_{12}	16	16	16	13	15	16	15	13	10	15	13	11	16
z_{13}	2	2	2	2	2	2	2	1	13	1	1	2	6
z_{14}	23	23	23	23	23	21	22	17	17	19	17	17	24
z_{15}	17	17	17	9	10	17	9	14	6	10	12	10	10
z_{16}	3	3	3	3	3	3	3	4	3	4	4	4	7
z_{17}	7	6	7	6	6	5	6	21	21	17	20	21	14
z_{18}	29	29	29	30	30	25	30	22	20	24	30	22	25
z_{19}	4	5	4	4	4	6	4	2	2	3	3	3	4
z_{20}	26	26	26	28	28	22	28	29	29	30	22	29	29
z_{21}	10	10	9	10	9	8	10	18	18	11	14	18	5
z_{22}	18	18	18	17	17	18	19	24	24	21	23	24	19
z_{23}	9	9	10	8	8	11	8	12	7	9	9	9	3
z_{24}	19	19	19	21	20	24	21	16	5	16	18	15	15
z_{25}	5	4	5	5	5	4	5	6	8	5	5	5	8
z_{26}	12	11	12	14	14	14	14	15	12	14	16	14	17
z_{27}	24	24	24	27	27	26	25	30	30	29	21	30	26
z_{28}	6	7	6	7	7	10	7	5	4	6	6	6	12
z_{29}	14	14	14	12	12	15	12	11	11	7	7	8	11
z_{30}	28	28	28	22	22	23	23	25	26	23	26	26	26