Société Francophone de Classification (SFC) Actes des 26èmes Rencontres

Miguel Couceiro, Amedeo Napoli

Société Francophone de Classification (SFC)

Actes des 26èmes Rencontres

3–5 septembre 2019

Inria Nancy Grand Est – LORIA Nancy

`https://project.inria.fr/sfc2019/`

Responsables de publication

Miguel Couceiro (Université de Lorraine, CNRS, Inria, LORIA, Nancy)

Amedeo Napoli (Université de Lorraine, CNRS, Inria, LORIA, Nancy)

Avant-Propos

La Société Francophone de Classification (SFC, `http://www.sfc-classification.net/`) organise chaque année les Rencontres de la Société Francophone de Classification. La classification sous toutes ces formes, mathématiques, informatique (apprentissage, fouille de données et découverte de connaissances ...), et statistiques, est la thématique étudiée par les membres de la société. Mais autant les variations autour du thème de la classification sont vastes autant les intérêts des chercheurs de la SFC et ses sympathisants sont variés et étendus. C'est cette idée d'illustrer les différentes facettes de la classification que nous allons favoriser et mettre en oeuvre lors des rencontres qui ont lieu à Nancy, au Centre de Recherche Inria Nancy Grand Est / LORIA entre les 3 et 5 septembre 2019.

Les rencontres de la SFC permettent aux membres de la société de se rencontrer, comme il se doit, de présenter des résultats récents et des applications originales en classification ou dans des domaines connexes. Elles permettent encore de favoriser les échanges scientifiques à l'intérieur de la société et de faire connaître à l'extérieur les travaux de ses membres. Cette année, nous allons insister sur le côté polymorphe de la classification et faire se croiser des chercheurs de domaines proches ou assez proches, en particulier dans le domaine de l'informatique, des mathématiques et des statistiques, dont les intérêts sont liés à la classification. Il faut encore noter les nombreux domaines d'applications, qui comprennent l'agronomie, la biologie, la médecine, la phylogénie, la recherche d'information, la surveillance, et le web. Enfin, la liste des présentations invitées et le tutoriel de l'édition 2019 des rencontres de la SFC témoignent encore de la mise en valeur du côté hybride de la classification.

Cette année, le Centre de Recherche Inria Nancy Grand Est / LORIA se charge de l'organisation des rencontres des 26èmes Rencontres de la Société Francophone de Classification entre les 3 et 5 septembre 2019. Pour finir et pour la petite histoire, c'est la seconde fois que les rencontres sont organisées à Nancy, et cela 20 ans après les rencontres de la SFC en 1999, à l'orée du 21ème siècle.

Les présidents du comité d'organisation et du comité de programme SFC 2019 :

Miguel Couceiro
Université de Lorraine, CNRS, Inria Nancy Grand Est, LORIA Nancy

Amedeo Napoli
Université de Lorraine, CNRS, Inria Nancy Grand Est, LORIA Nancy

Comité de programme

Présidents du comité de programme :
Miguel Couceiro
Université de Lorraine, CNRS, Inria Nancy Grand Est, LORIA Nancy
Amedeo Napoli
Université de Lorraine, CNRS, Inria Nancy Grand Est, LORIA Nancy

Khalid Benabdeslem, LIRIS, Université Claude Bernard, Lyon
Patrice Bertrand, Université Paris Dauphine
Paula Brito, Université de Porto, Portugal
François Brucker, Ecole Centrale Marseille
Véronique Cariou, ONIRIS Nantes
Guillaume Cleuziou, Université d'Orléans
Francisco De-Carvalho, Université Pernambuco, Brésil
Jean Diatta, Université de La Réunion
Bernard Fichet, Université Aix-Marseille
Nadia Ghazalli, Université du Québec à Trois-Rivières
Yann Guermeur, Université de Lorraine, LORIA
Marianne Huchard, Université de Montpellier, LIRMM
Mehdi Kaytoue, Université de Lyon, INSA, LIRIS
Pascale Kuntz, Université de Nantes
Labiod Lazhar, Université Paris Descartes
Mustapha Lebbah, Université Paris 13
Vincent Lemaire, Orange Lannion
Vladimir Makarenkov, Université UQAM, Canada
Ahmed Moussa, ENSA Tanger, Maroc
Mohamed Nadif, Université Paris Descartes Paris
Ndèye Niang-Keita, CNAM Paris
Marc Plantevit, Université Claude Bernard LIRIS, Lyon
Philippe Preux, Université de Charles de Gaulle
Fabrice Rossi, Université Panthéon-Sorbonne
Aghiles Salah, Singapore Management University
Arnaud Soulet, Université François Rabelais Tours
Julien Velcin, Université Lyon 2

Comité d'organisation

Anne-Lise Charbonnier, Inria Nancy Grand Est
Miguel Couceiro, Université de Lorraine, CNRS, Inria, LORIA, Nancy
Pascal Cuxac, INIST Nancy
Anne Gégout-Petit, Université de Lorraine, CNRS, Inria, IECL, Nancy
Amedeo Napoli, Université de Lorraine, CNRS, Inria, LORIA, Nancy
Jeremie Nevin, Inria Nancy Grand Est

Table des matières

Invited Talk
Monotonicity, a deep property in data science

Bernard De Baets

Department of Data Analysis and Mathematical Modelling
Ghent University
Coupure links 653, Ghent 9000, Belgium
Bernard.DeBaets@UGent.be

Abstract. In many modelling problems, there exists a monotone relationship between one or more of the input variables and the output variable, although this may not always be fully the case in the observed input-output data due to data imperfections. Monotonicity is also a common property of evaluation and selection procedures. In contrast to a local property such as continuity, monotonicity is of a global nature and any violation of it is therefore simply unacceptable. We explore several problem settings where monotonicity matters, including fuzzy modelling, machine learning and decision making. Central to the above three settings is the cumulative approach, which matches nicely with the monotonicity requirement.

By far the most popular fuzzy modelling paradigm, despite its weak theoretical foundations, is the rule-based approach of Mamdani and Assilian. In numerous applied papers, authors innocently assume that given a fuzzy rule base that appears monotone at the linguistic level, this will be the case for the generated input-output mapping as well. Unfortunately, this assumption is false, and we will show how to counter it. Moreover, we will show that an implication-based interpretation, accompanied with a cumulative approach based on at-least and/or at-most quantifiers, might be a much more reasonable alternative. Next, we deal with a particular type of classification problem, in which there exists a linear ordering on the label set (as in ordinal regression) as well as on the domain of each of the features. Moreover, there exists a monotone relationship between the features and the class labels. Such problems of monotone classification typically arise in a multi-criteria evaluation setting. When learning such a model from a data set, we are confronted with data impurity in the form of reversed preference. We present the Ordinal Stochastic Dominance Learner framework, which permits to build various instance-based algorithms able to process such data.

Finally, we explore a pairwise preference setting where each stakeholder expresses his/her preferences in the shape of a reciprocal relation that is monotone w.r.t. a linear order on the set of alternatives. The goal is to come up with an overall monotone reciprocal relation reflecting 'best' the opinions. We formulate the problem as an optimization problem, where the aggregated linear order is that for which the implied stochastic monotonicity conditions are closest to being satisfied by the distribution of the input monotone reciprocal relations. Interesting links with social choice will be pointed out.

Invited Talk
Discovering habits with periodic patterns

Alexandre Termier

Univ Rennes, Inria, CNRS, IRISA
35000 Rennes, France
Alexandre.Termier@irisa.fr

Abstract. In various fields, traces of timestamped events are captured, precisely describing the operation of a system or a process. It can as well be a web server log, the operation log of a manufacture, or even a personal activity journal. It can be interesting to analyze such data to discover periodic patterns, i.e. sets or sequences of events that occur within a regular delay. Two problems arise: first, precisely defining such patterns is non-trivial, as many straightforward definitions break in practice due to noise. Second, the search space of periodic patterns is huge, requiring some effort to make the enumeration efficient, and to output only the most interesting patterns. This talk will present the general problem of mining periodic patterns, our work to represent periodic patterns as formal concepts, and our latest work to use MDL approaches to select few relevant periodic patterns.

Présentation invitée
Structure de Treillis : panorama des aspects structurels et algorithmiques

Karell Bertet

Laboratoire L3i
Université de La Rochelle
Avenue Michel Crépeau, 17042 La Rochelle, France
Karell.Bertet@univ-lr.fr

Résumé Le premier ouvrage de référence de la théorie des treillis est la première édition du livre de Birkhoff en 1940. Cependant, la notion de treillis a été introduite dès la fin du 19ème siècle comme une structure algébrique munie de deux opérateurs appelés borne inférieure et borne supérieure. Depuis les années 2000, l'émergence de l'analyse formelle des concepts (FCA) dans divers domaines de l'informatique, que ce soit en analyse de données et classification, en représentation des connaissances ou en recherche d'information, a mis en avant les structures de treillis des concepts et de bases de règles d'implication. Plus récement, les structures de patrons permettent d'étendre FCA à des données non binaires.

Une manipulation efficace de ces structures passe par une bonne connaissance du formalisme, des propriétés structurelles et des principaux résultats de la théorie des treillis et de l'AFC. Nous présenterons un panorama des concepts de base de la théorie des treillis, de FCA et des structures de patrons, ainsi que les principaux algorithmes de génération des objets qui la composent.

Présentation invitée
Apprentissage et classification par méthodes collaboratives : comment choisir ses collaborateurs et qu'échanger avec eux ?

Antoine Cornuéjols

UMR MIA-Paris, AgroParisTech,
INRA, Université Paris-Saclay,
Paris, France
antoine.cornuejols@agroparistech.fr

Résumé Des problèmes d'intelligence artificielle aussi divers que les jeux à plusieurs joueurs, l'apprentissage supervisé par des méthodes d'ensemble ou le clustering collaboratif peuvent être considérées avec les mêmes questions fondamentales : (1) comment choisir ses collaborateurs c'est-à-dire les sources d'information ? (2) quelles informations il est intéressant d'échanger ? et (3) comment combiner les informations reçues ?

Dans cet exposé nous examinerons successivement les algorithmes de jeux, un nouveau algorithme d'apprentissage supervisé par transfert puis le clustering collaboratif pour essayer d'en tirer des leçons sur les méthodes collaboratives pour la classification et, à tout le moins, de poser les bonnes questions.

Présentation invitée
Apprentissage de représentations avec des "connaissances faibles" : application au clustering et à la classification

Dino Ienco

Maison de la Télédétection en Languedoc-Roussillon
UMR TETIS, IRSTEA, Université de Montpellier, LIRMM
Montpellier, France dino.ienco@irstea.fr

Résumé Dans le contexte de la classification semi-supervisée, nous avons à notre disposition de gros volumes de données non étiquetées, mais très peu de données étiquetées et de connaissances associées aux données. Ce scénario n'est pas propice aux techniques d'apprentissage profond. Dans cette présentation je vais illustrer mes derniers travaux de recherche en apprentissage semi-supervisé en utilisant des techniques d'apprentissage profond, avec des applications au clustering ainsi qu'à la classification transductive.

Tutorial
On Ordered Sets in Pattern Mining

Aimene Belfodil

Université de Lyon, INSA Lyon, CNRS, LIRIS UMR 5205
F-69621, LYON, France
Aimene.Belfodil@insa-lyon.fr

Abstract. This tutorial discusses the general task of pattern mining and its related problems from an order-theoretic point of view. The general aim is to provide a unified view defining a pattern mining task as well as the underlying pattern space. To achieve this, we start by presenting a quite simple, yet general, framework, dubbed pattern setup. We instantiate subsequently the framework on various pattern mining problems on different pattern languages. Next, we explore different algorithms to explore pattern search spaces. Finally, we discuss some open problems.

Pattern mining is a general data mining task which aims to discover useful and actionable patterns in databases. Different subtasks are identified in this field: frequent pattern mining, (class) association rule mining, subgroup discovery, exceptional model mining, redescription mining, and high utility pattern mining among others. These different pattern mining tasks share in their core definition common settings as for instance the pattern language they need to explore: itemsets, intervals, convex polygons, neighborhood patterns, (complex) sequential patterns, trajectory patterns, periodic patterns, subgraph patterns, etc.

More generally, one should answer the following questions in order to instantiate a pattern mining task: (1) What is the initial representation of the provided database? That is: What are the objects (rows) and what are the descriptive attributes (columns)? (2) What is the considered pattern language used to describe objects in the database w.r.t. the pattern mining task and the provided information? (3) How to check whether a pattern holds for some object in the database? (4) How to evaluate the "interestingness" of the findings (i.e. a quality measure, constraints that patterns or the pattern set need to have, etc.)?

Présentation spéciale *Class ?* : un jeu pour sensibiliser aux classifications

Line van den Berg et Jérôme Euzenat

Université Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG
F-38000 Grenoble France
{Line.Van-den-berg,Jerome.Euzenat}@inria.fr

Résumé Nous avons développé un jeu pour sensibiliser le grand public à certains aspects des classifications. Son but principal est de faire toucher du doigt à des écoliers que les mêmes individus peuvent être classifiés de différentes manières et qu'il est possible de transmettre une classification sans l'expliciter. Le jeu lui-même utilise des cartes représentant les individus et l'un des joueurs dispose d'une classification cachée. Nous avons utilisé ce jeu avec des élèves du CM2 à la seconde, mais même pour les plus grands il a des aspects intéressants.

Site web : https://moex.inria.fr/mediation/class/

Factorisation matricielle non-négative sémantique

Mickael Febrissy*, Aghiles Salah**, Melissa Ailem***, Mohamed Nadif*

*LIPADE, Université de Paris, 75006 Paris, France
{prénom.nom}@parisdescartes.fr,
**Singapore Management University
asalah@smu.edu.sg
***University of Southern California, Los Angeles, CA
ailem@usc.edu

Résumé. La factorisation matricielle non-négative (Non-negative Matrix Factorization) ou NMF est devenue populaire par ses applications pertinentes dans la classification de documents. Cependant, sous sa forme originale, elle ne propose aucune caractéristique permettant de quantifier ou d'interpréter les dépendances contextuelles au sein des mots pouvant aider à mieux appréhender les relations entre les documents. Partant du principe que deux mots apparaissant souvent ensemble seraient sémantiquement liés, nous proposons une approche permettant de prendre en compte ces deux informations en factorisant conjointement un corpus de textes régit sous la forme d'une matrice documents-termes $\boldsymbol{X}$ par le produit de deux matrices $\boldsymbol{Z}\boldsymbol{W}^T$, ainsi qu'une matrice termes-contextes $\boldsymbol{M}$ par le produit de deux matrices $\boldsymbol{W}\boldsymbol{Q}^T$. Nous verrons que cette approche conduit à de meilleurs résultats pour des tâches de classification de documents.

1 Introduction

La factorisation matricielle non-négative (Non-negative matrix Factorization ou NMF) consiste à approximer une matrice non négative $\boldsymbol{X} \in \mathbb{R}^{n\times d}$ par le produit de deux matrices non négatives $\boldsymbol{Z} \in \mathbb{R}_+^{n\times g}$ dite matrice de coeficients à partir de laquelle on détermine une partition pour les observations, $\boldsymbol{W} \in \mathbb{R}_+^{d\times g}$ dite matrice des bases pouvant être assimilée à la matrice des centres de rang g. Bien que la NMF pourrait être utilisée pour l'analyse de données conventionnelles, son intérêt croissant est dû principalement à sa capacité de résoudre certains problèmes d'apprentissage automatique d'une manière élégante et facile à mettre en œuvre. Si avec la NMF, la réduction de la dimension est évidente, à travers cette factorisation, on peut facilement établir des liens avec la classification non supervisée et notamment avec le k-means et d'autres variantes. Malgré des performances très notables pour des tâches de classification de corpus de documents (matrices documents-termes), cette méthode arbore quelques limitations quant à sa capacité à capturer les relations sémantiques pouvant exister entre les termes du corpus, en partie les mots synonymes, les mots décrivant le même sujet qui au final apparaîtront dans divers documents. Pour remédier à ces limites, nous proposons ici d'étendre les travaux réalisés par Ailem et al. (2017).

2 SNMF (Semantic Non-negative Matrix Factorization)

Le modèle repose sur les approximations conjointes de deux matrices à savoir $\boldsymbol{X}$ par $\boldsymbol{ZW}^T$ représentant la matrice documents-termes, et $\boldsymbol{M}$ par $\boldsymbol{WQ}^T$ représentant la matrice décrivant des relations contextuelles entre les termes, avec un partage de la matrice $\boldsymbol{W}$ par les approximations respectives de chacune des matrices d'entrées où $\boldsymbol{W}$ joue le rôle d'une matrice de bases pour $\boldsymbol{X}$ et de coefficients pour $\boldsymbol{M}$. La fonction objective du modèle s'établit comme étant :

$$F(\boldsymbol{Z},\boldsymbol{W},\boldsymbol{Q}) = \underbrace{D_1(\boldsymbol{X},\boldsymbol{ZW}^\top)}_{\text{NMF}} + \lambda \underbrace{D_2(\boldsymbol{M},\boldsymbol{WQ}^\top)}_{\text{word embedding}} \tag{1}$$

où D_1 et D_2 sont respectivement des mesures d'erreur permettant de quantifier la qualité de l'approximation, $\boldsymbol{M} \in \mathbb{R}^{d^{'} \times d}$, $\boldsymbol{Q} \in \mathbb{R}^{d^{'} \times g}$ et λ un paramètre de régularisation. Dans le cadre de nos expériences, nous considérons le cas où $D_1 = D_2$ est la norme de Frobenius :

$$F(\boldsymbol{Z},\boldsymbol{W},\boldsymbol{Q}) = \frac{1}{2}||\boldsymbol{X} - \boldsymbol{ZW}^\top||_F^2 + \frac{\lambda}{2}||\boldsymbol{M} - \boldsymbol{WQ}^\top||_F^2 \tag{2}$$

et $\boldsymbol{M}$ étant la matrice de PPMI (Positive Point-wise Mutual Information) des cooccurrences des termes du corpus contenues dans la matrice $\boldsymbol{C} \in \mathbb{R}^{d \times d^{'}}$, voir par exemple (Role et Nadif, 2011). Un algorithme itératif similaire à celui employé par Lee et Seung (2001) basé sur des règles de mise à jour multiplicatives est utilisé pour l'optimisation de (2). L'algorithme dont la convergence est démontrée est présenté dans *Algorithm* 1.

Algorithm 1 NMF-Sémantique (`SNMF`).

Entrées : $\boldsymbol{X}$, $\boldsymbol{M}$, λ et g.
Sorties : $\boldsymbol{Z}$, $\boldsymbol{W}$ et $\boldsymbol{Q}$.
Étapes :
1. Initialisation : $\boldsymbol{Z} \leftarrow \boldsymbol{Z}^{(0)}$; $\boldsymbol{W} \leftarrow \boldsymbol{W}^{(0)}$ et $\boldsymbol{Q} \leftarrow \boldsymbol{Q}^{(0)}$;
répéter
  2. $\boldsymbol{Z} \leftarrow \boldsymbol{Z} \odot \frac{\boldsymbol{XW}}{\boldsymbol{ZW}^\top\boldsymbol{W}}$;
  3. $\boldsymbol{W} \leftarrow \boldsymbol{W} \odot \frac{(\boldsymbol{X}^\top\boldsymbol{Z}+\lambda\boldsymbol{MQ})}{\boldsymbol{W}(\boldsymbol{Z}^\top\boldsymbol{Z}+\lambda\boldsymbol{Q}^\top\boldsymbol{Q})}$;
  4. $\boldsymbol{Q} \leftarrow \boldsymbol{Q} \odot \frac{\boldsymbol{M}^\top\boldsymbol{W}}{\boldsymbol{QW}^\top\boldsymbol{W}}$;
jusqu'à convergence
5. Normaliser $\boldsymbol{Z}$ de sorte à obtenir des vecteurs unitaires.

$\odot$ désigne le produit d'Hadamard.

3 Expériences

Cinq jeux de données ont été utilisés afin de conduire les différentes analyses, à savoir **CSTR**, **CLASSIC4** [1], Reuteurs **RCV1** contenant les 4 plus grandes classes du corpus Reuters [2], 20-newsgroups **NG20**[3] et le jeu de données **NG5** reprenant uniquement 5 classes [3]

1. http ://www.dataminingresearch.com/
2. http ://www.cad.zju.edu.cn/home/dengcai/Data/TextData.html
3. rec.sport.baseball, soc.religion.christian, talk.politics.mideast, sci.electronics et sci.med

TAB. 1: Description des jeux de données, # indique la cardinalité

Jeux de données	Caractéristiques					
	#Documents	#Termes	#Clusters	nz_X (%)	Équilibre	nz_M (%)
CSTR	475	1000	4	3.40	0.399	13.80
CLASSIC4	7095	5896	4	0.59	0.323	6.21
RCV1	6387	16921	4	0.25	0.080	3.07
NG5	4905	10167	5	0.92	0.943	22.04
NG20	18846	14390	20	0.59	0.628	25.06

de NG20 ; sélectionnés suivant des critères représentatifs des différentes situations réelles et contraignantes rencontrées dans l'analyse et la classification de matrices documents-termes comme le nombre de clusters, la proportion des clusters, la sparsité, le degré de mélange et l'équilibre des proportions. Leurs caractéristiques seront décrites dans la table 1. Une pondération TF-IDF et une normalisation l_2 ont été appliquées sur chacun des jeux de données. Le paramètre g représentant le rang des matrices factorisées a été fixé comme étant le nombre de clusters respectif de chaque jeu de données et le paramètre de régularisation du terme de word-embbedings λ a été fixé à 1 après une étude de ses variations suivant les performances de classification de l'algorithme.

La table 2 détaille des différentes performances de l'algorithme (SNMF) en termes de NMI (Normalized Mutual Information) (Strehl et Ghosh, 2002) et de ARI (Adjusted Rand Index) (Rand, 1971) comparée à d'autres méthodes de NMF ayant été introduites pour l'amélioration de la NMF dans des tâches de classification. Les résultats montrent que SNMF fourni de bien meilleurs résultats en termes de NMI et ARI.

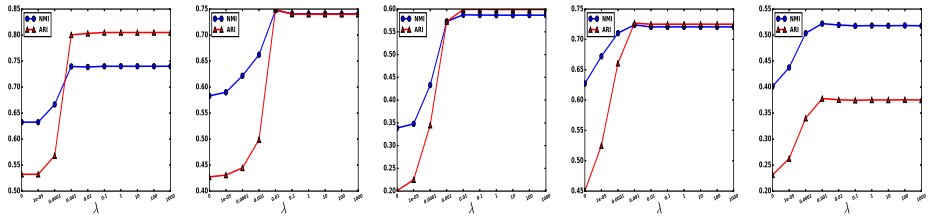
TAB. 2: Moyenne des résultats sur 30 essais.

Données	Critères	NMF	ONMF	PNMF	GNMF	SNMF
CSTR	NMI	0.65±0.01	0.65±0.05	0.66±0.01	0.57±0.08	**0.75**±0.01
	ARI	0.54±0.01	0.56±0.04	0.56±0.01	0.53±0.11	**0.80**±0.01
CLASSIC4	NMI	0.51±0.09	0.55±0.09	0.59±0.05	0.65±0.04	**0.72**±0.06
	ARI	0.36±0.10	0.39±0.09	0.44±0.01	0.49±0.05	**0.70**±0.09
RCV1	NMI	0.39±0.03	0.49±0.002	0.46±0.001	0.48±0.04	**0.56**±0.01
	ARI	0.29±0.02	0.39±0.004	0.37±0.001	0.39±0.03	**0.57**±0.01
NG5	NMI	0.65±0.05	0.65±0.04	0.65±0.05	0.63±0.07	**0.72**±0.04
	ARI	0.48±0.09	0.48±0.08	0.47±0.09	0.62±0.09	**0.70**±0.06
NG20	NMI	0.43±0.01	0.44±0.02	0.45±0.02	0.52±0.01	**0.53**±0.01
	ARI	0.24±0.01	0.22±0.02	0.24±0.02	0.35±0.05	**0.37**±0.01

La figure 1 illustre les variations de λ dans un intervalle entre 0 correspondant à la NMF originale et 1000. λ apparaît comme étant très stable et consistant suivant son augmentation. De manière générale, les premières améliorations apparaissent à partir de $\lambda = 10^{-3}$ et se stabilisent après 1. En d'autres termes l'apport de l'information contenue dans $\boldsymbol{M}$ semble régulièrement profiter à la classification des éléments de $\boldsymbol{X}$.

4 Conclusion

À travers ce modèle, nous avons présenté une nouvelle méthode de NMF permettant de prendre en compte les relations sémantiques et contextuelles existantes entre les différents



CSTR CLASSIC4 RCV1 NG5 NG20

FIG. 1: Impact du paramètre de régularisation λ.

termes d'un corpus résultant sur une amélioration des performances de la NMF pour la classification de documents. D'autre part, compte tenu des différents contextes pouvant être inférés à la matrice $\boldsymbol{M}$, ce modèle offre une grande flexibilité en fonction des différentes ressources pouvant être utilisées pour enrichir la classification des éléments du corpus de $\boldsymbol{X}$. Ainsi des bases de données telles que Wikipédia, ou Google pourraient être à l'initiative d'une représentation beaucoup plus vaste et générale des relations existantes, voire insoupçonnées au sein des termes d'une collection de documents.

Références

Ailem, M., A. Salah, et M. Nadif (2017). Non-negative matrix factorization meets word embedding. In *ACM SIGIR*, pp. 1081–1084.

Lee, D. D. et H. S. Seung (2001). Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pp. 556–562.

Rand, W. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association 66*(336), 846–850.

Role, F. et M. Nadif (2011). Handling the impact of low frequency events on co-occurrence based measures of word similarity-a case study of pointwise mutual information. In *KDIR*, pp. 226–231.

Strehl, A. et J. Ghosh (2002). Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research 3*(Dec), 583–617.

Summary

NMF approaches like other models do not explicitly account for the contextual dependencies between words. To remedy this limitation, we draw inspiration from neural word embedding and posit that words that frequently co-occur within the same context (e.g., sentence or document) are likely related to each other in some semantic aspect. We then propose to jointly factorize the document-word and word-word co-occurrence matrices. The decomposition of the latter matrix encourages frequently co-occurring words to have similar latent representations and thereby reflecting the relationships among them.

Hiérarchies, hiérarchies faibles et convexités d'intervalle

Patrice Bertrand*, Jean Diatta**

*Université Paris-Dauphine, PSL Research University, Ceremade, 75775 Paris Cedex 16, France
Patrice.Bertrand@ceremade.dauphine.fr
**LIM-EA2525, Université de La Réunion, Saint-Denis, France
jean.diatta@univ-reunion.fr

Résumé. Plusieurs approches ont montré que les hiérarchies, et leurs généralisations, peuvent être caractérisées comme des collections de parties convexes au sens de fonctions d'intervalle de types distincts. Nous proposons ici deux constructions de fonctions d'intervalle qui prennent en compte une dissimilarité arbitraire et un paramètre réglant la forme et la taille des intervalles. Les convexités définies au sens de ces fonctions d'intervalles, sont des classifications multi-niveaux qui sont emboîtées pour deux valeurs distinctes du paramètre. Il en résulte deux suites de classifications : chacune est croissante selon l'ordre d'inclusion et commence par la hiérarchie d'Asprejan, l'une se termine par la hiérarchie faible de Bandelt et Dress, et l'autre par la hiérarchie du lien simple. De ces suites de classifications, nous déduisons une évaluation du degré de robustesse de chaque classe de la hiérarchie du lien simple et de chaque classe la hiérarchie faible de Bandelt et Dress.

1 Introduction

La classification vise à révéler des regroupements homogènes, appelés classes, au sein d'une population (finie) d'objets. Dans ce texte, nous considérons les modèles de classification multi-niveaux, c'est-à-dire les modèles comportant des classes pouvant être incluses l'une dans l'autre, comme c'est le cas, par exemple, du modèle hiérarchique bien connu. Un autre modèle moins populaire, mais néanmoins central, est celui des hiérarchies faibles : une hiérarchie faible est une collection de parties non vides, stable par intersection non vide, contenant l'ensemble de tous les objets, et telle que l'intersection de trois classes est égale à l'intersection de deux d'entre les trois. La plupart de ces modèles requièrent que la classification soit stable par intersection non vide et contienne l'ensemble de tous les objets, noté S par la suite. Autrement dit selon la terminologie introduite par (Van de Vel, 1993), une classification multi-niveaux est une *convexité (abstraite)* privée de l'ensemble vide, et chacune de ses classes est donc une partie *convexe* non vide. Étant donné une convexité $\mathcal{C}$, il est naturel de considérer sa fonction segment $\mathrm{seg}(\mathcal{C})$ qui à tout couple $(x, y) \in S \times S$ associe la partie $\mathrm{seg}(\mathcal{C})(x, y) = \bigcap \{A \in \mathcal{C} \mid x, y \in A\}$. La fonction $\mathrm{seg}(\mathcal{C})$ est une *fonction intervalle*, i.e. une fonction symétrique qui à tout couple (x, y) d'objets d'un ensemble S, associe une partie de S contenant, a minima, les objets x et y. Si I est une fonction intervalle, l'ensemble $I(x, y)$ est

appelé I-*intervalle* d'extrémités x et y. Une partie A de S est dite I-*convexe* si elle contient tous les I-intervalles d'extrémités dans A. La collection $\mathcal{C}$ des parties de S qui sont I-convexes, est une convexité au sens défini ci-dessus : on l'appelle *convexité d'intervalle induite par* I, et on la note $\mathrm{conv}(I)$. En général, on a $\mathcal{C} \subseteq \mathrm{conv}(\mathrm{seg}(\mathcal{C}))$, et si $\mathcal{C}$ est faiblement hiérarchique, alors $\mathcal{C} = \mathrm{conv}(\mathrm{seg}(\mathcal{C}))$. Cette égalité permet de caractériser chacun des principaux modèles de classification multi-niveaux par une condition portant sur la fonction segment associée à la classification multi-niveaux (cf. Bertrand et Diatta (2017)). Cette approche a été récemment étendue au cas des hiérarchies k-faibles par Changat et al. (2019). Par la suite, nous adoptons un point de vue constructif qui permet de caractériser certaines des classifications multi-niveaux bien connues, comme par exemple la hiérarchie du lien simple, en tant que convexités induites par des fonctions intervalle définies simplement à partir de la seule dissimilarité, notée δ, qui compare les objets de l'ensemble S. Ce texte est organisé comme suit. La section 2 introduit un nouveau procédé général qui, à partir d'une fonction $g : S \times S \mapsto 2^S$ non nécessairement symétrique et vérifiant certaines conditions, construit une fonction d'intervalle qui induit soit une hiérarchie soit une hiérarchie faible. La section 3 considère le cas où g est la fonction boule au sens de la dissimilarité δ, et introduit un paramètre qui définit une suite de convexités d'intervalle. Cette suite est constituée de hiérarchies faibles emboîtées, la plus petite (resp. grande) d'entre elles au sens de l'inclusion étant la hiérarchie d'Asprejan (resp. la hiérarchie faible de Bandelt et Dress). La section 4 considère le cas où g a pour valeurs des unions de chemins, de longueur au plus ℓ, du graphe complet K_S valué par la dissimilarité δ. En faisant varier ℓ, il en résulte une suite croissante de hiérarchies qui va de la hiérarchie d'Asprejan à la hiérarchie du lien simple. En conclusion, la section 5 introduit des perspectives concernant notamment l'évaluation de la robustesse des classes d'une classification multi-niveaux.

2 Construction de hiérarchies et de hiérarchies faibles à l'aide d'une fonction d'intervalle

Par la suite, S désigne l'ensemble de base qui est supposé fini. Considérons une application de $g : S \times S \mapsto 2^S$ qui vérifie la condition suivante :

(C_0) Pour tout $x, y \in S,\ \{x, y\} \subseteq g(x, y)$.

Deux formules simples permettent de symétriser la fonction g, et ainsi définir deux fonctions d'intervalle sur S. Soient les applications J_g et M_g définies de $S \times S$ dans 2^S par :

$$\text{Pour tout } x, y \in S, \quad J_g(x, y) = g(x, y) \cup g(y, x),$$
$$\text{Pour tout } x, y \in S, \quad M_g(x, y) = g(x, y) \cap g(y, x).$$

Il est clair que J_g et M_g sont symétriques et vérifient (C_0), en d'autres termes J_g et M_g sont des fonctions d'intervalle définies sur S. Considérons les convéxités induites $\mathrm{conv}(J_g)$ et $\mathrm{conv}(M_g)$. Dans ce qui suit, les propriétés classificatoires de ces convexités d'intervalle sont déduites des conditions suivantes.

(H) pour tout $x_1, x_2, x_3 \in S$, on a $g(x_1, x_2) \subseteq g(x_1, x_3)$ ou $g(x_1, x_3) \subseteq g(x_1, x_2)$.

(W) pour tout $x_1, x_2, x_3 \in S$, il existe i, j, k avec $\{i, j, k\} = \{1, 2, 3\}$, tels que :
$$g(x_i, x_j) \subseteq g(x_i, x_k) \text{ et } g(x_k, x_j) \subseteq g(x_k, x_i).$$

(W′) pour tout $x_1, x_2, x_3 \in S$, il existe i, j, k avec $\{i, j, k\} = \{1, 2, 3\}$, tels que :
$$g(x_i, x_j) \subseteq g(x_i, x_k),\ g(x_k, x_j) \subseteq g(x_k, x_i) \text{ et } g(x_j, x_i) \subseteq g(x_j, x_k).$$

On vérifie aisément que (W′) $\Rightarrow$ (H) et (W′) $\Rightarrow$ (W), et qu'il n'existe pas d'autres implications entre les conditions (H), (W) et (W′). La proposition suivante montre que si g vérifie (C_0) et au moins l'une des conditions (H), (W) ou (W′), alors la convexité induite par J_g ou M_g est soit hiérarchique soit faiblement hiérarchique.

Proposition 2.1 *Soit g une application de $S \times S$ dans 2^S qui vérifie (C_0).*

(i) Si g vérifie (H)*, alors la convexité induite par J_g est hiérarchique.*

(ii) Si g vérifie (W)*, alors M_g induit une convexité faiblement hiérarchique, autrement dit* $\mathrm{conv}(M_g)$ *est faiblement hiérarchique.*

(iii) Si g vérifie (W′)*, alors J_g et M_g induisent des convexités qui sont respectivement hiérarchique et faiblement hiérarchique.*

Lemma 2.2 *Soit une application $g : S \times S \mapsto 2^S$ qui vérifie la condition (C_0). L'application g est symétrique si et seulement si $J_g = M_g$. Dans ce cas, on a $J_g = M_g = g$.*

3 De la hiérarchie d'Asprejan à la hiérarchie faible de Bandelt et Dress

Rappelons qu'une dissimilarité δ sur S est une application de $S \times S$ dans $\mathbb{R}^+$ telle que, pour tout $x, y \in S$, on a $\delta(x, y) = \delta(y, x) \geq \delta(x, x) = 0$. Plusieurs auteurs ont proposé de définir la notion de classe comme une partie de S possédant des degrés élevés de cohésion et d'isolation au sens des valeurs prises par la dissimilarité δ définie sur les données. Dans ce cadre général, Asprejan (1966) définit une classe comme une partie C qui vérifie :

$$\text{pour tout } x, y \in C,\ \delta(x, y) < \min_{z \notin C}\{\delta(x, z), \delta(y, z)\}. \tag{1}$$

Bandelt et Dress (1989) utilisent une condition plus faible pour définir une classe, i.e. :

$$\text{pour tout } x, y \in C,\ \delta(x, y) < \max_{z \notin C}\{\delta(x, z), \delta(y, z)\}. \tag{2}$$

Une partie C est appelée *classe d'Asprejan* (resp. *classe de Bandelt et Dress*) au sens de δ, si elle vérifie la condition (1) (resp. (2)). Étant donné une dissimilarité δ, $a \in S$ et $\rho \geq 0$, on note $B^c(a, \rho)$ la boule fermée de centre a et de rayon ρ. De plus si x, y sont deux éléments quelconques de S, nous adoptons les notations suivantes :

$$\begin{aligned}
g_B(x, y) &= B^c(x, \delta(x, y)) = \{s \in S \mid \delta(s, x) \leq \delta(x, y)\},\\
\mathbf{D}(x, y) &= g_B(x, y) \cup g_B(y, x) = \{z \in S \mid \min\{\delta(x, z), \delta(y, z)\} < \delta(x, y)\},\\
\mathbf{B}(x, y) &= g_B(x, y) \cap g_B(y, x) = \{z \in S \mid \max\{\delta(x, z), \delta(y, z)\} \leq \delta(x, y)\}.
\end{aligned}$$

Il est clair que $x, y \in g_B(x, y)$, autrement dit, g_B vérifie (C_0). On montre facilement que g_B vérifie (W′). Par ailleurs, $\mathbf{D} = J_{g_B}$ et $\mathbf{B} = M_{g_B}$, d'où le résultat suivant :

Corollary 3.1 *Les fonctions d'intervalle* $\mathbf{D}$ *et* $\mathbf{B}$ *induisent des convexités qui sont respectivement hiérarchique et faiblement hiérarchique.*

Proposition 3.2 *Pour toute dissimilarité* δ *définie sur* S*, on a :*

(i) *Une partie de* S *est une classe d'Asprejan si et seulement si elle est* $\mathbf{D}$*-convexe.*

(ii) *Une partie de* S *est une classe de Bandelt et Dress si et seulement si elle est* $\mathbf{B}$*-convexe.*

La proposition 3.2 et le corollaire 3.1 montrent que l'ensemble des classes d'Asprejan est une hiérarchie. Cette hierarchie est appelée *hiérarchie d'Asprejan* associée à la dissimilarité δ (cf. Asprejan (1966)). Cette proposition et ce corollaire montrent aussi que l'ensemble des classes de Bandelt et Dress forme une hiérarchie faible, appelée *hiérarchie faible de Bandelt et Dress* associée à la dissimilarité δ.
Considérons à présent deux fonctions d'intervalle I_1 et I_2 telles que $I_1(x, y) \subseteq I_2(x, y)$ pour tout $x, y \in S$. Si $A \in \mathrm{conv}(I_2)$, alors $I_1(x, y) \subseteq I_2(x, y) \subseteq A$ pour tout $x, y \in A$. Par conséquent $A \in \mathrm{conv}(I_1)$, d'où $\mathrm{conv}(I_2) \subseteq \mathrm{conv}(I_1)$. On en déduit la propriété suivante :

$$\text{Si, pour tout } x, y \in S, \text{ on a } I_1(x, y) \subseteq I_2(x, y), \text{ alors } \mathrm{conv}(I_2) \subseteq \mathrm{conv}(I_1). \tag{3}$$

Comme $\mathbf{B}(x, y) \subseteq \mathbf{D}(x, y)$ pour tout $x, y \in S$, il en résulte que la hiérarchie d'Asprejan est incluse dans la hiérarchie faible de Bandelt et Dress. Considérons à présent la famille de fonctions d'intervalles $\{I_\alpha\}_{0 \leq \alpha \leq 1}$ définie par :

$$I_\alpha(x, y) = \mathbf{B}(x, y) \cup \{z \in S \mid \min\{\delta(x, z), \delta(y, z)\} \leq \alpha\, \delta(x, y)\},$$

pour tout $x, y \in S$. Si δ est propre, il est clair que l'on a :

$$I_0(x, y) = \mathbf{B}(x, y) \subseteq I_\alpha(x, y) \subseteq I_{\alpha'}(x, y) \subseteq \mathbf{D}(x, y) = I_1(x, y),$$

pour tout $x, y \in S$, avec $0 \leq \alpha \leq \alpha' \leq 1$. D'après (3), on en déduit que si $(\alpha_n)_{0 \leq n \leq N}$ est une suite décroissante dans $[0, 1]$, avec $\alpha_0 = 1$ et $\alpha_N = 0$, alors $\{\mathrm{conv}(I_{\alpha_n})\}_{0 \leq n \leq N}$ est une suite croissante (au sens de l'inclusion) de hiérarchies faibles emboîtées, partant pour $n = 0$ de la plus simple, i.e. la hiérarchie d'Asprejan, à la plus complexe, i.e. la hiérarchie faible de Bandelt et Dress, obtenue pour $n = N$.

4 De la hiérarchie d'Asprejan à la hiérarchie du lien simple

Dans cette section, on note $\Gamma[\alpha]$ le graphe seuil (inférieur) de la dissimilarité δ au niveau α. Ce graphe, appelé aussi graphe de Vietoris-Rips, est défini comme étant le graphe qui admet S pour ensemble de sommets, et pour lequel une paire $\{x, y\} \subseteq S$ est une arête ssi $\delta(x, y) \leq \alpha$. Pour tout $\alpha \geq \mathrm{diam}(\delta)$, le graphe seuil $\Gamma[\alpha]$ coïncide avec le graphe complet défini sur S, noté K_S dans ce texte. Pour tout chemin $P = \{u_0, u_1, \ldots, u_m\}$ de K_S, on note :

$$\mathrm{val}(P) = \max_{1 \leq i \leq m} \delta(u_{i-1}, u_i).$$

Par la suite, ℓ désigne un entier compris entre 1 et $n - 1$. On note $n = |\, S \,|$, et l'ensemble des chemins de K_S de longueur au plus ℓ, est noté $\mathcal{P}^{(\ell)}$. Pour tout $x, y \in S$, nous notons

$\mathcal{P}_x^{(\ell)}$ le sous-ensemble de $\mathcal{P}^{(\ell)}$ constitué des chemins dont une extrémité est x, et on pose $\mathcal{P}_{x-y}^{(\ell)} = \mathcal{P}_x^{(\ell)} \cap \mathcal{P}_y^{(\ell)}$, i.e. $\mathcal{P}_{x-y}^{(\ell)}$ est l'ensemble des $x-y$ chemins $\mathcal{P}^{(\ell)}$. Avec ces notations, introduisons la dissimilarité π^{ℓ} définie, pour tout $x, y \in S$, par :

$$\pi^{\ell}(x,y) = \min_{P \in \mathcal{P}_{x-y}^{(\ell)}} [\mathrm{val}(P)].$$

Remark 4.1 Remarquons que lorsque $\ell \geq n-1$, alors $\mathcal{P}^{(\ell)}$ est l'ensemble de tous les chemins de K_S. Dans ce cas, écrivons simplement π au lieu de $\pi^{(\ell)}$ et $\mathcal{P}$ au lieu de $\mathcal{P}^{(\ell)}$. On vérifie aisément que π est une ultramétrique. Il est facile aussi de vérifier que π minore δ : il suffit de considérer le $x-y$ chemin de K_S, qui est constitué de la seule arête xy, et d'observer que :

$$\pi(x,y) = \min_{P \in \mathcal{P}_{x-y}} [\mathrm{val}(P)] \leq \delta(x,y),$$

pour tout $x, y \in S$. D'où $\pi \preceq \delta$, où $\preceq$ désigne la relation d'ordre partiel entre les dissimilarités définies sur S. En fait, il est bien connu dans la littérature que l'ultramétrique π est l'ultramétrique sous-dominante de δ.

Étant donné $\ell \in \{1, \ldots, n-1\}$, nous définissons l'application g^{ℓ} de $S \times S$ dans 2^S, par :

$$g^{\ell}(x,y) = \bigcup \{P \in \mathcal{P}_x^{(\ell)} \mid \mathrm{val}(P) \leq \pi^{\ell}(x,y)\}.$$

Il en résulte que $x, y \in g^{\ell}(x,y)$. Afin d'alléger cette présentation, nous simplifions quelques notations : pour tout $\ell \in \{1, \ldots, n-1\}$, on note J^{ℓ} et M^{ℓ} au lieu de $J_{g^{(\ell)}}$ et $M_{g^{(\ell)}}$, respectivement. Autrement dit, pour tout $x, y \in S$, on a :

$$\begin{aligned} J^{\ell}(x,y) &= g^{\ell}(x,y) \cup g^{\ell}(y,x), \\ M^{\ell}(x,y) &= g^{\ell}(x,y) \cap g^{\ell}(y,x). \end{aligned}$$

Rappelons que le *diamètre d'un sous-ensemble A de sommets d'un graphe G*, noté $\mathrm{diam}_G(A)$, est la distance [1] maximale entre deux sommets de A.

Proposition 4.2 *Pour toute dissimilarité δ, les propriétés suivantes sont vérifiées :*

(i) Pour tout $\ell \in \{1, \ldots, n-1\}$, l'application g^{ℓ} vérifie la condition (W′).

(ii) L'application g^{n-1} est symétrique.

(iii) Pour tout $\ell \in \{1, \ldots, n-1\}$, une partie A de S est J^{ℓ}-convexe si et seulement si il existe $\alpha \geq 0$ telle que A est une composante connexe du graphe seuil $\Gamma[\alpha]$ avec $\mathrm{diam}_{\Gamma[\alpha]}(A) \leq \ell$.

La propriété (iii) du corollaire suivant met en évidence une suite croissante de hiérachies emboîtées, allant de la hiérarchie d'Asprejan ($\ell = 1$) à la hiérarchie du lien simple ($\ell = n-1$).

Corollary 4.3 *Pour toute dissimilarité δ, on a*

(i) La convexité induite par $J^{(n-1)}$ coincide avec la hiérarchie du lien minimum.

(ii) La convexité induite par $J^{(1)}$ coincide avec la hiérarchie d'Asprejan.

(iii) Pour tout $\ell_1, \ell_2 \in \{1, \ldots, n-1\}$ tels que $\ell_1 \leq \ell_2$, on a $\mathrm{conv}J^{\ell_1} \subseteq \mathrm{conv}J^{\ell_2}$.

1. La distance entre deux sommets x et y d'un graphe G est définie comme étant le nombre d'arêtes d'un plus court chemin de G reliant x et y.

5 Conclusions

Nous introduisons ici deux suites de fonctions d'intervalle qui sont construites simplement à partir d'une dissimilarité, et qui, pour les valeurs limites de leur paramètre, induisent des classifications connues, i.e. la hiérarchie du lien simple, la hiérarchie d'Asprejan et la hiérarchie faible de Bandelt et Dress. En raison du caractère très exigeant de la définition de ses classes, la hiérarchie d'Asprejan ne contient que peu de classes, i.e. celles qui sont extrêmement séparées des objets qui leur sont extérieur, d'où son manque d'efficacité pour identifier des classes évidentes. À l'opposé, la hiérarchie du lien simple et la hiérarchie faible de Bandelt et Dress peuvent (pour des raisons différentes) identifier des classes artefacts de la méthode. Partant d'une classification multi-niveaux de l'un de ces deux derniers types, et considérant les convexités induites par des fonctions d'intervalle ayant pour valeurs des intervalles de taille de plus en plus grande, donc de plus en plus proche de la hiérarchie d'Asprejan, on peut ainsi identifier les classes de la hiérarchie du lien simple et celles de la hiérarchie faible de Bandelt et Dress, qui persistent lorsque le critère de classification est plus exigeant, ce qui donne une évaluation du degré de robustesse de chacune de ces classes.

Références

Asprejan, J. D. (1966). Un algorithme pour construire des classes d'après une matrice de distance. *Mashinnyi Perevod i Prikladnaja Lingvistika, Institut Maurice Thorez, Moscou 9*, 3–18.

Bandelt, H.-J. et A. Dress (1989). Weak hierarchies associated with similarity measures: an additive clustering technique. *Bull. Math. Biology 51*, 133–166.

Bertrand, P. et J. Diatta (2017). Multilevel clustering models and interval convexities. *Discrete Applied Mathematics 222*, 54–66.

Changat, M., P.-G. Narasimha-Shenoi, et P.-F. Stadler (2019). Axiomatic characterization of transit functions of weak hierarchies. *The Art of Discrete and Applied Mathematics 2*, #P1.01.

Van de Vel, M. (1993). *Theory of Convex Structures*. Amsterdam, North-Holland: Elsevier.

Summary

Several approaches have shown that hierarchies, and their generalizations, can be characterized as collections of interval function based convex clusters. In this note, we introduce interval functions built from an arbitrary dissmimilarity function, with a parameter enabling to tune the shape and the size of the intervals. The convexities induced by these interval functions are nested for two distinct values of the parameter. In this way, two sequences of nested clusterings can be produced, each going from the Aspresjan's hierarchy. One of them increases to the Bandelt and Dress weak hierarchy, while the other one increases to the single linkage hierarchy. These two sequences can be used to assess the robustness of each cluster of the single linkage hierarchy as well as of the Bandelt and Dress weak hierarchy.

Three-way clustering around latent variables approach with constraints to improve configurations' interpretability

Véronique Cariou*,
Tom F. Wilderjans**, ***, ****

* StatSC, ONIRIS, INRA, 44322 Nantes, France
veronique.cariou@oniris-nantes.fr
** Methodology and Statistics Research Unit, Institute of Psychology,
Faculty of Social and Behavioral Sciences, Leiden University, Pieter de la Court Building,
Wassenaarseweg, 52, 2333 AK Leiden, The Netherlands
t.f.wilderjans@fsw.leidenuniv.nl
*** Leiden Institute for Brain and Cognition (LIBC), Leiden, The Netherlands
**** Research Group of Quantitative Psychology and Individual Differences,
Faculty of Psychology and Educational Sciences, KU Leuven,
Tiensestraat 102, Box 3713, 3000 Leuven, Belgium

Abstract. With the amount of data generated at an ever-increasing rate, three-way data occur regularly in different domains as measurements of variables on a set of units/products at several occasions (or by several subjects). In parallel to ordination techniques, cluster analysis of variables is an important analysis technique that can detect heterogeneity within three-way data. For this purpose, CLV3W has been proposed, which extends the Clustering around Latent Variables (CLV) approach to a three-way data structure. CLV3W groups the variables from the second mode (e.g., attributes) into Q clusters and simultaneously determines, for each cluster, a rank-1 Parafac model providing: (1) a cluster-specific latent component associated with the first mode, (2) a vector of loadings representing the degree of closeness of each variable to its latent component and (3) a cluster-specific weighting system of the third mode. In this presentation, different constraints are proposed on the cluster-specific weighting systems and loading vectors that aim at improving the interpretability of the obtained configurations.

1. Introduction

In the context of two-way data, the clustering of variables approach has proven useful to discover latent structures in the data. In the statistics community, a well-known clustering of variables algorithm is the Varclus SAS/STAT procedure (Sarle, 1990). In this presentation, we focus on the Clustering around Latent Variables (CLV)

approach (Vigneau & Qannari, 2003), a method that is gaining ground in sensometrics and chemometrics. CLV aims at clustering variables along with summarizing each cluster by a latent component that captures the underlying dimension.

In the context of three-way data, the development of a clustering of variables strategy that integrates simultaneously the three modes is of paramount interest. In this regard, the extension of the CLV strategy has led to the development of CLV3W (Cariou & Wilderjans, 2016). CLV3W gives a clustering of the variables and estimates a rank-1 Parafac model per (variable) cluster, implying a latent component together with a weighting scheme associated with the third mode per cluster. Moreover, a loading is associated to each variable reflecting its degree of agreement with its group latent component. Herein, we propose several constraints on the CLV3W solution in order to provide configurations that are easier to interpret:

- Restricting the vector of loadings to only have positive values, thus assuming that the variables belonging to the same cluster are all positively correlated with their latent component;
- A common weighting system across clusters, thus assuming that the clusters share the same weighting values of the third mode occasions/subjects.

2. CLV3W for the clustering of variables within the scope of three-way data

Let us denote by $\boldsymbol{X}_j$ ($I \times K$) the j^{th} lateral slice of the three-way array $\underline{\boldsymbol{X}}$, which contains ratings for I products on J variables at K occasions (or alternatively by K subjects). Without loss of generality, we assume that all $\boldsymbol{X}_j$ ($j = 1, \ldots, J$) are column-wise centered.

2.1 CLV3W global criterion

The goal of the CLV3W analysis is to partition the J variables into Q clusters and to determine Q latent variables $\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_Q$ of length one (indicating product scores) along with normalized cluster-specific weights $\boldsymbol{w}_q$ such that the following function is maximized:

$$g = \sum_{j=1}^{J} \sum_{q=1}^{Q} p_{jq} cov^2(\boldsymbol{X}_j \boldsymbol{w}_q, \boldsymbol{t}_q)$$

with p_{jq} indicating whether variable j belongs ($p_{jq} = 1$) or not ($p_{jq} = 0$) to G_q.

As such, this is equivalent to minimize:

$$f = \sum_{j=1}^{J} \sum_{q=1}^{Q} p_{jq} \left\| \boldsymbol{X}_j - \alpha_{jq} (\boldsymbol{t}_q \boldsymbol{w}_q^{\mathrm{T}}) \right\|_F^2$$

where α_{jq} corresponds to the loading of the variable j in cluster G_q (with α_{jq} being zero when j does not belong to G_q) and $\|\cdots\|_F^2$ denoting the Frobenius norm. It is worth noting that: $\alpha_{jq} = cov(\boldsymbol{X}_j\boldsymbol{w}_q, \boldsymbol{t}_q)$ and reflects the degree of proximity between variable j block component, $\boldsymbol{X}_j\boldsymbol{w}_q$, and its associated cluster latent component, $\boldsymbol{t}_q$. This latter criterion is exactly the same as the Clusterwise Parafac criterion when applied to the second mode with Q clusters and one component in each cluster (Wilderjans & Ceulemans, 2013; Krijnen, 1993).

To partition the variables, CLV3W runs an ALS algorithm, which alternates two updating steps as follows: (1) To update the cluster membership of a variable j, the criterion $f_{jq} = \left\|\boldsymbol{X}_j - \alpha_{jq}\left(\boldsymbol{t}_q\boldsymbol{w}_q^{\intercal}\right)\right\|_F^2$ is computed for each cluster G_q and variable j is assigned to the cluster G_q for which f_{jq} is minimal; (2) After updating the cluster membership of all the variables, a Parafac model (Harshman, 1970; Carroll & Chang, 1970) with one component is carried out on each cluster G_q, that is to say on the three-way array that only consists of the variables belonging to G_q. This leads to an updated estimate for each G_q, its specific parameters $\boldsymbol{t}_q$, $\boldsymbol{w}_q$ and α_{jq} with $j \in G_q$. This procedure is repeated until convergence.

2.2 Constraint on the cluster-specific vector of loadings

In some cases, the goal of clustering is to find groups of variables which are homogeneous in the sense that the variables are as much related - in terms of a positive covariance - as possible with their cluster-specific latent variable. This constraint implies that for a variable j belonging to a particular cluster G_q the weighted average of its occasion scores $\boldsymbol{X}_j\boldsymbol{w}_q$ is positively related to the latent variable $\boldsymbol{t}_q$, which is associated to the cluster in question: $cov(\boldsymbol{X}_j\boldsymbol{w}_q, \boldsymbol{t}_q) \geq 0$. As it has been shown from the CLV3W optimization criterion that $\alpha_{jq} = cov(\boldsymbol{X}_j\boldsymbol{w}_q, \boldsymbol{t}_q)$, this constraint can be obtained by imposing a non-negativity constraint on the CLV3W optimization criterion :

$$g_{NN} = \sum_{j=1}^{J}\sum_{q=1}^{Q} p_{jq} cov^2(\boldsymbol{X}_j\boldsymbol{w}_q, \boldsymbol{t}_q), \text{ with } cov(\boldsymbol{X}_j\boldsymbol{w}_q, \boldsymbol{t}_q) \geq 0.$$

Alternatively, this is equivalent to minimizing the least squares loss function:

$$f_{NN} = \sum_{j=1}^{J}\sum_{q=1}^{Q} p_{jq}\left\|\boldsymbol{X}_j - \alpha_{jq}\left(\boldsymbol{t}_q\boldsymbol{w}_q^{\intercal}\right)\right\|_F^2, \text{ with } \alpha_{jq} \geq 0.$$

The modification of the previous ALS algorithm to fulfill the non-negativity constraint boils down to compute each α_{jq} given $\boldsymbol{t}_q$ and $\boldsymbol{w}_q$ by means of a non-negativity constrained linear regression.

In terms of interpretation, this leads to cluster variables into homogeneous groups, all variables within a group having the same main direction, in a similar rationale than consumers' segmentation in the scope of preference studies.

2.3 Constraint on the cluster-specific weighting system

In some situations, it may occur that the distribution of the weights is basically the same across the different clusters. This similarity in weight distributions may indicate that the variables share the same overall behavior among the third mode units (which can be occasions or alternatively subjects), that is to say they weight them in a similar way. It thereby suggests that the clusters of variables differ only in the way the first mode units are scored by each variable (cluster) across all occasions. Such a property leads to a simpler (more parsimonious) configuration, which may be easier to interpret. This corresponds to the following constrained optimization criterion:

$$g_W = \sum_{j=1}^{J}\sum_{q=1}^{Q} p_{jq} cov^2\left(\boldsymbol{X}_j \mathbf{w}, \boldsymbol{t}_q\right)$$

in which **w** is kept constant across clusters. Alternatively, this is equivalent to minimizing the least squares loss function:

$$f_W = \sum_{j=1}^{J}\sum_{q=1}^{Q} p_{jq} \left\| \boldsymbol{X}_j - \alpha_{jq}\left(\boldsymbol{t}_q \boldsymbol{w}^{\intercal}\right)\right\|_F^2.$$

This constraint requires a modification of step (2) of the ALS algorithm. The latent variables $\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_Q$ together with the common weighting system $\boldsymbol{w}$ are adjusted in an alternating fashion until convergence. In particular, after initializing $\boldsymbol{w}$, the two following steps are alternated until convergence:

- Each $\boldsymbol{t}_{\mathrm{q}}$ is updated as the left singular vector corresponding to the largest singular value of the $(I \times \#G_q)$ matrix, which consists in the subset of the weighted average of the occasions scores associated with the variables belonging to G_q: $\left[\cdots | \boldsymbol{X}_j \boldsymbol{w} | \cdots\right]_{j \in G_q}$.
- The common weight $\boldsymbol{w}$ is updated as the left singular vector associated with the largest singular value from of the $(K \times J)$ matrix $\left[\cdots | \boldsymbol{X}_j^{\intercal} \boldsymbol{t}_q | \cdots\right]_{j \in J}$.

This parsimonious model involves determining a single vector of weights, common to all clusters. This makes it easier to interpret: the vertical slices of the three-way array can then be aggregated on the basis of this overall weighting scheme into a two-way matrix so that clusters of variables can be represented on the score and loading plots of a principal component analysis performed on such a matrix.

3. Application

In sensory evaluation, there is an increasing appeal to new consumer-based methods that do not require any specific training. These methods such as emotions analysis or check-all-that-apply evaluations naturally generate three-way data structures where the three modes respectively correspond to samples, consumers and attributes. To illustrate the use of CLV3W with a non-negativity constraint, we consider a case study pertaining to consumer emotions associating fifteen affective

terms for a variety of twelve coffee aromas. Aromas were presented with pillboxes labelled with a random three-digit code. Eighty-four participants were asked to complete each rating (i.e., rating the odor of 12 aromas on 15 emotion terms) on a 5-point rating scale. More details regarding the dataset can be found in Cariou & Wilderjans (2018).

We analyzed the consumer' segmentation obtained with CLV3W and a non-negativity constraint with one up to ten clusters. Note that in this application the consumers (instead of the attributes) were clustered. The evolution of the loss criterion against the number of clusters led to retain a partition with two clusters. The product scores, corresponding to the latent component of each cluster, are depicted in Figure 1, while the attribute weighting system for each cluster is represented in Figure 2.

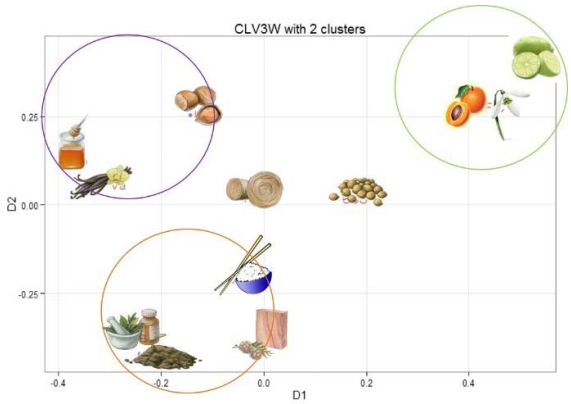


FIG. 1 – *Configuration of the products (i.e., latent components) for the two-cluster solution with D1 (D2) corresponding to the latent component for cluster 1 (cluster 2).*

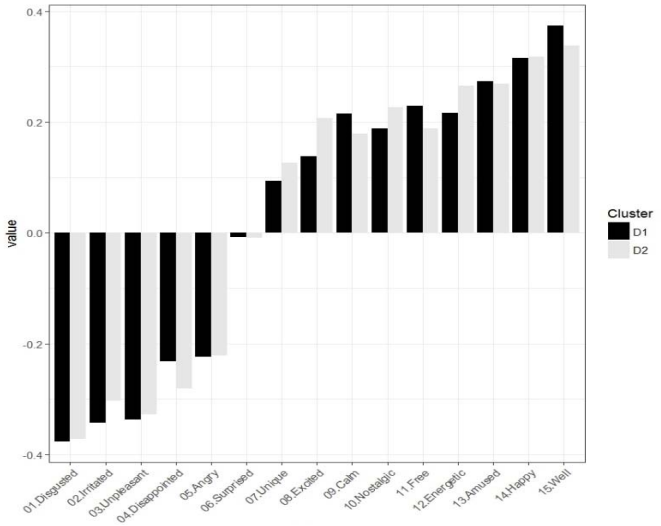


FIG. 2 – *Attribute weights for the two-cluster solution, with D1 (D2) denoting the weighting system associated with cluster 1 (cluster 2).*

When inspecting the product scores (see Figure 1), three sets of coffee aroma products can be identified. A first set of products, consisting of Basmati rice, Cedar, Earth, and Medicinal, has a negative score on the latent variable for each cluster. Secondly, Apricot, Flower coffee and Lemon aromas are encountered with positive scores on the two latent variables. It appears that consumers (clusters) more or less agree on the rating of these products. Three products stress the opposition between the two consumer clusters in the evaluation of the aromas. These products correspond to Hazelnut, Honey and Vanilla, which are three aromas that yield negative emotions with regard to the first consumer cluster, whereas they yield positive emotions for the second consumer cluster.

In Figure 2, cluster-specific attributes are presented in (more or less) ascending order according to the weighting system for each cluster. Looking at this order, one can associate it with the bipolar dimension of pleasant-unpleasant in which disgusted, irritated and unpleasant (i.e., having negative weights) are opposed to amused, happy and well emotions (i.e., positive weights). Remarkably, the distribution of the weights is basically the same across the two clusters and therefore a simpler (more parsimonious) clustering model is suggested that has a common weighting scheme across the clusters (as proposed in section 2.3).

4. Conclusion

In the cluster analysis framework, we proposed two different constraints on the CLV3W model which aims at identifying simultaneously subsets of variables and a latent component associated with each group, given a three-way structure. Compared to a classical approach consisting of performing a cluster analysis on each attribute slice of the three-way array, CLV3W offers an overall output that is easier to interpret and which does not require additional consensus methods to aggregate the various obtained partitions (one per attribute slice). It provides a crisp partition of variables which is easy to tune and to interpret by the user. Ongoing research concerns the adaptation of CLV3W to more complex data structures combining several blocks of data together with a three-way array such as the L-shape structure. Another perspective aims at extending our approach to the co-clustering of three-way data.

Références

Cariou, V., & Wilderjans, T. F. (2018). Consumer segmentation in multi-attribute product evaluation by means of non-negatively constrained CLV3W. *Food Quality and Preference*, *67*, 18-26.

Représentation condensée de règles d'association multidimensionnelles

Alexandre Bazin* [1], Aurélie Bertaux**, Christophe Nicolle**

*Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
**CIAD, Univ. Bourgogne Franche-Comté, UB, F-21000 Dijon, France.
contact@alexandrebazin.com,
aurelie.bertaux@iut-dijon.u-bourgogne.fr,
Christophe.Nicolle@u-bourgogne.fr

Résumé. La fouille de règles d'association est un problème qui a donné lieu à une littérature foisonnante, notamment dans les données binaires bidimensionnelles classiques. En particulier, la relation entre les ensembles fermés et les règles d'association est bien connue. Tel n'est pas le cas dans les données multidimensionnelles. Dans ce papier, nous montrons que la connaissance des n-ensembles fermés d'un tenseur booléen multidimensionnel est suffisante pour inférer la confiance de toutes les règles d'association multidimensionnelles.

1 Règles d'association dans le cas multidimensionnel

Nous présentons ici un résumé étendu des résultats publiés dans Bazin et al. (2019).

Nous appelons *dimension* un ensemble fini $\mathcal{D}_i = \{d_{i_1}, \ldots, d_{i_k}\}$ d'éléments de même nature. Soient $\mathcal{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_n\}$ un ensemble fini de dimensions et $\mathcal{R} \subseteq \prod_{\mathcal{D}_i \in \mathcal{D}} \mathcal{D}_i$ une *relation n-aire* entre les éléments des dimensions. Ensemble, $\mathcal{D}$ et $\mathcal{R}$ forment le *tenseur booléen* $\mathcal{T} = (\mathcal{D}, \mathcal{R})$, une matrice binaire n-dimensionnelle représentant des données. Ce tenseur est aussi appelé *contexte n-dimensionnel* ou *n-contexte* dans le domaine de l'analyse formelle de concepts. Le tenseur illustré dans la Figure 1 servira d'exemple tout au long de ce papier.

	p_1	p_2	p_3	p_1	p_2	p_3	p_1	p_2	p_3
c_1	×		×				×		
c_2		×		×	×	×	×		×
c_3			×		×			×	
		m_1			m_2			m_3	

FIG. 1 – *Tenseur tridimensionnel représentant des clients (c_1,c_2,c_3) achetant des produits (p_1,p_2,p_3) dans différents magasins (m_1,m_2,m_3).*

Différentes généralisations des règles d'association dans les relations n-aires ont été étudiées. Dans Nguyen et al. (2011), les auteurs proposent ce qui est, pour nous, la plus générale. Nous la présentons ici et l'utilisons dans le reste de ce travail.

1. Ce travail a été réalisé au sein du laboratoire CIAD.

Soit $D \subseteq \mathcal{D}$ un ensemble de dimensions. Sans perte de généralité, nous supposons que $D = \{\mathcal{D}_1, \ldots, \mathcal{D}_{|D|}\}$. Soit $X_d \subseteq \mathcal{D}_d$, $\mathcal{D}_d \in D$, un ensemble non vide d'éléments de la dimension $\mathcal{D}_d$. L'ensemble de tuples $\prod_{\mathcal{D}_d \in D} X_d$ est appelé une *association sur* D et D est appelé son *domaine*. Nos règles d'association seront entre deux telles associations. Nous utiliserons $dom(X)$ pour noter le domaine d'une association X.

Nous omettrons les accolades des ensembles lorsque cela n'induit pas d'ambiguité et nous utiliserons la notation $X.Y$ à la place de $X \times Y$ pour représenter le produit cartésien de deux ensembles. Dans notre exemple, p_1 et $p_3.m_1$ sont des associations sur, respectivement, les domaines $\{\mathcal{D}_{produits}\}$ et $\{\mathcal{D}_{produits}, \mathcal{D}_{magasins}\}$ et $p_1 \rightarrow p_3.m_1$ est une règle d'association.

Soient $\mathcal{D}_i$ une dimension et $X = \prod_{\mathcal{D}_d \in Dom(X)} X_d$ une association. La *projection* $\pi_{\mathcal{D}_i}(X)$ de X sur $\mathcal{D}_i$ est égale à X_i si $\mathcal{D}_i \in Dom(X)$ ou à $\emptyset$ sinon.

Dans notre exemple, $\pi_{\mathcal{D}_{produits}}(p_3.m_1) = p_3$, $\pi_{\mathcal{D}_{clients}}(p_3.m_1) = \emptyset$ et $\pi_{\mathcal{D}_{magasins}}(p_3.m_1) = m_1$.

Dans le cas bidimensionnel, le support d'une association sur une dimension est lié à un sous-ensemble de l'autre dimension : soit le sous-ensemble lui-même, soit sa cardinalité. Nous utilisons ici l'ensemble lui-même. De la même façon, dans le cas multidimensionnel, le support d'une association est calculé sur les dimensions qui ne sont pas dans son domaine. Soit X une association, le *support de* X, noté $s(X)$, est l'ensemble $\{t \in \prod_{\mathcal{D}_i \in \overline{dom(X)}} \mathcal{D}_i \mid \forall x \in X, x.t \in \mathcal{R}\}$ des tuples dans le produit cartésien des dimensions absentes du domaine de X qui forment un élément de $\mathcal{R}$ avec un élément de X.

Dans notre exemple, nous avons que $s(p_1) = \{(c_1, m_1), (c_2, m_2), (c_1, m_3), (c_2, m_3)\}$ et $s(p_3.m_1) = \{c_1, c_3\}$.

Soient X et Y deux associations. Leur *union* est l'association $X \sqcup Y$ telle que, pour tout $\mathcal{D}_i \in \mathcal{D}$, $\pi_{\mathcal{D}_i}(X \sqcup Y) = \pi_{\mathcal{D}_i}(X) \cup \pi_{\mathcal{D}_i}(Y)$. Le motif $X \rightarrow Y$ est une *règle d'association multidimensionnelle sur le domaine* $dom(X \sqcup Y)$ si et seulement si $X \sqcup Y$ est une association sur $dom(X \sqcup Y)$.

Dans notre exemple, $p_1 \rightarrow p_3.m_1$ est une règle d'association sur le domaine $\{\mathcal{D}_{produits}, \mathcal{D}_{magasins}\}$.

Soit $X \rightarrow Y$ une règle sur $dom(X \sqcup Y)$. Si $dom(X)$ est différent de $dom(X \sqcup Y)$, les supports $s(X)$ et $s(X \sqcup Y)$ sont définis sur des ensembles différents et ne peuvent donc pas être comparés pour calculer la confiance de la règle. Le support de la prémisse doit donc être défini différemment. Le support de X par rapport à un domaine $D \supseteq dom(X)$ est défini par

$$s_{\overline{D}}(X) = \{t \in \prod_{\mathcal{D}_d \in \overline{D}} \mathcal{D}_d \mid \exists u \in \prod_{\mathcal{D}_i \in D \setminus dom(X)} \mathcal{D}_i \text{ such that } \forall x \in X, x.u.t \in \mathcal{R}\}$$

Grâce à ce support, nous pouvons définir la *confiance naturelle* de $X \rightarrow Y$ sur le domaine $D = dom(X \sqcup Y)$ par

$$conf(X \rightarrow Y) = \frac{|s(X \sqcup Y)|}{|s_{\overline{D}}(X)|}$$

Dans notre exemple, $s_{\mathcal{D}_{clients}}(p_1) = \{c_1, c_2\}$ et $s(p_1 p_3.m_1) = \{c_1\}$. Ainsi, la confiance de $p_1 \to p_3.m_1$ est

$$\frac{|\{c_1\}|}{|\{c_1, c_2\}|} = \frac{1}{2}$$

Ces règles d'association multidimensionnelles conservent la propriété que

$$conf(X \to Y) = conf(X \to X \sqcup Y)$$

Le nombre de ces règles est, évidemment, encore plus élevé que dans le cas bidimensionnel. Dans Nguyen et al. (2011), les auteurs utilisent la fréquence et la confiance pour le réduire. Il paraîtrait donc naturel d'imiter le cas bidimensionnel et de représenter aussi l'ensemble des règles d'associations n-dimensionnelles avec des n-ensembles fermés. Cependant, peu de résultats existent sur le sujet.

1.1 Transformations de tenseurs

Cette section présente la définition de transformations de tenseurs que nous utilisons dans la suite.

Soient $\mathcal{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_n\}$ un ensemble de dimensions, $\mathcal{R} \subseteq \mathcal{D}_1 \times \cdots \times \mathcal{D}_n$ une relation n-aire et $\mathcal{T} = (\mathcal{D}, \mathcal{R})$ un tenseur. Soient $D \subseteq \mathcal{D}$ un sous-ensemble des dimensions et $\mathcal{D}_d \in D$ une dimension.

La principale opération réalisable sur le tenseur consiste en sa restriction à un sous-ensemble de l'une de ses dimensions. Soient $X_d \subseteq \mathcal{D}_d$ un ensemble d'éléments de la dimension $\mathcal{D}_d$ et $X_D = \{X_{j_1}, \ldots, X_{j_{|D|}}\}$ une collection d'ensembles d'éléments des dimensions dans D. Le tenseur $\mathcal{T}_{X_d} = (\mathcal{D} \setminus \mathcal{D}_d, \mathcal{R}_{X_d})$ avec

$$\mathcal{R}_{X_d} = \{(x_1, \ldots, x_{d-1}, x_{d+1}, \ldots, x_n) \mid \forall x_d \in X_d, (x_1, \ldots, x_d, \ldots, x_n) \in \mathcal{R}\}$$

est construit en intersectant les "couches" de $\mathcal{T}$ correspondantes aux éléments de X_d. Le résultat est un tenseur $(n-1)$-dimensionnel. Lorsque plusieurs dimensions sont fixées simultanément, nous écrivons $\mathcal{T}_{X_D} = (((\mathcal{T}_{X_{j_1}})_{X_{j_2}}) \ldots)_{X_{j_{|D|}}}$. La Figure 2 illustre cette transformation.

	p_1	p_2	p_3
c_1	×		×
c_2		×	
c_3			×

p_1	p_2	p_3
×		×

	m_1	m_2	m_3
c_1	×		
c_2		×	×
c_3			

FIG. 2 – *Transformations $\mathcal{T}_{m_1}$, $\mathcal{T}_{m_1,c_1}$ et $\mathcal{T}_{p_1,p_3}$ du tenseur $\mathcal{T}$ présenté dans la Figure 1.*

2 Dériver le support d'associations

Nous cherchons à identifier un ensemble restreint de règles d'association multidimensionnelles suffisantes pour dériver la confiance de toutes les autres. Pour ce faire, nous commencerons par montrer que la taille du support de n'importe quelle association intéressante peut être dérivée de la taille des supports de n-ensembles fermés. Nous supposons uniquement qu'une des dimensions n'apparait dans le domaine d'aucune association intéressante. Nous estimons cette supposition raisonnable car, en pratique, une dimension contient habituellement les "objets" ou "transactions" et ses éléments n'apparaissent pas dans les règles. Sans perte de généralité, nous supposerons que cette dimension est $\mathcal{D}_1$. Les preuves des propositions ont été publiées dans Bazin et al. (2019).

Le tenseur n-dimensionnel $\mathcal{T}$ peut être vu comme un empilement de tenseurs $(n-1)$-dimensionnels. De ce fait, la taille du support d'une association X est la somme des tailles de ses supports dans les différentes couches composant le tenseur.

Proposition 1. *Soit X une association. Soit*

$$S = \prod_{\mathcal{D}_i \in \mathcal{D} \setminus (dom(X) \cup \mathcal{D}_1)} \mathcal{D}_i$$

le produit cartésien de toutes les dimensions support de X à l'exception de $\mathcal{D}_1$. Nous avons que

$$|s(X)| = \sum_{d \in S} |s_d|$$

tel que s_d est le support de X dans $\mathcal{T}_d$.

Supposons que nous voulons connaître la taille des supports de p_1 et p_1p_3 dans notre exemple $\mathcal{T}$ et que $\mathcal{D}_{clients}$ est la dimension n'apparaissant pas dans les règles. Le produit cartésien de la seule dimension support qui n'est pas $\mathcal{D}_{clients}$ est $\{m_1, m_2, m_3\}$. Les supports de p_1 dans $\mathcal{T}_{\{m_1\}}$, $\mathcal{T}_{\{m_2\}}$ et $\mathcal{T}_{\{m_3\}}$ sont, respectivement, de taille 1, 1 et 2 donc $|s(p_1)| = 4$ dans $\mathcal{T}$. Les tailles des supports de p_1p_3 dans les mêmes tenseurs sont 1, 1 et 1 donc $|s(p_1p_3)| = 3$ dans $\mathcal{T}$. Ainsi, $p_1 \rightarrow p_1p_3$ a une confiance de $\frac{3}{4}$.

Proposition 2. *Soient X une association dans $\mathcal{T}$ et Z un élément du produit cartésien de dimensions supports de X. Le support de X dans $\mathcal{T}_Z$ est le support de $X \sqcup Z$ dans $\mathcal{T}$.*

Des Propositions 1 et 2, nous pouvons déduire que le support de n'importe quelle association dans $\mathcal{T}$ peut être dérivé des supports des associations sur le domaine $\overline{\mathcal{D}_1}$.

3 Règles entre associations sur différents domaines

Dans la Section 2, nous avons montré que la taille du support de n'importe quelle association peut être dérivée des tailles des supports d'associations sur $\overline{\mathcal{D}_1}$. Ce résultat est suffisant lorsque nous voulons calculer la confiance d'une règle entre deux associations sur le même domaine. Cependant, les règles de la forme $X \rightarrow Y$ avec $dom(X) \subset dom(X \sqcup Y)$ requièrent la

connaissance du support $s_{\overline{dom(X \sqcup Y)}}(X)$ de X par rapport à $dom(X \sqcup Y)$. Dans cette section, nous montrons que le tenseur peut être transformé pour unifier les domaines des prémisses et conclusions de façon à ce que le support de n'importe quelle association par rapport à n'importe quel domaine puisse être dérivé d'associations sur $\overline{\mathcal{D}_1}$.

Comme nous l'avons présenté dans la Section 1, le support d'une association X par rapport à un domaine D est l'union des supports de toutes les associations pouvant être construites en augmentant de façon minimale X pour que D soit son domaine. Ce support peut être vu comme le support d'une association augmentée de façon à ce que chaque dimension additionnelle contienne un élément qui représente une disjonction sur l'entièreté de la dimension.

Définition 3. *Soit $\mathcal{T} = (\mathcal{D}_1, \ldots, \mathcal{D}_n, \mathcal{R})$ un tenseur. Nous définissons le tenseur $\mathcal{T}^\uparrow$ par $\mathcal{T}^\uparrow = (\mathcal{D}_1, \mathcal{D}_2 \cup \{\vee_2\}, \ldots, \mathcal{D}_n \cup \{\vee_n\}, \mathcal{R}^\uparrow)$ avec*

$$\mathcal{R}^\uparrow = \mathcal{R} \cup \{(x_1, \ldots, x_n) \mid \forall x_i = \vee_i, \exists x'_i \neq x_i \textit{ such that } (x_1, \ldots, x'_i, \ldots, x_n) \in \mathcal{R}^\uparrow\}$$

En d'autres termes, le tenseur $\mathcal{T}^\uparrow$ est construit à partir de $\mathcal{T}$ en ajoutant un élément à chaque dimension (à l'exception de $\mathcal{D}_1$) et en projetant les croix sur ces éléments jusqu'à saturation tel qu'illustré dans la Figure 3.

	p_1	p_2	p_3	$\vee_p$	p_1	p_2	p_3	$\vee_p$	p_1	p_2	p_3	$\vee_p$	p_1	p_2	p_3	$\vee_p$
c_1	×		×	×					×			×	×		×	×
c_2		×		×	×	×	×	×	×		×	×	×	×	×	×
c_3			×	×		×		×		×		×		×	×	×
	m_1				m_2				m_3				$\vee_m$			

FIG. 3 – *Le tenseur $\mathcal{T}^\uparrow$ correspondant à notre exemple $\mathcal{T}$ de la Figure 1. $\mathcal{D}_{clients}$ joue le rôle de $\mathcal{D}_1$.*

Définition 4. *Soient X une association dans $\mathcal{T}$ et $D \supseteq dom(X)$ un domaine. $X^{D\uparrow}$ est l'association dans $\mathcal{T}^\uparrow$ telle que*

$$\pi_{\mathcal{D}_i}(X^{D\uparrow}) = \begin{cases} \pi_{\mathcal{D}_i}(X) \cup \{\vee_i\} & \textit{si } \mathcal{D}_i \in D \\ \emptyset & \textit{sinon} \end{cases}$$

Proposition 5. *Soient X une association dans $\mathcal{T}$ et $D \supseteq dom(X)$ un domaine. $s_{\overline{D}}(X)$ dans $\mathcal{T}$ est égal à $s(X^{D\uparrow})$ dans $\mathcal{T}^\uparrow$.*

La Proposition 5 implique que le support de X par rapport à un domaine D dans $\mathcal{T}$ est le même que le support de $X^{D\uparrow}$ dans $\mathcal{T}^\uparrow$. A partir de cela et des Propositions 1 et 2, nous pouvons déduire que la taille du support de n'importe quelle association X par rapport à n'importe quel domaine dans $\mathcal{T}$ peut être dérivé des tailles des supports d'associations sur $\overline{\mathcal{D}_1}$ dans $\mathcal{T}^\uparrow$.

4 Fermés et supports

Dans les Sections 2 et 3, nous avons dit que le support de n'importe quelle association par rapport à n'importe quel domaine peut être dérivé des supports des associations sur $\overline{\mathcal{D}_1}$ dans le tenseur $\mathcal{T}^{\uparrow}$. Il ne nous reste plus qu'à montrer que la connaissance de l'ensemble des n-ensembles fermés est suffisante pour retrouver ces supports.

Soit X une association sur le domaine $\overline{\mathcal{D}_1}$ dans $\mathcal{T}^{\uparrow}$. De par la définition d'une association et de son support, nous savons que $s(X).X \subseteq \mathcal{R}$. En d'autres termes, $s(X).X$ est une boîte de croix n-dimensionnelle dans le tenseur. Elle n'est pas nécessairement maximale sur toutes les dimensions mais le support lui-même l'est. De ce fait, il y a au moins un n-ensemble fermé $(s(X), C_2, \ldots, C_n)$ avec $\pi_{\mathcal{D}_i}(X) \subseteq C_i$, $\forall i \in \{2, \ldots, n\}$. Cela implique que $s(X) = s(\prod_{i\in\{2,\ldots,n\}} C_i)$. Dans le cas bidimensionnel, il n'y a qu'un tel 2-ensemble fermé pour X. Lorsque $n \geq 3$, il peut y en avoir plusieurs.

Définition 6. *Pour une association X, $c(X)$ désigne l'association résultante du produit cartésien des $n-1$ derniers composants d'**un** des n-ensembles fermés $(s(X), C_2, \ldots, C_n)$ avec $\pi_{\mathcal{D}_i}(X) \subseteq C_i$, $\forall i \in \{2, \ldots, n\}$.*

Par exemple, $c(\vee_p.m_1) = \vee_p.m_1m_3\vee_m$ parce que $s(\vee_p.m_1) = \{c_1c_2c_3\}$ et le triplet $(c_1c_2c_3, \vee_p, m_1m_3\vee_m)$ est un 3-ensemble fermé dans $\mathcal{T}^{\uparrow}$ (Figure 3).

Puisque X et $c(X)$ ont le même support, la connaissance de tous les n-ensembles fermés de $\mathcal{T}^{\uparrow}$ est suffisante pour dériver le support de n'importe quelle association X sur $\overline{\mathcal{D}_1}$ et donc de toute autre association, permettant ainsi de calculer la confiance de n'importe quelle règle d'association. Par extension, les règles de la forme $c(X) \rightarrow c(Y)$ telles que $c(X) \subseteq c(Y)$ sont suffisantes pour résumer toutes les autres règles.

Références

Bazin, A., A. Bertaux, et C. Nicolle (2019). Représentation condensée de règles d'association multidimensionnelles. In *Extraction et Gestion des Connaissances : Actes de la conférence EGC'2019*, Volume 79. BoD-Books on Demand.

Nguyen, K.-N. T., L. Cerf, M. Plantevit, et J.-F. Boulicaut (2011). Multidimensional association rules in boolean tensors. In *Proceedings of the 2011 SIAM International Conference on Data Mining*, pp. 570–581. SIAM.

Summary

Association rules mining is a problem that gave rise to a rich literature, especially in classic binary bidimensional data. In particular, the relation between closed sets and association rules is well understood. This is not the case in multidimensional data. In this paper, we show that the knowledge of the closed n-sets of a multidimensional boolean tensor is enough to allow for the derivation of the confidence of every multidimensional association rule.

On Entropy in Pattern Mining

Tatiana Makhalova[*,**] Sergei O. Kuznetsov[**]
Amedeo Napoli[*]

[*]National Research University Higher School of Economics, Moscow, Russia
{tpmakhalova,skuznetsov}@hse.ru,
https://cs.hse.ru/en/
[**]Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
{firstname.secondname}@loria.fr
http://www.loria.fr/

Résumé. In this paper we consider different entropy-based approaches to Pattern Mining. We discuss how entropy on pattern sets can be defined and how it can be incorporated into different stages of mining, from computing candidates to interesting patterns to assessing quality of pattern sets.

1 Introduction

Information theory (IT) is now a widely used framework in Machine Learning (MD) and Data Mining (DM). In this paper we give an overview on application of a fundamental concept of IT, namely, entropy, in Pattern Mining (PM). PM takes an important place in Data Science, and has many applications related to computing classes of patterns generated under specific objectives (Aggarwal et Han, 2014). To apply PM methods under supervised settings (e.g., Subgroup Discovery, classification) one needs to use an objective that takes into account true class labels. The objective of unsupervised ML problems deals with a pattern as a subset of attributes and objects this pattern describes.

A generic objective of PM is to discover a small set of non-redundant and interesting patterns that describe together a large portion of data and that can be easily interpreted. There are two approaches to define pattern "interestingness", namely, *static* and *dynamic* (Aggarwal et Han, 2014). The static approaches envelop a large number of interestingness measures (Kuznetsov et Makhalova, 2018). The patterns are mined under non-changeable assumptions about interestingness. For example, in frequent PM, one assumes that all the patterns with a support greater than a minimum threshold are interesting. Usually, a set of discovered patterns is redundant, i.e., it contains a lot of similar patterns. This problem is solved by post-processing pruning. Apart of redundancy, the use of an interestingness measure is quite subjective and most of the time it is not easy to provide explanation or justification about using one measure w.r.t. some others.

In paper of (Aggarwal et Han, 2014), it is argued that instead of finding *all the patterns that satisfy some given constraints* (the concern of static approachs) one should ask for a small (easily interpretable) and non-redundant (with high diversity) set of interesting patterns. This

is precisely what dynamic approaches are aimed at. A dynamic approach to PM implies taking into account initial assumptions, e.g., background knowledge, and then adding gradually patterns that "add some new knowledge" to the current pattern set. Most of existing dynamic approaches (Vreeken et al., 2011; Siebes et Kersten, 2011; Smets et Vreeken, 2012) are based on Minimum Description Length (MDL) principle (Grünwald, 2007) that is aimed at selecting a pattern set that compresses a dataset at most.

Pattern mining is generally performed in two steps (i) computing a candidate pattern set, a search space for interesting patterns, (ii) selection interesting ones. In (i), the search space is restricted to frequent patterns (Vreeken et al., 2011), low-entropy sets (Heikinheimo et al., 2009), a set where patterns ensure the maximal entropy (Mampaey et al., 2012), or other types of patterns (Gallo et al., 2007). Step (ii) consists in selecting patterns that satisfy a chosen criteria, i.e., an interestingness measure (static approaches) or a greedy strategy for extending a pattern set by patterns that bring "something new" into the pattern set (dynamic approaches).

In this paper we discuss how entropy can be applied at every step of PM, namely, generating candidates, mining itself, and assessing the quality of pattern sets.

The paper is organized as follows. In Section 2 we recall the main notions used in the paper. In Section 3 we discuss how entropy can be incorporated in PM. In Section 4 we conclude and give the direction of future work.

2 Basic notions

In this paper we consider transaction databases. Since any transactional database or categorical dataset can be trivially converted into a binary dataset, in this paper we use binary datasets. In transactional databases, patterns are also called itemsets. We present (closed) itemsets in the framework of Formal Concept Analysis (Ganter et Wille, 1999).

2.1 Formal Concept Analysis

A formal context is a triple (G, M, I), where $G = \{g_1, g_2, ..., g_n\}$ is called a set objects, $M = \{m_1, m_2, \ldots, m_k\}$ is called a set attributes and $I \subseteq G \times M$ is a relation called incidence relation, i.e. $(g, m) \in I$ if the object g has the attribute m. The derivation operators $(\cdot)'$ are defined for $A \subseteq G$ and $B \subseteq M$ as follows :

$$A' = \{m \in M \mid \forall g \in A : gIm\}\,,\ B' = \{g \in G \mid \forall m \in B : gIm\}\,.$$

A' is the set of attributes common to all objects of A and B' is the set of objects sharing all attributes of B. An object g is said to contain a pattern (set of items) $B \subseteq M$ if $B \subseteq g'$. The double application of $(\cdot)'$ is a closure operator. Sets $A \subseteq G$, $B \subseteq M$, such that $A = A''$ and $B = B''$, are said to be closed.

A (formal) concept is a pair (A, B), where $A \subseteq G$, $B \subseteq M$ and $A' = B$, $B' = A$. A is called the (formal) extent and B is called the (formal) intent of the concept (A, B). The *support* of an itemset I is defined as follows : $sup(I) = |\{g \mid g \in G, I \subseteq g'\}|$. An itemset I is *frequent* with threshold q if $sup(I) \geq q$. Formal concept has the twofold nature, since it can be considered as a set of objects and attributes. We discuss in Section 3.3 that entropy that takes into account this duality allows for computing pattern sets of better quality.

In PM closed itemsets are of a big importance since (i) a closed itemset is a maximal set that embodies all the patterns with the same frequency, (ii) a closed itemset provides a lossless representation of these patterns.

2.2 Entropy and related notions

Entropy is a central notion of IT, where entropy or mutual information are used for assessing data compression and transmission. The both notions are functions of the probability distribution that underlies a describing process.

The entropy $H(X)$ of a discrete random variable X is defined by

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x).$$

In this paper we will use logarithms to the base 2, thus the entropy then is measured in bits. The entropy is a measure of the average uncertainty in the random variable, i.e., the number of bits required on the average to describe the random variable.

The mutual information (MI), as a measure of the dependence between two random variables X and Y, is defined as

$$I(X,Y) = I(Y,X) = H(X) - H(X|Y) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y) \log \frac{p(x,y)}{p(x)p(y)},$$

where $H(X|Y) = -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y) \log \frac{p(x,y)}{p(y)}$ is a conditional entropy. MI is a special case of relative entropy $D(p||q) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)}$, that is the "distance" between two probability mass functions p and q. Relative entropy is not a true metric, it has some of the metric properties, e.g., $D(p||q) \geq 0$ and $D(p||q) = 0$ iif $p = q$. However, there exist entropy-based distance measures (De Mántaras, 1991; Wang, 2012). The properties of entropy, MI can be found in (Cover et Thomas, 2012).

3 Entropy in Pattern Mining

As it was mentioned above, a pattern can be considered not only a set of attributes, but also as a set of objects it describes. Thus entropy in PM can be defined in several ways.

3.1 Object-based entropy

Considering patterns (itemsets) in terms of objects is more common under supervised settings, where for all objects their class labels are available. For example, cross-entropy (Hastie et al., 2002) is used for building decision trees. In the unsupervised settings entropy can be defined in the similar way, i.e., $H(X) = -p_X \log p_X - (1-p_X)\log(1-p_X)$, where $p_X = |X'|$.

To evaluate diversity of a pattern set $\mathcal{X}$ we introduce *shattering matrix* as a $|G| \times |\mathcal{X}|$ binary matrix induced by a set of columns $\{|X|' \mid X \in \mathcal{X}\}$. The (normalized) entropy of $\mathcal{X}$ is then given by $H(\mathcal{X}, \mathcal{D}) = -\sum_{r \in \mathcal{Q}} p_r \log(p_r)$ ($H_N(\mathcal{X}, \mathcal{D}) = -H(\mathcal{X}, \mathcal{D})/\log|\mathcal{Q}|$), where $\mathcal{Q}$ is a set of unique rows of the shattering matrix, $p_r = n_r/|G|$, n_r is the support of row $r \in \mathcal{Q}$. The

t_1	$A\,B\,C$
t_2	$B\,C\,D\,E$
t_3	$D\,E$
t_4	$A\ \ C\,D\,E$
t_5	$A\ \ C$

(a)

	AC	DE	CDE	BC
t_1	×			×
t_2		×	×	×
t_3		×		×
t_4	×	×	×	
t_5	×			

(b)

t_1	$A\,B\,C$
t_2	$B\,C\,D\,E$
t_3	$D\,E$
t_4	$A\ \ C\,D\,E$
t_5	$A\ \ C$

(c)

X	$usage(X)$	$P(X)$
AC	3	3/8
DE	3	3/8
BC	1	1/8
B	1	1/8

(d)

FIG. 1 – *A binary dataset (a) , a shattering matrix induces by closed itemsets of frequency at least 2 (b), a covering by patterns AC, DE, BC and B (c), and the probability distribution AC, DE, BC and B induced by the covering (d).*

entropy $H(\mathcal{X}, \mathcal{D})$ characterizes diversity of all possible groups of objects that can be induced by combinations of patterns. The normalized entropy $H_N(\mathcal{X}, \mathcal{D})$ characterises "skewness" of the frequency distribution of the obtained groups.

Example. Let us consider an example in Fig. 1. Entropy of the shattering matrix (b) for dataset $\mathcal{D}$ (a) induced by patterns $\mathcal{X} = \{AC, DE, CDE, BC\}$ is $H(\mathcal{X}, \mathcal{D}) = -5 \cdot 1/5 \log(1/5) = 2.32$, since all the rows in the shattering matrix are different.

It is clear to see that $\mathcal{Q}$ is a partition of G. Let $PART(G)$ be collection of partitions on G. The function $d : PART(G) \times PART(G) \to R_{\geq 0}$ given by $d(\mathcal{P}, \mathcal{Q}) = H(\mathcal{P}|\mathcal{Q}) + H(\mathcal{Q}|P)$, where $\mathcal{P}, \mathcal{Q} \in PART(G)$, is a metric on $PART(G)$ (De Mántaras, 1991).

The object-based entropy of patterns can be defined differently. Let us consider a cover $\mathcal{C}$ of binary dataset $\mathcal{D}$ by patterns $\mathcal{X}$, where every object is covered by a set of disjoint patterns from $\mathcal{X}$. The loglikelihood of $\mathcal{X}$ w.r.t. cover $\mathcal{C}$ is defined as $l(\mathcal{C}) = \sum_{x \in \mathcal{X}} usage(X) \log P(X)$, where $usage(X)$ is frequency of X in $\mathcal{C}$ and probability of X is given by

$$P(X) = \frac{usage(X)}{\sum_{X^* \in \mathcal{X}} usage(X^*)}. \quad (1)$$

It follows directly from the formulas above that entropy of $\mathcal{X}$ under the given probability distribution is related to the loglikelihood as follows : $(\sum_{X \in \mathcal{X}} usage(X)) \cdot H(\mathcal{X}) = -l(\mathcal{C})$.

Example. The entropy of the pattern set in Fig. 1 w.r.t. the probability distribution (d) induced by a covering (c) is equal to $H(\mathcal{X}) = -2 \cdot (3/8 \log 3/8 + 1/8 \log 1/8) = 1.81$.

In the supervised settings, a partition can be reformulated in terms of classification, i.e., the rows of a shattering matrix correspond to classes of objects. That point of view gave raise to normalized/expected mutual information and the adjusted Rand index (Vinh et al., 2009). Some variations of these measures were proposed in (Vinh et al., 2010). In (Rosenberg et Hirschberg, 2007) it was proposed to assess homogeneity and completeness of classification (or clustering, if the ground true is known) using conditional entropy of two labelings.

3.2 Attribute-based Approaches

Similarly to object-based entropy, we can define entropy on an attribute set M. Moreover, the probability of singleton patterns $m \in M$ (see Formula 1) can be used to define the length

m	$P(\{m\})$	l_m
A	3/15	l_A
B	2/15	l_B
C	4/15	l_C
D	3/15	l_D
E	3/15	l_E

(a)

X	$P(X)$	l_X
AC	3/8	l_{AC}
DE	3/8	l_{DE}
BC	1/8	l_{BC}
B	1/8	l_B

(b)

t_1	$l_{AC} + l_B$
t_2	$l_{BC} + l_{DE}$
t_3	l_{DE}
t_4	$l_{AC} + l_{DE}$
t_5	l_{AC}

(e) $L(\mathcal{D}|CT)$

X	$length(X)$	l_X
AC	$l_A + l_C$	l_{AC}
DE	$l_D + l_E$	l_{DE}
BC	$l_B + l_C$	l_{BC}
B	l_B	l_B

(b) $L(CT|\mathcal{D})$

FIG. 2 – *Patterns and encoding of a dataset from Fig. 1 : (a) singletons and their associated code length $l_m = -\log P(\{m\})$; (b) a code table corresponding to covering given in Fig. 1, (c); (c) encoding of dataset by patterns given in Fig. 2, (b); (d) encoding of patterns in the code table.*

of pattern $X \subseteq M$ under the Shannon code scheme as

$$length(X) = -\sum_{m \in X} \log P(\{m\}). \tag{2}$$

3.3 Combined Entropy

In Sections 3.1 and 3.2 we considered different entropy-based approaches to assessing/mining pattern sets. They are based either on object or attribute distributions. The modern methods for PM are based on objectives that use the both entropy types (Vreeken et al., 2011; Siebes et Kersten, 2011; Smets et Vreeken, 2012). All of them mine patterns under the Minimum Description Length principle (Grünwald, 2007). The goal is to minimize the two-part description length $L(\mathcal{D}, CT) = L(CT|\mathcal{D}) + L(\mathcal{D}|CT)$, where CT is a two-column code table, that contains patterns and their code lengths, and $\mathcal{D}$ is a binary dataset. The length of dataset $\mathcal{D}$ encoded by patterns from CT is given by $L(\mathcal{D}|CT) = \sum_{X \in CT} usage(X) \cdot l_X$. The length of CT is given by length of its right and left columns, i.e., $L(CT|\mathcal{D}) = \sum_{X \in CT} length(X) + l_X$, where $length(X)$ is given in Formula 2 and $l_X = -\log P(X)$, probability $P(X)$ is computed by Formula 1. An example of encoding is s given in Fig. 2. The details on the presented MDL-approach can be found in (Vreeken et al., 2011).

4 Conclusion

In this paper we consider how entropy can be incorporated in Pattern Mining for transactional databases (categorical/binary datasets). The most successful approaches are based on the combination of object- and attribute-based entropies (based on MDL principle).

One of the most challenging directions of future work is the adaptation of entropy-based measures to numerical Pattern Mining.

Références

Aggarwal, C. C. et J. Han (2014). *Frequent pattern mining*. Springer.

Cover, T. M. et J. A. Thomas (2012). *Elements of information theory*. John Wiley & Sons.

De Mántaras, R. L. (1991). A distance-based attribute selection measure for decision tree induction. *Machine learning 6*(1), 81–92.

Gallo, A., T. De Bie, et N. Cristianini (2007). Mini : Mining informative non-redundant itemsets. In *PKDD*, pp. 438–445. Springer.

Ganter, B. et R. Wille (1999). Formal concept analysis : Logical foundations. Springer Verlag Berlin, RFA.

Grünwald, P. D. (2007). *The minimum description length principle*. MIT press.

Hastie, T., R. Tibshirani, et J. Friedman (2002). The elements of statistical learning ; data mining, inference and prediction.

Heikinheimo, H., A. Siebes, J. Vreeken, et H. Mannila (2009). Low-entropy set selection. In *Proceedings of SIAM*, pp. 569–580. SIAM.

Kuznetsov, S. O. et T. Makhalova (2018). On interestingness measures of formal concepts. *Information Sciences 442-443*, 202 – 219.

Mampaey, M., J. Vreeken, et N. Tatti (2012). Summarizing data succinctly with the most informative itemsets. *TKDD 6*(4), 16.

Rosenberg, A. et J. Hirschberg (2007). V-measure : A conditional entropy-based external cluster evaluation measure. In *Proceedings of EMNLP-CoNLL*.

Siebes, A. et R. Kersten (2011). A structure function for transaction data. In *Proceedings of SDM*, pp. 558–569. SIAM.

Smets, K. et J. Vreeken (2012). Slim : Directly mining descriptive patterns. In *Proceedings of SDM*, pp. 236–247. SIAM.

Vinh, N. X., J. Epps, et J. Bailey (2009). Information theoretic measures for clusterings comparison : is a correction for chance necessary ? In *Proceedings of ACM*, pp. 1073–1080. ACM.

Vinh, N. X., J. Epps, et J. Bailey (2010). Information theoretic measures for clusterings comparison : Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research 11*(Oct), 2837–2854.

Vreeken, J., M. Van Leeuwen, et A. Siebes (2011). Krimp : mining itemsets that compress. *Data Mining and Knowledge Discovery 23*(1), 169–214.

Wang, Z. (2012). Entropy on covers. *Data mining and knowledge discovery 24*(1), 288–309.

Summary

In this paper we consider different entropy-based approaches to Pattern Mining. We discuss how entropy on pattern sets can be defined and how it can be incorporated into different stages of mining, from computing candidates to interesting patterns to assessing quality of pattern sets.

Méthodes d'évaluation pour la substitution de vecteurs de mots

Stanislas Morbieu*, François Role*, Mohamed Nadif*

*LIPADE, Université de Paris, 75006 Paris, France
{prénom.nom}@parisdescartes.fr

Résumé. Les vecteurs de mots entraînés sur de grands corpus de textes servent souvent comme entrée pour entraîner des modèles dans plusieurs domaines. Cependant, puisque les données sur lesquelles les vecteurs de mots ont été entraînés diffèrent des données sur lesquelles ils sont utilisés, le problème de vecteurs manquants apparait : certains mots trouvés dans la tâche en aval peuvent être absents du vocabulaire des vecteurs de mots. Divers procédés ont été proposés pour résoudre ce problème en calculant des vecteurs de mots de substitution. Cependant, bien que la qualité de ces substituts puisse avoir un impact important sur le processus d'apprentissage, leur efficacité intrinsèque n'est généralement pas directement mesurée, elle est uniquement estimée à l'aide d'études d'ablation. L'objectif principal de cet article est donc de proposer des méthodes d'évaluation indépendantes de la tâche finale, permettant d'évaluer directement dans quelle mesure les vecteurs de remplacement utilisés sont de bons substituts des vecteurs d'origine.

1 Introduction

Les vecteurs de mots entraînés sur de grands corpus de textes servent souvent comme entrée pour entraîner des modèles dans plusieurs domaines. Étant donné que les données sur lesquelles les vecteurs de mots ont été créés diffèrent de celles sur lesquelles ils sont utilisés, de nombreux mots rencontrés dans la tâche en aval peuvent ne pas avoir de vecteur associé. Ces mots manquants, communément appelés OOVs, peuvent avoir un impact très négatif sur le processus d'apprentissage (Dhingra et al., 2017). Par exemple, près de 61% des mots du jeu de données NG20 [1] sont absents dans le modèle Word2vec (Mikolov et al., 2013b) entraîné sur le corpus Google News. De nombreux praticiens sont quotidiennement confrontés à ce problème dans un large éventail d'applications (classification de texte, étiquetage morpho-syntaxique, systèmes de recommandation, etc.).

Malgré cela, il ne semble pas y avoir d'étude systématique des méthodes utilisées pour traiter ce problème, et de leurs résultats. Les OOVs sont soit simplement ignorés, soit le plus souvent substitués par un vecteur de remplacement construit à l'aide de méthodes dont la qualité n'est pas connue avec précision, car elle est mesurée indirectement à l'aide de tâches aval qui diffèrent d'un papier à l'autre. L'objectif principal de cet article est donc de proposer

1. http://qwone.com/~jason/20Newsgroups/

des méthodes d'évaluation génériques permettant d'évaluer dans quelle mesure les vecteurs de remplacement utilisés sont de bons substituts des vecteurs d'origine.

Pour ce faire, nous proposons de comparer les vecteurs de substitution et les vecteurs originaux correspondants selon deux points de vue. D'une part, nous mesurons la similitude des vecteurs de mots de substitution avec les vecteurs d'origine qu'ils remplacent. D'autre part, nous mesurons le comportement des substituts sur des tâches intrinsèques d'évaluation de vecteurs de mots, à savoir les tâches de similarité et d'analogie. Nous présentons dans une première partie les méthodes permettant de créer des substituts pour les OOVs, puis nous proposons de nouvelles méthodes pour les évaluer, et expérimentons pour étudier la qualité des substituts.

2 Travaux connexes

Parmi les méthodes populaires pour créer des vecteurs de substitution, nous avons :

— **METH1.** Elle consiste à initialiser les vecteurs inconnus avec des zéros ou des valeurs aléatoires.

— **METH2.** Elle prend le centroïde des vecteurs des mots d'une fenêtre de contexte. C'est la méthode la plus utilisée et nous l'utilisons comme base dans nos tâches d'évaluation.

— **METH3.** Elle consiste à calculer des vecteurs de sous-mots (vecteurs de caractères (Ling et al., 2015) ou n-grams (Bojanowski et al., 2017)) et les assembler pour former un vecteur de substitution. L'exemple le plus connu de cette approche est FastText (Bojanowski et al., 2017) où le vecteur pour un mot est donné par la somme des vecteurs de ses n-grams.

— **METH4.** Elle exploite des données auxiliaires (terminologies, thésaurus, bases de données lexicales telles que WordNet, etc.) (Prokhorov et al., 2017; Pilehvar et Collier, 2017).

Par exemple, en traduction automatique, une solution consiste à rechercher des paraphrases des OOVs pour lesquels les traductions sont disponibles, puis agréger les traductions des paraphrases trouvées (Razmara et al., 2013; Chu et Kurohashi, 2016).

La première des méthodes mentionnées ci-dessus (**METH1**) est généralement trop simpliste pour donner de bons résultats alors que la dernière méthode (**METH4**) peut s'avérer efficace mais dépend fortement de la disponibilité des ressources externes. Dans la suite, nous nous concentrons sur les méthodes les plus génériques, à savoir **METH2** et **METH3**, et utilisons ces deux méthodes pour démontrer la possibilité d'utiliser des méthodes d'évaluation indépendantes des tâches en aval.

3 Méthodes d'évaluation

L'idée principale est d'utiliser autant que possible les mêmes méthodes et les mêmes ensembles de données que ceux utilisés pour évaluer les vecteurs de mots "normaux". Plus spécifiquement, nous nous concentrons sur la qualité intrinsèque des vecteurs de substitution au lieu de dépendre de tâches en aval diverses et en constante évolution. Bien sûr, les méthodes utilisées pour évaluer les vecteurs de mots originaux ne peuvent pas être utilisées telles quelles : des ajustements sont nécessaires, ce qui est le sujet de cette section.

Nous proposons de mesurer la qualité d'un vecteur de substitution selon deux axes :

— Un point de vue statique : nous mesurons à quel point les vecteurs de mots générés sont similaires à leurs originaux selon une mesure appropriée.
— Un point de vue dynamique : nous mesurons dans quelle mesure ils se comportent comme l'original sur les tâches de similarité et d'analogie basées sur des jeux de données de test communs et largement disponibles.

Pour évaluer la méthode qui crée un vecteur pour un OOV, nous devons comparer le substitut avec le vecteur d'origine. Pour FastText, le vecteur substitut d'un mot est créé à partir des n-grams qui le composent. Il existe donc un biais lorsque le modèle est formé sur le corpus contenant le mot que nous considérons comme OOV pour nos expériences. Pour atténuer ce biais, lors de la formation du vecteur de mot, nous remplaçons les mots utilisés dans nos tâches par d'autres, de sorte qu'une mise à jour du mot ne mette pas à jour les poids de ses n-grams. Le nouveau mot est généré par une fonction de hachage qui permet d'une part à FastText d'avoir suffisamment de sous-mots, et permet d'autre part à une mise à jour sur un vecteur de mot de ne pas avoir d'incidence sur le vecteur d'un autre.

3.1 Point de vue statique : similarité entre vecteurs de mots et substituts

Pour mesurer la proximité des vecteurs de mots générés et leurs vecteurs connus, nous sélectionnons de manière aléatoire un ensemble de mots et calculons pour chaque mot le cosinus des deux vecteurs de mots (vecteur d'origine et substitution). Pour les valeurs élevées, on peut considérer que les vecteurs peuvent être remplacés par des substituts.

Nous devons également vérifier si une paire de mots similaires dans l'espace des vecteurs originaux se trouve être similaire dans l'espace des substituts. Nous calculons donc, pour un ensemble de paires de mots, la différence entre le score de similarité sur les vecteurs originaux et celui sur les substituts.

3.2 Point de vue dynamique : tâches de similarité et d'analogie

Les tâches de similarité et d'analogie sont souvent utilisées pour évaluer les vecteurs de mots : les scores des jugements humains sont comparés aux calculs effectués sur les vecteurs.

Tâches de similarité. Pour les tâches de similarité, un score de similarité est associé à chaque couple de mots par l'homme, et par le calcul sur les vecteurs de mots. Le coefficient de corrélation de rang de Spearman entre les deux classements est ensuite calculé.

— **Jugement humain.** Les tâches de similarité mesurent différents niveaux de similitude ou de relation. Nous utilisons les tâches d'évaluation suivantes : wordsim (Finkelstein et al., 2001), mturk (Halawi et al., 2012), rg (Rubenstein et Goodenough, 1965) et mc (Miller et Charles, 1991).
— **Vecteurs de mots.** Le cosinus est utilisé comme score de similarité entre deux vecteurs de mots. Pour évaluer la méthode de création de vecteurs des OOVs, nous considérons les scores obtenus avec les vecteurs originaux pour comparaison.
— **Substituts.** Nous comparons les scores obtenus sur les tâches de similarité en utilisant les vecteurs de substitution à ceux obtenus avec les vecteurs originaux, ainsi qu'au jugement humain. La première comparaison mesure la différence entre les substituts et

le vecteur original, et la seconde permet de mesurer le comportement de ces derniers sur la tâche de similarité.

— **Vecteurs originaux et substituts.** En général, les OOVs sont rares, nous considérons donc un mélange de vecteurs originaux et de substituts pour vérifier si les vecteurs de mots inconnus peuvent être remplacés par des substituts.

Tâches d'analogie. Pour les tâches d'analogie, trois mots sont donnés et un quatrième mot doit être trouvé par analogie. Par exemple dans "roi - homme + femme = ?", "reine" doit être trouvé. La proportion de bonnes réponses est ensuite calculée comme score final. Étant donné que la probabilité de trouver le substitut d'un OOV en premier du classement est faible, nous calculons son rang pour chaque question et présentons des statistiques récapitulatives. Pour chaque question $w_1 - w_2 + w_3 = w_4$, on note $\mathbf{E}_c$ le vecteur du mot c, on calcule $v = \mathbf{E}_{w_1} - \mathbf{E}_{w_2} + \mathbf{E}_{w_3}$ et les distances de tous les vecteurs (tous les originaux ainsi que les substituts pour les OOVs) par rapport à v. Le rang du vecteur substitut de w_4 est retenu comme score.

4 Évaluation de deux méthodes de substitution communes

Dans cette partie, nous utilisons text8 [2] comme jeu d'entraînement, et aussi comme corpus sur lequel les substituts sont calculés. Pour l'entraînement, nous utilisons les paramètres par défaut de FastText, et l'implémentation de Gensim (Řehůřek et Sojka, 2010) pour Word2vec. La taille de la fenêtre de contexte pour calculer les centroïdes pour les substituts est de 2.

4.1 Point de vue statique

Pour évaluer dans quelle mesure les substituts sont proches des originaux correspondants (Fig 1), nous sélectionnons au hasard 1000 mots du modèle entraîné sur le corpus et calculons les distances cosinus entre les deux vecteurs (original et substitut). Word2vec donne deux vecteurs de mots, un calculé à partir de la matrice d'entrée et un avec la matrice de sortie du réseau de neurones. Calculer les centroïdes avec la matrice d'entrée entraîne des valeurs de similarité plus élevées que celles des obtenues avec la matrice de sortie. Dans le second cas, les vecteurs se révèlent très différents (faible valeur de similarité). Pour FastText, la méthode du centroïde, et celle de sous-mots intégrée à FastText, entraînent des valeurs de similarité élevées, la méthode du centroïde entraînant une plus grande similarité.

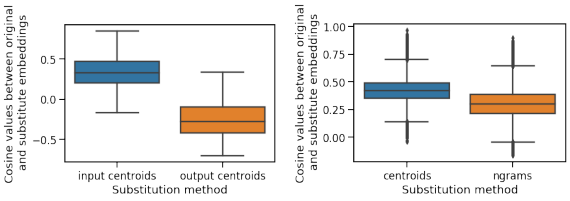


FIG. 1 – *Similarité cosinus entre les vecteurs substituts et originaux pour Word2vec (gauche) et FastText (droite).*

2. http://mattmahoney.net/dc/textdata

Pour évaluer si deux mots similaires d'après leur représentation vectorielle originale sont similaires dans l'espace de substitution, nous calculons les différences entre leurs scores de similarité dans les deux espaces (Fig 2). Prendre les vecteurs de la matrice d'entrée de Word2vec s'avère être meilleur que de prendre ceux de la matrice de sortie. Par conséquent, pour les deux tâches statiques, il est plus approprié de prendre les vecteurs de la matrice d'entrée. Pour FastText, la méthode des sous-mots permet de réduire les différences et est donc préférable à prendre le centroïde, ce qui contraste avec les résultats précédents.

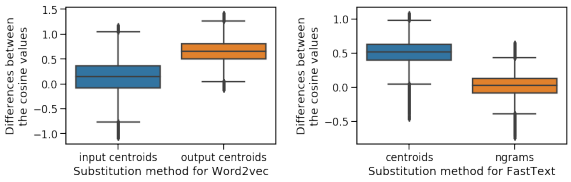


FIG. 2 – *Différence entre la similarité cosinus obtenue avec les vecteurs originaux et ceux obtenus avec les substituts, pour Word2vec (gauche) et FastText (droite).*

4.2 Point de vue "dynamique"

Pour les tâches de similarité, avec Word2vec, prendre les deux vecteurs dans la matrice de sortie donne de meilleurs résultats que dans la matrice d'entrée. Ceci correspond à la situation où deux OOVs sont comparés. Pour FastText, les résultats dépendent fortement du jeu de données d'évaluation. Prendre le substitut d'un mot, et le vecteur original pour l'autre mot de la paire, correspond à la situation plus courante où un OOV est comparé à un mot dont on connait la représentation vectorielle originale. Dans ce cas, prendre les centroïdes de la matrice d'entrée pour les substituts donne des classements similaires à ceux obtenus par jugement humain et ceux obtenus avec les vecteurs originaux. Pour FastText, la méthode des sous-mots se révèle plus performante. Les tests d'analogie de Google (Mikolov et al., 2013a) donnent des résultats très différents sur les 14 ensembles de questions.

5 Conclusion

Nous avons présenté un ensemble de méthodes normalisées centrées sur les mots, permettant d'évaluer de manière comparative la qualité des vecteurs de substitution produits à l'aide de différentes méthodes. Les résultats expérimentaux donnent plusieures indications sur la façon dont certaines méthodes de substitution se comportent et se comparent les unes aux autres.

Références

Bojanowski, P., E. Grave, A. Joulin, et T. Mikolov (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics 5*.

Chu, C. et S. Kurohashi (2016). Paraphrasing out-of-vocabulary words with word embeddings and semantic lexicons for low resource statistical machine translation. In *LREC*.

Dhingra, B., H. Liu, R. Salakhutdinov, et W. W. Cohen (2017). A comparative study of word embeddings for reading comprehension. *CoRR abs/1703.00993*.

Finkelstein, L., E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, et E. Ruppin (2001). Placing search in context : The concept revisited. WWW '01.

Halawi, G., G. Dror, E. Gabrilovich, et Y. Koren (2012). Large-scale learning of word relatedness with constraints. KDD '12. ACM.

Ling, W., C. Dyer, A. W. Black, I. Trancoso, R. Fermandez, S. Amir, L. Marujo, et T. Luis (2015). Finding function in form : Compositional character models for open vocabulary word representation. In *CEMNLP*. Association for Computational Linguistics.

Mikolov, T., K. Chen, G. S. Corrado, et J. Dean (2013a). Efficient estimation of word representations in vector space.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, et J. Dean (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*. Curran Associates, Inc.

Miller, G. A. et W. G. Charles (1991). Contextual correlates of semantic similarity. *Language & Cognitive Processes 6*(1).

Pilehvar, M. T. et N. H. Collier (2017). Inducing embeddings for rare and unseen words by leveraging lexical resources. Association for Computational Linguistics.

Prokhorov, V., M. T. Pilehvar, D. Kartsaklis, P. Lió, et N. Collier (2017). Learning rare word representations using semantic bridging. *arXiv preprint arXiv :1707.07554*.

Razmara, M., M. Siahbani, R. Haffari, et A. Sarkar (2013). Graph propagation for paraphrasing out-of-vocabulary words in statistical machine translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*.

Řehůřek, R. et P. Sojka (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, Malta. ELRA.

Rubenstein, H. et J. B. Goodenough (1965). Contextual correlates of synonymy. *Commun. ACM 8*(10).

Summary

Word embedding vectors trained on large amount of publicly available texts commonly serve as input to learning models in numerous domains. However, since the data on which the word embeddings have been trained differs from the data on which they are used, the so-called "out-of-vocabulary problem" appears: many words encountered in the downstream task may be missing in the word embeddings vocabulary. Various methods have been proposed to tackle this problem by computing substitute word vectors. However, although the quality of these substitutes may have an important impact on the learning process, their intrinsic effectiveness is most of the time not directly measured, being only estimated by ablation studies. The main goal of this paper is therefore to propose task-independent evaluation methods allowing to directly assess to what extent the used replacement vectors are good substitutes for the original ones.

Une approche simultanée pour la réduction de la dimension et la classification d'un graphe attribué

Lazhar Labiod*, Mohamed Nadif*

*Lipade, Université de Paris, 75006 Paris, France
<prénom.nom>@parisdescartes.fr,

Résumé. Nous présentons une nouvelle approche pour la réduction de la dimension et le *clustering* de graphes attribués. Le modèle proposé exploite simultanément les informations du contenu (description des noeuds) et de la structure (relations entre les noeuds). Il tire ainsi parti du renforcement mutuel entre les tâches de réduction de la dimensionalité et de *clustering* d'un graphe attribué.

1 Introduction

Ces dernières années, les réseaux attribués sont utilisés pour modéliser une grande variété de réseaux du monde réel (Qi et al., 2012; Chang et al., 2015; Pan et al., 2018; Wang et al., 2017; Guo et al., 2019), tels que les réseaux universitaires et les systèmes de santé, dans lesquels les liens entre les nœuds et les attributs / caractéristiques sont disponibles pour l'analyse. La réduction de dimension et la classification non supervisée de graphes attribués ont suscité beaucoup d'attention dans de nombreuses applications, notamment les réseaux sociaux, les réseaux de citations universitaires et les réseaux d'interaction protéine-protéine.

En raison de l'hétérogénéité des deux sources d'information, il est difficile d'appliquer directement les algorithmes de réduction de dimension existants à un graphe attribué. Dans un graphe attribué, les nœuds avec la même étiquette ont généralement des relations sociales et des attributs similaires. Les étiquettes sont donc fortement influencées par les liens de nœuds et les attributs associés aux nœuds et intrinsèquement corrélées à la structure du graphe et aux informations fournies par les d'attributs.

Bien que le *clustering* des grahes attribués ait été largement appliqué dans la pratique, il est facile d'obtenir des résultats non profitables du fait des inconvénients suivants : (1) risque élevé de déviation de la solution de réduction de dimension continue par rapport à la bonne solution de classification, (2) perte d'informations entre les deux différentes étapes, c'est-à-dire la génération d'une nouvelle représentation réduite continue (*data embedding*), et la solution discrète de classification. Pour surmonter ces difficultés, nous proposons une approche permettant d'apprendre simultanément une représentation réduite et la classification d'un graphe attribué apprenant directement les étiquettes continues et la classification discrète dans un cadre unifié. Le principal défi consiste à intégrer les informations des liens et des attributs des noeuds pour un apprentissage simultané de la représentation et le partitionnement des noeuds.

2 Méthode proposée

Dans cette section, nous développons la méthode proposée SAGEC (Simultaneous Attribute Graph Embedding and Clustering). Nous présentons d'abord la formulation, puis développons un algorithme d'optimisation efficace.

Formellement, le graphe peut être représenté par deux types d'informations, l'information de contenu $\mathbf{X} \in \mathbb{R}^{n \times d}$ et l'information de structure $\mathbf{A} \in \mathbb{R}^{n \times n}$, où $\mathbf{A}$ est une matrice de contiguïté de terme général $a_{ij} = 1$ si $e_{ij} \in E$ sinon 0. Nous considérons que chaque nœud est un voisin de lui-même, puis nous définissons $a_{ii} = 1$ pour tous les nœuds. Nous modélisons la proximité des nœuds par une matrice de transition : $\mathbf{W} = \mathbf{D}^{-1}\mathbf{A}$, où $\mathbf{D}$ est la matrice des degrés obetnue de $\mathbf{A}$ avec $d_{ii} = \sum_{i'} a_{i'i}$.

Afin d'exploiter des informations supplémentaires sur la similarité des nœuds de $\mathbf{X}$, nous avons prétraité le jeu de données $\mathbf{X}$ pour produire un graphe de similarité $\mathbf{W_x}$; nous avons construit un graphe K-plus proches voisins KNN ($K = 15$) en utilisant le noyau gaussien, la distance L_2 et la largeur du voisinage $\sigma = 1$. Nous combinons ces deux informations supplémentaires (les nœuds proches des informations de contenu $\mathbf{X}$ et de la structure $\mathbf{W}$) dans $\mathbf{S}$ par $\mathbf{S} = \frac{1}{2}(\mathbf{W} + \mathbf{W_x})$. Nous obtenons ensuite une nouvelle représentation de données $\mathbf{M}$, qui est une intégration multiplicative des informations $\mathbf{W}$ et $\mathbf{X}$ en remplaçant chaque nœud par le centroïde de leur voisinage (barycentre) ; $\mathbf{m}_{ij} = \sum_k \mathbf{w}_{ik}\mathbf{x}_{kj}, \forall i, j$, en formulation matricielle, on écrit $\mathbf{M} = \mathbf{WX}$.

2.1 Modèle

Soit k le nombre de clusters et le nombre de composants de la nouvelle représentation réduite des données. La méthode SAGEC, qui peut être considérée comme une procédure visant à apprendre une nouvelle représentation réduite qui soit la plus informative sur la structure de classifcation, elle est définie comme le problème de minimisation suivant :

$$\min_{\mathbf{B},\mathbf{Z},\mathbf{Q},\mathbf{G}} \left\|\mathbf{M} - \mathbf{B}\mathbf{Q}^\top\right\|^2 + \lambda\left\|\mathbf{S} - \mathbf{G}\mathbf{Z}\mathbf{B}^T\right\|^2, \quad \mathbf{B}^\top\mathbf{B} = \mathbf{I}, \mathbf{Z}^\top\mathbf{Z} = \mathbf{I}, \mathbf{G} \in \{0,1\}^{n \times k} \tag{1}$$

où $\mathbf{Z} \in \mathbb{R}^{k \times k}$ est une matrice de rotation qui satisfait les conditions d'orthogonalité $\mathbf{Z}^\top\mathbf{Z} = \mathbf{Z}\mathbf{Z}^\top = \mathbf{I}$. La matrice non négative $\mathbf{G} = (g_{ij})$ de taille $(n \times k)$ est une matrice d'appartenance à une classe, $\mathbf{B} = (b_{ij})$ de taille $(n \times k)$ est la matrice de représentation réduite des noeuds et $\mathbf{Z} = (z_{ij})$ de taile $(k \times k)$ est une matrice de rotation orthonormale qui mappe le plus étroitement $\mathbf{B}$ à $\mathbf{G} \in \{0,1\}^{n \times k}$. $\mathbf{Q} \in \mathbb{R}^{n \times k}$ est la matrice de représentation réduite des attributs. Enfin, le paramètre λ est une valeur non négative et peut être considéré comme un paramètre de régularisation. Le deuxième terme dans (1) peut être décomposé de la manière suivante :

$$\min_{\mathbf{Z}} \left\|\mathbf{S} - \mathbf{G}\mathbf{Z}\mathbf{B}^T\right\|^2 \Leftrightarrow \min_{\mathbf{Z}} \left\|\mathbf{S} - \mathbf{B}\mathbf{B}^T\mathbf{S}\right\|^2 + \left\|\mathbf{S}\mathbf{B} - \mathbf{G}\mathbf{Z}\right\|^2 \tag{2}$$

2.2 Algorithme SAGEC

Les principales étapes de l'algorithme SAGEC sont décrites dans *Algorithm 1*. Par une optimisation alternée, la convergence de SAGEC est garantie et dépend de l'initialisation pour atteindre un optimum local.

Algorithm 1 : Algorithme SAGEC

Input : matrice du graphe $\mathbf{W}$ et la matrice du contenu $\mathbf{X}$;
Initialize : $\mathbf{B}$, $\mathbf{Q}$ et $\mathbf{Z}$ avec des matrices orthonormales ;
repeat
(a) - Calcul de $\mathbf{G}$ par optimisation du terme 2 dans (2)
(b) - Calcul de $\mathbf{B}$ par optimisation de (1)
(c) - Calcul de $\mathbf{Q}$ par optimisation du terme 1 dans (1)
(d) - Calcul de $\mathbf{Z}$ par optimisation du terme 2 dans (2)
until convergence
Output : $\mathbf{G}$: matrice de classification, $\mathbf{Z}$: matrice de rotation, $\mathbf{B}$: représentation réduite des noeuds et $\mathbf{Q}$: représentation réduite des attributs.

3 Expériences numériques

Nous illustrons les performances de SAGEC en terme de clustering, montrant l'impact de la combinaison de differentes sources d'information ($\mathbf{W}$ et $\mathbf{X}$) et du renforcement mutuel entre réduction de dimension et clustering. Nous utilisons pour cela trois jeux de données de graphes attribués, Cora, Citeseer et Wiki. Les caractéristiques des jeux de données utilisés sont résumées dans le tableau 1. Pour notre méthode, on définit le paramètre de régularisation $\lambda \in \{10^6, 10^3, 10^1, 10^0, 10^1, 10^3\}$ et on choisit la meilleure valeur comme résultat final.

TAB. 1 – *Description des données (# désigne la cardinalité)*

Données	# Nodes	# Attributes	# Edges	#Classes
Cora	2708	1433	5294	7
Citeseer	3312	3703	4732	6
Wiki	2405	4973	17981	19

On compare SAGEC avec différents algorithmes appliqués à la matrice de graphe W, à la matrice de contenu $\mathbf{X}$, ou combinant les deux sources d'information ; pour plus de détails sur les méthodes comparées voir (Pan et al., 2018). Nous réalisons 50 initialisations aléatoires, et calculons les mesures Accuracy (Acc) qui représente le taux des bien classés, l'indice de Rand ajusté (ARI) et l'information mutuelle normalisée (NMI) pour évaluer la qualité de la partition au regard de la partition connue. Les résultats obtenus sont presentés dans la table 2. Nous pouvons remarquer qu'en termes de ces mesures, SAGEC permet de donner de bons résultats et donc de trouver le meilleur compromis exploitant toutes les informations issues de $\mathbf{W}$ et $\mathbf{X}$.

4 Conclusion

Dans cet article, nous avons proposé un nouveau cadre de décomposition matricielle pour l'apprentissage simultané d'une représentation réduite continue et d'une classification discrète des noeuds à partir d'un graphe atribué. La méthode proposée SAGEC permet de trouver le meilleur compromis en considérant toutes les informations ($\mathbf{W}$, $\mathbf{X}$) et donne ainsi de bonnes performances de classification non supervisée.

TAB. 2 – *Performances en terme de* clustering *(Acc % , NMI % et ARI %) sur Cora, Citeseer et Wiki.*

Méthodes	Données								
	Cora			Citeseer			Wiki		
	Acc	NMI	ARI	Acc	NMI	ARI	Acc	NMI	ARI
K-means	49.22	32.10	22.96	54.01	30.54	27.86	41.72	44.02	15.07
Spectral	36.72	12.67	03.11	23.89	05.57	01.00	22.04	18.17	01.46
GraphEncoder	32.49	10.93	00.55	22.52	03.30	01.00	20.67	12.07	0.49
DeepWalk	48.40	32.70	24.27	33.65	08.78	09.22	38.46	32.38	17.03
DNGR	41.91	31.84	14.22	32.59	18.02	04.29	37.58	35.85	17.97
ARGE	64.0	44.9	35.2	57.3	35.0	34.1	47.34	47.02	28.16
ARVGE	63.8	45.0	37.74	54.4	26.1	24.5	46.45	47.8	29.65
SAGEC	**67.38**	**47.14**	**39.88**	**66.77**	**40.60**	**41.78**	**49.57**	**48.30**	**33.14**

Références

Chang, S., W. Han, J. Tang, G.-J. Qi, C. C. Aggarwal, et T. S. Huang (2015). Heterogeneous network embedding via deep architectures. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pp. 119–128.

Guo, T., S. Pan, X. Zhu, et C. Zhang (2019). Cfond : Consensus factorization for co-clustering networked data. *IEEE Transactions on Knowledge Data Engineering 31*(04), 706–719.

Pan, S., R. Hu, G. Long, J. Jiang, L. Yao, et C. Zhang (2018). Adversarially regularized graph autoencoder for graph embedding. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, pp. 2609–2615.

Qi, G., C. C. Aggarwal, Q. Tian, H. Ji, et T. S. Huang (2012). Exploring context and content links in social media : A latent space method. *IEEE Trans. Pattern Anal. Mach. Intell. 34*(5), 850–862.

Wang, C., S. Pan, G. Long, X. Zhu, et J. Jiang (2017). Mgae : Marginalized graph autoencoder for graph clustering. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, pp. 889–898.

Summary

We present a novel way to deal simultaneously with the attributed network embedding and clustering. Our model exploits both content and structure information and therefore takes advantage of the mutual reinforcement between attributed network embedding and clustering tasks. It is able to better approximate the relaxed continuous embedding solution by the true discrete clustering one.

Streaming constrained binary logistic regression with online standardized data

Benoît Lalloué[*,**], Jean-Marie Monnez[*,**], Eliane Albuisson[***,****,♯]

[*] Université de Lorraine, CNRS, Inria [1], IECL [2], Nancy, France
[**] CHRU Nancy, INSERM, Université de Lorraine, CIC [3], Plurithématique, Nancy, France
[***] Université de Lorraine, CNRS, IECL [4], Nancy, France
[****] BIOBASE, Pôle S2R, CHRU de Nancy, Vandoeuvre-lès-Nancy, France
[♯] Faculté de Médecine, InSciDenS, Vandoeuvre-lès-Nancy, France
benoit.lalloue@univ-lorraine.fr
jean-marie.monnez@univ-lorraine.fr
eliane.albuisson@univ-lorraine.fr

Abstract. We study a stochastic gradient algorithm for performing online a constrained binary logistic regression in the case of streaming or massive data. Assuming that observed data are realizations of a random vector, these data are standardized online in particular to avoid a numerical explosion or when a shrinkage method such as LASSO is used. We prove the almost sure convergence of a variable step-size constrained stochastic gradient process with averaging when a varying number of new data is introduced at each step. Several stochastic approximation processes with raw data or online standardized data are compared on observed or simulated datasets. The best results are obtained by processes with online standardized data.

1 Introduction

One type of method to analyse streaming or massive data is online learning which proceeds in successive steps, the results of the analysis being updated at each step taking into account a batch of new data. Recursive stochastic algorithms can be used for observations arriving sequentially to estimate for example parameters of a linear regression model (Duarte et al., 2018) or principal components of a factorial analysis (Monnez and Skiredj, 2018) or centres of classes in non-hierarchical clustering (Cardot et al., 2012), whose estimations are updated by each new arriving data batch. In this context, it is not necessary to store the data and, due to the relative simplicity of the computation involved, much more data than with classic methods can be taken into account during the same duration of time. For massive datasets, recursive algorithms can be used by randomly drawing at each step a data batch from the dataset.

Why use online standardized data (each continuous variable is standardized with respect to the estimations at the current step of its expectation and of its standard deviation computed online) and a constrained process? First *to avoid a numerical explosion* as it is studied in Duarte et al. (2018) in the case of sequential multidimensional linear regression. The experiments conducted have shown better performance of processes with online standardized data compared to

those with raw data. Second, *when using a shrinkage method such as LASSO or ridge*, we have first to standardize the explanatory variables. In the case of a data stream, when the mathematical expectation and the variance of each variable are a priori unknown, these variables can be standardized online and a process of the same type can be used but with a projection at each step on the convex set defined by the constraint on the parameters of the regression function. More generally *this type of process can be used for any convex set,* for example if it is imposed that the parameters associated to the explanatory variables are positive. Third we can consider the case where a logistic model with standardized explanatory variables is defined and *where explanatory variables have an expectation and a variance that may depend on time or on the values of controlled variables* according to a specific model; this assumes that we can estimate online the expectation and the variance of these variables.

A suitable choice of step-size is often crucial for obtaining good performance of a stochastic gradient process. If the step-size is too small, the convergence will be slower. Conversely, if it is too large, a numerical explosion may occur during the first iterations. We use here *an averaged stochastic gradient process, with a piecewise constant step-size* as suggested in Bach (2014) in order that the step-size does not decrease too quickly and reduces the speed of convergence.

2 Study of a stochastic gradient process

Suppose that data are realizations of a random vector $(R^1, ..., R^p, S)$ in $\mathbb{R}^p \times \{0, 1\}$.

Let A' be the transpose of a matrix A. Let R be the random column vector $\left(R^1 \ ... \ R^p \ 1\right)'$, $m = \left(E\left[R^1\right] \ ... \ E\left[R^p\right] \ 0\right)'$, $R^c = R - m$, r^c a realization of R^c, σ^k the standard deviation of R^k, $k = 1, ..., p$, Γ the diagonal $(p+1, p+1)$ matrix with diagonal elements $\frac{1}{\sigma^1}, ..., \frac{1}{\sigma^p}, 1$ (taking by convention $\sigma^k = 1$ for a discrete variable), $Z = \Gamma R^c$, whose continuous components are standardized, $z = \Gamma r^c$ a realization of Z, $\theta = \left(\theta^1 \ ... \ \theta^P \theta^{P+1}\right)'$ a column vector of real parameters.

Consider the logistic model with standardized covariates:

$$P\left(S = s \mid R = r\right) = f(s; z, \theta) = \left(\frac{e^{z'\theta}}{1 + e^{z'\theta}}\right)^s \left(\frac{1}{1 + e^{z'\theta}}\right)^{1-s} = \frac{e^{z'\theta s}}{1 + e^{z'\theta}}.$$

$E\left[S \mid R\right] = h\left(Z'\theta\right)$ with $h(u) = \frac{e^u}{1+e^u} = \frac{1}{1+e^{-u}}$.

Define the loss function -ln $f\left(s; z, x\right) = -z'xs + \ln\left(1 + e^{z'x}\right)$. The cost function

$$F(x) = -E\left[\ln f\left(S; Z, x\right)\right] = E\left[-Z'xS + \ln\left(1 + e^{Z'x}\right)\right]$$

has θ for unique minimizer since F is a convex function with positive hessian. θ is the unique solution of:

$$F'(x) = E\left[-ZS + \frac{Ze^{Z'x}}{1 + e^{Z'x}}\right] = E\left[Z\left(h\left(Z'x\right) - S\right)\right] = 0.$$

Let $\left((R_n^1, ..., R_n^p, S_n), n \geqslant 1\right)$ be an i.i.d. sample of $(R^1, ..., R^p, S)$, for $n \geqslant 1$, $R_n = \left(R_n^1 \ ... \ R_n^p \ 1\right)'$, for $k = 1, ..., p$, $\overline{R}_n^k$ the mean of the sample $\left(R_1^k, ..., R_n^k\right)$ of R^k and $\left(V_n^k\right)^2 =$

$\frac{1}{n}\sum_{i=1}^{n}\left(R_i^k - \overline{R_n^k}\right)^2$ its variance, both recursively computed, $\overline{R}_n = \left(\overline{R_n^1} \ldots \overline{R_n^p}\ 0\right)'$ and Γ_n the diagonal $(p+1, p+1)$ matrix with diagonal elements $\frac{1}{\sqrt{\frac{n}{n-1}V_n^1}}, ..., \frac{1}{\sqrt{\frac{n}{n-1}V_n^p}}, 1$.

Suppose that m_n observations (R_i, S_i) are taken into account at step n of the defined process. Let $\mu_n = \sum_{i=1}^{n} m_i, I_n = \{\mu_{n-1}+1, ..., \mu_n\}$, $\widehat{R}_n = \overline{R}_{\mu_n}$, $\widehat{\Gamma}_n = \Gamma_{\mu_n}$ and for $j \in I_n, \widetilde{Z}_j = \widehat{\Gamma}_{n-1}\left(R_j - \widehat{R}_{n-1}\right)$: for $k = 1, ..., p$, each component R_j^k of R_j is pseudo-standardized with respect to the empirical mean $\widehat{R}_{n-1}^k$ and to the empirical estimation of σ^k, $\sqrt{\frac{n}{n-1}}V_{\mu_{n-1}}^k$.

Suppose that θ is constrained to belong to a convex subset K of $\mathbb{R}^{p+1}$. Let Π be the projection operator on K. Recursively define the stochastic approximation processes (X_n) of the Robbins-Monro type (Robbins and Monro, 1951) and $\left(\overline{X}_n\right)$ in $\mathbb{R}^{p+1}$:

$$X_{n+1} = \Pi\left(X_n - a_n \frac{1}{m_n}\sum_{j\in I_n}\widetilde{Z}_j\left(h\left(\widetilde{Z}_j' X_n\right) - S_j\right)\right),\ \overline{X}_{n+1} = \frac{1}{n+1}\sum_{1}^{n+1} X_i.$$

Theorem 1 *Suppose there is no affine relation between the components of R, the moments of order 4 of R exist and* $a_n > 0$, $\sum_{n=1}^{\infty} a_n = \infty$, $\sum_{n=1}^{\infty}\frac{a_n}{\sqrt{n}} < \infty$, $\sum_{n=1}^{\infty} a_n^2 < \infty$. *Then* (X_n) *and* $\left(\overline{X}_n\right)$ *converge to* θ *a.s.*

The proof is in Lalloué et al. (2019b).

3 Experiments

Stochastic approximation processes were compared, including classic stochastic gradient descent (SGD) with a variable step-size, averaged stochastic gradient descent (ASGD) with a piecewise constant step-size with level sizes 50, 100 or 200, and the same processes but with online standardization of the data (Section 2). For these 8 processes, 3 variants with 1, 10 or 100 new observations per step were tested. Therefore 24 processes are studied. For processes with a variable step-size, we have defined $a_n = \frac{c}{(b+n)^\alpha}$, for those with a piecewise constant step-size, $a_n = \frac{c}{(b+\lfloor\frac{n}{\tau}\rfloor)^\alpha}$ where $\lfloor.\rfloor$ denotes the integer part and τ is the size of the levels. We set $\alpha = 2/3$, $b = 1$, $c = 1$. All processes were initialized with $X_1 = 0$.

We used as "gold standard" the vector of coefficients θ^c obtained by classic logistic regression (using R's glm function). Let $\hat{\theta}_{n+1}$ be the estimated vector obtained by a tested process after n iterations. The cosine of the angle between θ^c and $\hat{\theta}_{n+1}$ was used as a convergence criterion: $cos(\theta^c, \hat{\theta}_{n+1}) = \frac{\theta^{c\prime}\hat{\theta}_{n+1}}{\|\theta^c\|\|\hat{\theta}_{n+1}\|}$.

The processes were tested on five datasets available on the Internet (Twonorm, Ringnorm, Quantum, Adult, EEG) and the HOSPHF30D dataset derived from the EPHESUS study (Pitt et al., 2003), all already used to test the performance of processes with online standardized data in the case of online linear regression (Duarte et al., 2018). Twonorm and Ringnorm contain

simulated data. Adult, EEG and HOSPHF30D contain observed data with outliers, variables of different types and scales, unlike Quantum.

At each step of a process a data batch is randomly drawn from the dataset. All processes were applied on all datasets for a fixed number of observations used and for a fixed processing time (the cumulative time to compute the process updates, excluding operations such as data sampling, data management, formatting and recording of results). As an example, for a processing time of 60s (Figure 1) all tested processes using raw observed data, except Quantum, had a numerical explosion. Abbreviations used in Figure 1 are: C for classic SGD or A for ASGD, R for raw data or S for online standardized data, V for variable step-size or P for piecewise constant step-size; for instance, AR1P50 is the averaged process with raw data, 1 observation per step, piecewise constant step-size with level size 50, CS1V is the classic process with online standardized data, 1 observation per step and variable step-size.

Process	Twonorm	Ringnorm	Quantum	Adult	EEG	HOSPHF30D	Mean rank
CR1V	0.9999	0.9999	0.9709	EXPL	EXPL	EXPL	20.8
CR10V	1.0000	1.0000	0.9683	EXPL	EXPL	EXPL	21.8
CR100V	1.0000	1.0000	0.9659	EXPL	EXPL	EXPL	22.5
AR1P50	1.0000	1.0000	0.9978	EXPL	EXPL	EXPL	19.3
AR10P50	1.0000	1.0000	0.9960	EXPL	EXPL	EXPL	18.3
AR100P50	1.0000	1.0000	0.9948	EXPL	EXPL	EXPL	20.0
AR1P100	1.0000	1.0000	0.9991	EXPL	EXPL	EXPL	18.0
AR10P100	1.0000	1.0000	0.9972	EXPL	EXPL	EXPL	17.5
AR100P100	1.0000	1.0000	0.9959	EXPL	EXPL	EXPL	19.0
AR1P200	1.0000	1.0000	0.9999	EXPL	EXPL	EXPL	16.8
AR10P200	1.0000	1.0000	0.9981	EXPL	EXPL	EXPL	16.5
AR100P200	1.0000	1.0000	0.9970	EXPL	EXPL	EXPL	18.2
CS1V	0.9997	0.9998	0.9987	0.9979	0.9988	0.9898	17.5
CS10V	0.9996	1.0000	0.9989	0.9968	0.9994	0.9932	15.8
CS100V	0.9994	1.0000	0.9992	0.9953	0.9993	0.9840	15.3
AS1P50	0.9999	1.0000	0.9959	0.9964	0.9993	0.9854	17.2
AS10P50	1.0000	1.0000	0.9999	0.9998	0.9997	0.9986	8.2
AS100P50	1.0000	1.0000	0.9999	0.9999	1.0000	0.9999	6.5
AS1P100	0.9999	1.0000	0.9948	0.9888	0.9992	0.9841	19.2
AS10P100	1.0000	1.0000	0.9999	0.9998	0.9996	0.9987	8.8
AS100P100	1.0000	1.0000	0.9999	0.9999	1.0000	0.9999	6.7
AS1P200	0.9999	0.9999	0.9934	0.9823	0.9987	0.9812	19.8
AS10P200	1.0000	1.0000	0.9999	0.9996	0.9996	0.9986	8.2
AS100P200	1.0000	1.0000	0.9999	1.0000	1.0000	0.9999	4.8

EXPL: numerical explosion.
Process type: C for classic SGD, A for ASGD. *Data type:* R for raw data, S for online standardized data.
First number: number of new observations at each step.
Step-size: V for variable, P for piecewise constant (*second number* is the levels size).

FIG. 1 – *Cosines for 1 minute of processing time*

For each dataset and at each recording point (see below), processes were ranked from the best (highest cosine) to the worst (lowest cosine). The mean rank over all datasets was used to compare the processes at a given recorded point and globally. Over all datasets, the processes with the best results after 60s are averaged processes with online standardization and

piecewise constant step-sizes, the best one with levels of size 200 and 100 new observations per step (AS100P200).

As in Duarte et al. (2018), the values of the criterion for each process were recorded every N observations used from N to $100 \times N$, N being the number of observations in a dataset, and every second of processing time from 1 to 120s. As an example, when studying the evolution of the rankings with the processing time, two groups of processes appear clearly from the beginning and remain during all the studied period. The group with the worst rankings (at the top in Figure 2) contains all processes using raw data, all processes using only one new observation at each step, and all "classic" processes. The group with the best rankings (at the bottom in Figure 2) contains all averaged processes with online standardization, piecewise constant step-sizes, and using 10 or 100 new observations per step, the best one with levels of size 200 and 100 new observations per step. Other results can be found in Lalloué et al. (2019b).

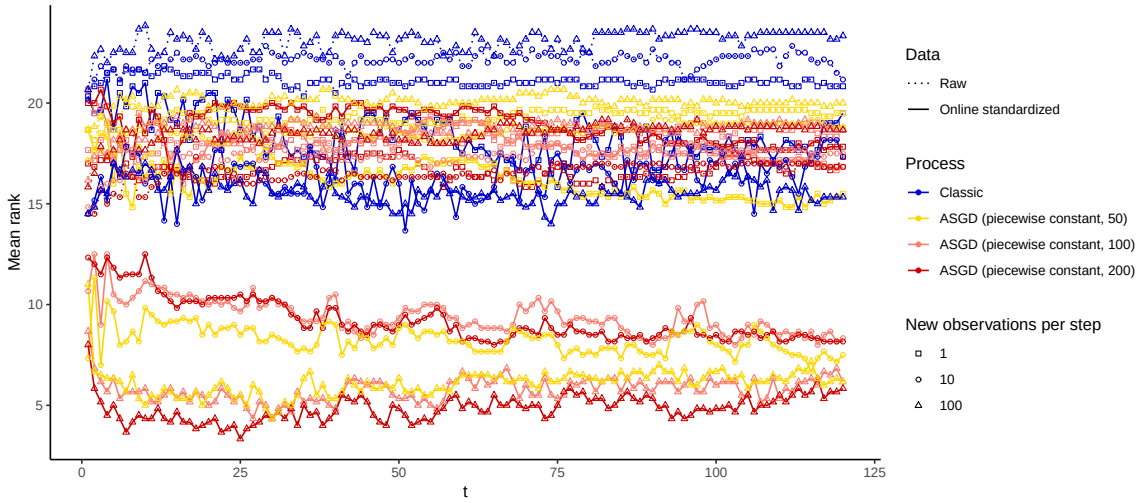


FIG. 2 – *Evolution with the processing time*

Conclusion

We have studied an averaged constrained stochastic gradient algorithm for performing online a constrained binary logistic regression. We have proposed to use an online standardization of the data to avoid a numerical explosion, or when a shrinkage method (such as LASSO) is used, or even when expectations or variances of explanatory variables change (varying with time or depending on the values of controlled variables) and can be estimated online. We have proposed to use a decreasing piecewise constant step-size in order that it does not decrease too quickly and consequently reduces the speed of convergence of the process. We have made experiments on observed and simulated datasets. The results confirm the validity of the choices

made: online standardization of the data, averaged process and piecewise constant step-size. This algorithm is used for scoring online heart failure (Lalloué et al., 2019a).

Acknowledgement

Results incorporated in this article received funding from the investments for the Future Program under grant agreement No ANR-15-RHU-0004.

References

Bach, F. (2014). Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression. *Journal of Machine Learning Research 15*, 595–627.

Cardot, H., P. Cénac, and J.-M. Monnez (2012). A fast and recursive algorithm for clustering large datasets with k-medians. *Computational Statistics & Data Analysis 56*(6), 1434–1449.

Duarte, K., J.-M. Monnez, and E. Albuisson (2018). Sequential linear regression with online standardized data. *PLOS ONE 13*(1), e0191186.

Lalloué, B., J.-M. Monnez, and E. Albuisson (2019a). Actualisation en ligne d'un score d'ensemble. In *51e Journées de Statistique*, Nancy, France. Société Française de Statistique. *hal-02152352*.

Lalloué, B., J.-M. Monnez, and E. Albuisson (2019b). Streaming constrained binary logistic regression with online standardized data. Application to scoring heart failure. *hal-02156324*.

Monnez, J.-M. and A. Skiredj (2018). Convergence of a normed eigenvector stochastic approximation process and application to online principal component analysis of a data stream. *hal-01844419*.

Pitt, B., W. Remme, F. Zannad, J. Neaton, F. Martinez, B. Roniker, R. Bittman, S. Hurley, J. Kleiman, and M. Gatlin (2003). Eplerenone, a selective Aldosterone blocker, in patients with left ventricular dysfunction after myocardial infarction. *New England Journal of Medicine 348*(14), 1309–1321.

Robbins, H. and S. Monro (1951). A stochastic approximation method. *The Annals of Mathematical Statistics 22*(3), 400–407.

Résumé

Nous étudions un algorithme de gradient stochastique pour réaliser une régression logistique sous contraintes dans le cas de données massives ou en ligne. En supposant que les données observées sont les réalisations d'un vecteur aléatoire, ces données sont standardisées en ligne pour éviter une explosion numérique ou lorsqu'une méthode de pénalisation telle que LASSO est utilisée. Nous démontrons la convergence presque sûre d'un processus de gradient stochastique moyenné à pas variable lorsqu'un nombre variable de nouvelles données sont introduites à chaque étape. Vingt-quatre processus d'approximation stochastique avec des données brutes ou standardisées en ligne sont comparés sur des données réelles ou simulées. Les meilleurs résultats sont obtenus pour des processus avec données standardisées.

Weighted consensus clustering for multiblock data

Ndèye Niang*, Mory Ouattara**

*Statistique Appliquée, CNAM 292, rue Saint Martin, 75141 Paris Cedex 03, France
**Laboratoire de Mathématique Informatique,
UFR-SFA, Université Nangui Abrogoua, 21 BP 1288 Abidjan 21,
ouattaramory.sfa@univ-na.ci

Résumé. Nous nous intéressons au problème de classification d'individus décrits par plusieurs variables structurées en blocs homogènes. Nous le formulons comme la recherche de consensus de partitions et nous proposons une méthode de consensus pondéré de partitions basée sur le coefficient RV de corrélation vectorielle. L'idée principale de la méthode proposée consiste à agréger les partitions initiales séparées obtenues à partir de chaque bloc en une partition globale la plus similaire à ces partitions initiales au sens du coefficient RV. La méthode proposée est comparée à la méthode CSPA (Cluster based Similarity Partitioning Algorithm) et une méthode de consensus pondéré basée sur la factorisation non négative de matrices.

1 Introduction

The general problem addressed in this paper is clustering individuals described by variables which are divided in several homogeneous and meaningful blocks. Since blocks are assumed to be homogeneous, preserving this block homogeneity would help to exhibit the underlying structure of individuals. Thus in a first level, the individuals are clustered according to each block separately and the resulting partitions (called contributory partitions) are aggregated into a consensus partition in a second step. Therefore, the issue of this paper is reformulated as a problem of consensus of partitions. The choice of the first step clustering method is not addressed here. We only focus on the aggregation of the obtained partitions.

Clustering multiblock data has been addressed by several consensus methods. A survey on these methods can be found in Day (1994). The main idea of consensus methods is to agglomerate the separate partitions obtained from each block into a global partition that must be as similar as possible to the contributory partitions according to some index, eg. the Rand index. Other more recent methods in the ensemble clustering approach (Strehl, 2003) have also addressed the consensus clustering problem. The contributory partitions are seen as categorical variables and then are associated with their indicator matrices and connectivity matrices whose entries are 1 if two individuals are in the same cluster and 0 otherwise. (Strehl, 2003) proposed CSPA (Cluster based Similarity Partitioning Algorithm) which consists of re-clustering the individuals using a so-called association matrix considered as a similarity matrix. The association matrix is obtained by simply averaging the connectivity matrices. Thus its entries are defined as the fraction of partitions in which two individuals are in the same cluster. In this method,

implicitly, all contributory partitions are treated equally, despite the facts that (1) partitions could differ significantly and (2) subsets of partitions could be highly correlated. Therefore as pointed out in (Tao, 2008), this approach essentially based on a simple averaging process is inadequate. CSPA yields unstable and biased results. Tao (2008) proposed a weighted consensus clustering method (denoted WNMF) that do not have the CSPA drawbacks. It is based on a non-negative matrix factorization (NMF) and the optimization of a quadratic function to obtain the consensus partition and the weights respectively.

We propose a weighted consensus clustering method based on the RV correlation coefficient (Robert, 1976) between the connectivity matrices to find an unique partition of individuals from contributory partitions. A so-called compromise matrix, weighted average of connectivity matrices, is used to re-cluster the individuals with a classical hierarchical algorithm. In the next section we detail our method which can be seen as the combination of WNMF in the sense that we obtain a weighted consensus matrix and of CSPA as we cluster this later compromise matrix.

2 Weighted consensus clustering

2.1 RV coefficient

We consider P variables divided into T blocks. In each block N outcomes of p_t variables are available. The RV index Robert (1976) is a measure of the relationship between two sets of variables X_t and $X_{t'}$. Prior to the analysis, data matrices are usually centered and normalized to remove scale effects. X_t is associated with the matrix of scalar products $W_t = X_t X_t'$, where X_t' denotes the transpose of X_t. This step is necessary to have square matrices of same dimension N and then allows to have X_t matrices with different number of columns. The RV between X_t and $X_{t'}$ derived from Hilbert-Schmidt's scalar product, is defined as follows :

$$RV(W_t, W_t') = \frac{trace(W_t W_{t'})}{\sqrt{trace(W_t^2)trace(W_{t'}^2)}}$$

RV index is non-negative and scaled between 0 and 1. The closer the RV index is to 1, the more similar the matrices X_t and $X_{t'}$ are.

2.2 RV based consensus clustering

Let P_t be the t^{th} contributory partition obtained from the clustering of the t^{th} block of variables into K_t clusters (which may differ from one partition P_t to another). P^* denotes the final partition, consensus of the T partitions P_t. We consider $X_t,\ t = 1, \ldots, T$ the indicator matrix of K_t dummy variables related to the categorical variable associated with the t^{th} contributory partition. Each of these matrices is associated with a matrix $W_t = X_t X_t'$ which is a connectivity matrix whose entries are : $W_t(i,j) = \begin{cases} 1 & if \quad P_t(i) = P_t(j) \\ 0 & otherwise \end{cases}$

The first step of the proposed weighted consensus clustering method is to find a so-called compromise matrix W^*, weighted average of the W_t which has to be the most similar to the W_t (representing the contributory partitions). Our proposition is to use the RV index as a

measure of this similarity between data tables. Therefore the criterion to optimize to get the compromise matrix is the following one :

$$\underset{\alpha_1,\ldots,\alpha_T}{max} \sum_{t=1}^{T} RV(W_t, W^*) = \underset{\alpha_1,\ldots,\alpha_T}{max} \sum_{t=1}^{T} RV(W_t, \sum_{t=1}^{T} \alpha_t W_t)$$

We are looking for the best linear combination of the matrices W_t that maximizes its vectorial correlation RV with the W_t matrices. As in principal component analysis, the solution is obtained through weights equal to the coordinates of the first standardized eigenvector α^1 of the $T \times T$ square matrix S whose elements are RV coefficients between every pair of indicator matrices (Lavit, 1984). Since all elements of S are non-negative, the coordinates α_t^1 are all non-negative and used to get the consensus matrix W^* between the T connectivity matrices : $W^* = \sum_{t=1}^{T} \alpha_t^1 W_t$. The weights α_t^1 represent the agreement between data tables and the compromise matrix.
We propose to apply hierarchical ascendant clustering algorithm on this compromise matrix considered as a similarity matrix to re-cluster the individuals as in CSPA. Therefore our method, denoted RVCONS, can be seen as a combination of WNMF considering the weighted aggregate matrix and of CPSA as we cluster the compromise matrix using classical hierarchical algorithm (rather than NMF). However, there are differences between the sets of weights associated with WNMF and RVCONS. Firstly, the way to get these weights is different : WNMF uses a iterative algorithm while RVCONS weights come from eigen elements derivations or singular value decomposition. Secondly, WNMF removes redundancy by giving an important weight to one single partition among highly correlated ones but in the same time it would give a important weight to a partition very different from the others. At the opposite, with RVCONS, this later partition would be considered as an outlier and then associated with a small weight.

3 Applications

We exemplify the proposed method on a simulated data set and on one from the UCI repository. We compare our consensus clustering method RVCONS with CSPA and WNMF. We use data sets with labelled individuals in order to have a reference partition.

3.1 Data sets descriptions

The simulated data set denoted D1, consists of 11 blocks of 5 variables. The RV coefficients between pair of blocks among the first 10 blocks are about 0.80, that is the blocks are highly correlated. We add one more block with a lower RV coefficient equal to 0.01 with each of the first 10 blocks.
The UCI Multiple Features data set contains 2000 individuals of handwritten numerals that were extracted from a collection of Dutch utility maps. These patterns were classified into 10 clusters ("0"–"9"), each having 200 individuals. Each individual is described by 649 features divided into the following 6 feature groups denoted B_1, B_2, B_3, B_4, B_5 and B_6 :

— mfeat-fou block : contains 76 Fourier coefficients of the character shapes ;
— mfeat-fac block : contains 216 profile correlations ;
— mfeat-kar block : contains 64 Karhunen-Loeve coefficients ;

	B_1	B_2	B_3	B_4	B_5	B_6
B_1	1	0.985	0.706	0.312	0.967	0.987
B_2	0.985	1	0.678	0.433	0.935	0.965
B_3	0.706	0.678	1	0.199	0.764	0.736
B_4	0.312	0.433	0.199	1	0.273	0.272
B_5	0.967	0.935	0.764	0.273	1	0.979
B_6	0.987	0.965	0.736	0.272	0.979	1

TAB. 1 – *the RV coefficients between the 6 blocks*

— mfeat-pix block : contains 240 pixel averages in 2D 3 windows ;
— mfeat-zer block : contains 47 Zernike moments ;
— mfeat-mor block : contains six morphological features.

Table 1 contains the RV coefficients between these 6 blocks. In this exemple there are different levels of similarity between blocks : blocks 1, 2, 5 and 6 are very highly correlated, block 3 has quite high RV values while block 4 has the lowest RV values and can be considered as not similar to the others blocks.

3.2 Application of consensus methods

The consensus methods have been applied with 30 initialisations of K-means to obtain initial partitions. The results in table 2 for the simulated data show quite similar initial partitions for the first 10 blocks with high accuracy mean values. The partition associated with the last block has lower accuracy mean values (around 0.5) for others contributory partitions as well as the reference partition. In table 3 are the means of 30 accuracy and Adjusted Rand indices computed between the consensus and contributory partitions. Table 4 contains the means of the weights used to get the compromise matrix.

Table 3 shows that RVCONS performs better than WNMF for the contributory partitions (excepted for P_{10}) as well as the reference partition. Moreover, comparing the last columns of table 2 and table 3, it can be seen that RVCONS consensus partition improves the accuracy and Adjusted Rand indices for the reference partition. RVCONS provides equal weights for the 10 highly correlated blocks and assigns a quite zero weight to the noisy block. This can explain the similar performances of RVCONS and CSPA. At the opposite, WNMF gives the most important weight to the last noisy block which has the lowest similarity with the reference partition. We consider this as a drawback of the WNMF method.
For the DMU data set, table 6(a), shows that the 3 consensus partitions have quite similar accuracy and Adjusted Rand indices for the reference partition. But there are differences for the contributory partitions : P_1, P_3 and P_5 have greater WNMF accuracy or Adjusted Rand indices than the RVCONS ones. RVCONS provides weights for the 6 blocks according more or less to their level of accuracy with the reference partition while WNMF gives the most important weight to the block 1 (associated with one of the lowest accuracy level with the reference partition) and the block 3 (associated with one with the highest accuracy with the reference partition). This may be related to LASSO constraints on WNMF weights (Tao, 2008).

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	Label
P_1	1.00	0.83	0.85	0.78	0.73	0.77	0.82	0.80	0.82	0.74	0.51	0.76
P_2	0.83	1.00	0.92	0.84	0.78	0.86	0.90	0.88	0.89	0.83	0.51	0.85
P_3	0.85	0.92	1.00	0.87	0.79	0.88	0.91	0.90	0.91	0.84	0.53	0.86
P_4	0.78	0.84	0.87	1.00	0.78	0.83	0.85	0.85	0.85	0.78	0.51	0.79
P_5	0.73	0.78	0.79	0.78	1.00	0.76	0.80	0.77	0.78	0.73	0.53	0.81
P_6	0.77	0.86	0.88	0.83	0.76	1.00	0.86	0.86	0.85	0.81	0.52	0.81
P_7	0.82	0.90	0.91	0.85	0.80	0.86	1.00	0.87	0.88	0.84	0.53	0.85
P_8	0.80	0.88	0.90	0.85	0.77	0.86	0.87	1.00	0.87	0.81	0.50	0.83
P_9	0.82	0.89	0.91	0.85	0.78	0.85	0.88	0.87	1.00	0.82	0.51	0.84
P_{10}	0.74	0.83	0.84	0.78	0.73	0.81	0.84	0.81	0.82	1.00	0.51	0.80
P_{11}	0.51	0.51	0.53	0.51	0.53	0.52	0.53	0.50	0.51	0.51	1.00	0.52

TAB. 2 – *The mean of the Accuracy between the initial partitions and the reference partition for D1 data set*

Method	Index	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	Label
CSPA	Acc	0.89	0.97	0.94	0.92	0.88	0.92	0.93	0.95	0.94	0.88	0.62	0.88
	AR	0.91	0.93	0.93	0.92	0.89	0.91	0.89	0.93	0.93	0.86	0.22	0.83
WNMF	Acc	0.80	0.84	0.83	0.80	0.77	0.81	0.83	0.83	0.82	0.79	0.70	0.85
	AR	0.89	0.90	0.90	0.89	0.86	0.88	0.86	0.90	0.90	0.88	0.25	0.80
RVCONS	Acc	0.89	0.97	0.94	0.91	0.88	0.92	0.96	0.95	0.94	0.87	0.62	0.88
	AR	0.92	0.93	0.94	0.92	0.89	0.91	0.89	0.93	0.93	0.86	0.22	0.83

TAB. 3 – *Accuracy and Adjusted Rand index between the initial partitions, the reference partition and the consensus partition for D1 data set*

Table	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8	W_9	W_{10}	W_{11}
RVCONS	0.10	0.10	0.10	0.09	0.10	0.10	0.10	0.10	0.10	0.10	**0.04**
WNMF	**0.16**	0.08	0.06	**0.12**	0.06	0.05	0.09	0.09	**0.14**	**0.15**	**0.41**

TAB. 4 – *Compromise coefficients for D1 data set*

	P_1	P_2	P_3	P_4	P_5	P_6	Label
P_1	1.00	0.72	0.57	0.69	0.46	0.49	0.64
P_2	0.72	1.00	0.71	0.84	0.53	0.53	0.86
P_3	0.57	0.71	1.00	0.66	0.46	0.40	0.70
P_4	0.69	0.84	0.66	1.00	0.53	0.50	0.81
P_5	0.46	0.53	0.46	0.53	1.00	0.46	0.57
P_6	0.49	0.53	0.40	0.50	0.46	1.00	0.51

TAB. 5 – *The mean of the Accuracy between the initial partitions and the reference partition for DMU data set*

4 Conclusion

We presented a method for clustering multiblock data based on the RV index and a simple eigenvector derivation to define a consensus similarity matrix which is used to re-cluster the individuals to find a consensus partition. The first results of its application on simulated data as

(a) Accuracy and Adjusted Rand Index between the consensus partition and the initial partitions and the reference partition

Method	Index	P_1	P_2	P_3	P_4	P_5	P_6	Label
CSPA	Acc	0.87	0.93	0.90	0.94	0.86	0.85	0.81
	AR	0.46	0.55	0.71	0.71	0.35	0.31	0.71
WNMF	Acc	0.87	0.91	0.92	0.91	0.86	0.83	0.80
	AR	0.47	0.40	0.65	0.55	0.47	0.31	0.70
RVCONS	Acc	0.86	0.93	0.90	0.94	0.86	0.84	0.80
	AR	0.45	0.56	0.74	0.74	0.35	0.31	0.70

(b) Compromise coefficients

Table	W1	W2	W3	W4	W5	W6
RVCONS	0.15	0.19	0.17	0.19	0.16	0.15
WNMF	0.24	0.12	0.26	0.14	0.14	0.10

TAB. 6 – *The mean of the Accuracy, Adjusted Rand Index and the weights of WNMF and RVCONS for the DMU data set*

well as one real data show better performances compare to WNMF, another weighted consensus clustering method. Work is on progress for more formal evaluations on simulated data as well as real one from batch process monitoring. In addition, we still studying the weights assignment step particulary in case of large number of blocks considering sparsity.

Références

Day, W.E. (1994). *Foreword: Comparison and consensus of classifications.*. Journal of Classification, 3(2), pp. 183-185.

Lavit, C., Escoufier,Y., Sabatier, R., et P. Traissac (1984). *The act (statis method)*. Statistics & Data Analysis. 18(1), 97)-119

Strehl, A., et Ghosh, J (2003). *Cluster ensembles a knowledge reuse framework for combining multiple partitions*. The Journal of Machine Learning Research. 3, 583-617

Tao, L. et Ding, C (2008). *Weighted consensus clustering*. Mij. 1(2), 97-119 (2008)

Robert, P. et Escoufier, Y (1976). *A unifying tool for linear multivariate statistical methods: the RV-coefficient*. Applied statistics. 27(3), 257-265

Summary

We address the issue of clustering individuals described by several homogeneous blocks of variables. Reformulating it as a problem of consensus of partitions, we propose a weighted consensus method based on the RV index to find the partition of the individuals. The main idea of this consensus method is to agglomerate the separate initial partitions obtain from each block into a global partition which has to be the most similar to the initial partitions according to the RV coefficient. The proposed method is illustrated and compared to CSPA (Cluster based Similarity Partitioning Algorithm) and a weighted consensus clustering method based on non-negative matrix factorization.

Application de la classification symbolique à l'estimation des coûts de production agricoles[i]

Dominique Desbois *

* Economie publique, Inra, 16 rue Claude Bernard, F-75231 PARIS CEDEX 05.
dominique.desbois@inra.fr
https://www6.versailles-grignon.inra.fr/economie_publique_eng/PersonalPages2/Dominique-Desbois

Réumé. Cette communication utilise la classification des données symboliques pour explorer les similitudes entre distributions d'estimations quantiles conditionnelles, en l'appliquant au problème de l'allocation des coûts spécifiques en agriculture. Après avoir rappelé le cadre conceptuel de l'estimation des coûts de production agricole, la première partie présente le modèle empirique, l'approche de régression quantile et la technique de classification des données d'intervalle utilisée. La seconde partie présente l'analyse comparative entre douze États membres européens des résultats issus de la classification hiérarchique divisive des intervalles d'estimation.

1. Introduction

L'intégration de l'agriculture dans les 28 États membres résultant de l'élargissement de l'Union européenne (UE) a suscité des besoins récurrents d'estimation des coûts de production des principaux produits agricoles, tout au long des réformes de la politique agricole commune (PAC), sur les marchés concurrentiels comme réglementés. L'analyse des coûts de production agricole est un outil d'analyse des marges des agriculteurs : elle permet d'évaluer la compétitivité prix des exploitations agricoles, l'un des éléments majeurs du développement et de la durabilité des chaînes alimentaires dans les régions européennes. Pour répondre aux besoins de simulations et d'analyses d'impact pour les différentes organisations communes de marchés, nous devons fournir des informations sur l'ensemble de la répartition des coûts de production afin d'évaluer les options de politique agricole publique. En se basant sur le constat de l'asymétrie et de l'hétérogénéité au sein de la distribution empirique des intrants agricoles, nous avons proposé une méthodologie adaptée à l'estimation de la distribution empirique des coûts de production spécifiques des principaux produits agricoles dans un contexte européen où les exploitations agricoles restent principalement multi-productives (Desbois, Butault et Surry, 2017).

À partir de cette approche, nous présentons le modèle empirique d'estimation des coûts de production spécifiques, inspirée d'une approche micro-économétrique de répartition des coûts pour construire une matrice entrées-sorties au niveau national (Divay et Meunier, 1980). Puis, nous rappelons la méthodologie d'estimation

permettant de prendre en compte l'hétérogénéité des exploitations agricoles, selon l'approche du quantile conditionnel proposée par Koenker et Bassett (1978). Ensuite, pour explorer les distributions empiriques des intervalles d'estimation de quantiles conditionnels, nous présentons la procédure de classification utilisée (Chavent *et al.*, 2007) dans le cadre conceptuel de l'analyse symbolique de données (Billiard et Diday, 2006). Nous introduisons alors le graphique des résultats de la procédure de classification appliquée aux intervalles d'estimation des quantiles conditionnels. Enfin, nous concluons sur la pertinence de cette approche appliquée à la production de porc.

2. Cadre conceptuel et aspects méthodologiques

Nous présentons d'abord la méthodologie d'estimation des coûts spécifiques. Puis, nous introduisons l'outil de classification des intervalles d'estimation dans le formalisme de l'analyse symbolique de données.

2.1 Le modèle d'estimation des coûts spécifiques de production

Inspiré de Divay et Meunier (1980), l'affectation de la somme x_i des coûts des intrants pour l'exploitation agricole i est réalisée par décomposition linéaire le long des produits bruts Y_i^j de l'exploitation agricole i pour chaque production j, où u_i est un vecteur aléatoire d'espérance nulle :

(1) $$x_i = \sum_{j=1}^{p} \beta_j Y_i^j + u_i$$

Comme Cameron et Trivedi (2005), nous supposons que le processus générateur de données est un modèle linéaire à hétéroscédasticité multiplicative caractérisé par :

(2) $$x = Y'\beta + u \text{ avec } \quad u = Y'\alpha \times \varepsilon \quad \text{et} \quad Y'\alpha > 0$$

où $\varepsilon \sim iid[0, \sigma]$ est un vecteur aléatoire identique et indépendant à moyenne nulle et variance constante σ^2. Sous cette hypothèse, $\mu_q(x|Y, \beta, \alpha)$, le q^e quantile conditionnel du coût de production *x*, conditionné par *Y* et les paramètres, α et β se déduit analytiquement comme suit :

(3) $$\mu_q(x|Y, \beta, \alpha) = Y'[\beta + \alpha \times F_\varepsilon^{-1}(q)] = Y'\gamma.$$

où F_ε est la distribution cumulée des erreurs.

Le coefficient technique du q^e quantile de coûts spécifiques pour le j^e produit est défini par le j^e composant du vecteur de pente :

(4) $$\gamma^j(q) = [\beta + \alpha \times F_\varepsilon^{-1}(q)]^j$$

Au moins deux types de modèle peuvent être dérivés de cette spécification (D'Haultfœuille et Givord, 2014) :

i) $x = Y'\beta + u$ avec $u = K\epsilon$, à résidus homoscédastiques ($V(\epsilon|Y) = \sigma^2$), dénommé modèle à *translation simple, i.e.* soit un modèle linéaire à pentes

homogènes ; puisque $Y'\alpha = K$ est constant, les quantiles conditionnels $\mu_q(x|Y,\beta,\alpha) = Y'\beta + KF_e^{-1}(q)$ ont tous la même pente β, mais diffèrent seulement d'un écart constant, croissant à mesure que l'ordre q du quantile augmente ;

ii) $x = Y'\beta + (Y'\alpha)\varepsilon$ et $Y'\alpha > 0$ à résidus hétéroscédastiques, dénommé modèle à *translation-échelle, i.e.* le modèle linéaire de quantiles conditionnels à pentes hétérogènes.

2.2 Classification par intervalles des distributions de coûts spécifiques

Pour un produit donné j_0 tel que le porc et le l^e pays européen, l'intervalle d'estimation des coefficients techniques pour les coûts spécifiques

(5) $$z_l^q = \left[Inf\_\hat{\gamma}_l^{j_0}(q); Sup\_\hat{\gamma}_l^{j_0}(q)\right]$$

est obtenu par *bootstrap* marginal par chaînes de Markov (He et Hu, 2002). Objets symboliques, les L distributions nationales $\Omega = \{\omega_1, \cdots, \omega_l, \cdots \omega_L\}$ sont décrites par un ensemble de $Q = 5$ descripteurs[1], qui sont les intervalles d'estimation des coefficients techniques pour les quantiles conditionnels $Z = \{z^1, \ldots, z^q, \ldots, z^Q\} = \{z^{0.10}, z^{0.25}, z^{0.50}, z^{0.75}, z^{0.90}\}$.

Les dissimilarités locales entre pays *l* et pays *l'*, associées aux intervalles d'estimation des coefficients techniques pour le q^e quantile conditionnel, sont calculées selon la distance euclidienne :

(6) $$\delta_M\left(z_l^q, z_{l'}^q\right) = \sqrt{\left(Inf\_\hat{\gamma}_l^{j_0}(q) - Inf\_\hat{\gamma}_{l'}^{j_0}(q)\right)^2 + \left(Sup\_\hat{\gamma}_l^{j_0}(q) - Sup\_\hat{\gamma}_{l'}^{j_0}(q)\right)^2}$$

Pour cette métrique *M*, une dissimilarité globale entre pays *l* et pays *l'* basée sur les différences entre distributions nationales des intervalles d'estimation des coefficients techniques est calculée selon le critère quadratique suivant :

(7) $$d(\omega_l, \omega_{l'}) = \left(\sum_{q=1}^{Q} \delta_M^2\left(z_l^q, z_{l'}^q\right)\right)^{1/2}.$$

Étant donné la matrice des dissimilarités entre distributions nationales de coûts spécifiques issues des calculs précédents, nous pouvons utiliser les méthodes de classification non supervisée. De façon similaire à la méthode de Ward, Chavent *et al.* (2007) proposent un algorithme de classification hiérarchique par division hiérarchique sur données symboliques (DIVCLUS-T), valable pour les données d'intervalle et les données catégorielles. Par la suite, nous détaillons pour les données

1 Le choix d'un petit nombre de descripteurs a été fait pour des raisons de comparabilité avec des approches graphiques plus classiques (Desbois, 2015). Cependant, si les objectifs de l'analyse l'exigeaient, il pourrait être étendu sans inconvénient aux ensembles de descripteurs de cardinalité supérieure : déciles (*Q = 9*), voire centiles (*Q = 99*).

d'intervalle les principes opérationnels de cette procédure de classification non supervisée.

L'algorithme divisif de classification hiérarchique partage récursivement chaque classe en deux sous-classes, à partir de l'ensemble des pays en tant qu'objets symboliques $\Omega = \{\omega_1, \cdots, \omega_l, \cdots \omega_L\}$. À chaque partition en k classes symboliques $\mathrm{P}_K = \{C_1, \cdots, C_k \cdots, C_K\}$, une classe doit être scindée pour obtenir une partition P_{K+1}, à $K+1$ classes, optimisant le critère de sélection basé sur l'inertie.

L'inertie de la k^{e} classe est définie par $I(C_k) = \sum_{l \in C_k} \mu_l d_M^2 \left(z_l, g(C_k)\right)$ où μ_l est le poids du l^{e} pays et $g(C_l)$ est le barycentre de classe définie par :

(8) $$g(C_k) = \frac{1}{\sum_{l \in C_k} \mu_l} \sum_{l \in C_k} \mu_l z_l.$$

L'inertie intra est définie par la somme des inerties des classes à leurs barycentres :

(9) $$W(P_K) = \sum_{k=1,\ldots,K} I\,(C_k)$$

L'inertie inter est définie par l'inertie des barycentres relatives au barycentre global g de l'ensemble Ω , comme suit:

(10) $$B(P_K) = \sum_{k=1,\ldots,K} \mu_k d_M^2\,(g(C_k), g) \text{ where } \mu_k = \sum_{l=1,\ldots,k} \mu_l.$$

Pour une partition P_K, l'inertie totale regroupe l'inertie intra avec l'inertie inter :

(11) $$I(\Omega) = W(P_K) + B(P_K).$$

Ainsi, minimiser l'hétérogénéité (mesurée par *W*) est équivalent à maximiser l'homogénéité (mesurée par *B*).

Généré par la réponse binaire (*oui*/*non*) à une question $\Psi = [z^q \leq c\,?]$, notons $\left\{A_k, \overline{A_k}\right\}$ la bipartition induite de la classe C_k formée de n_k objets. Afin de choisir parmi les $n_k - 1$ bipartitions possibles de la classe C_k, le critère discriminant est defini par le ratio suivant :

(12) $$D(\Psi) = \frac{B^q\left(A_k, \overline{A}_k\right)}{I^q(C_k)} = 1 - \frac{W^q\left(A_k, \overline{A}_k\right)}{I^q(C_k)},$$

où l'inertie inter $B^q\left(A_k, \overline{A}_k\right)$ et l'inertie $I^q(C_k)$ sont calculées par rapport au q^e quantile conditionnel. Ainsi, minimiser l'inertie intra $\mathrm{W}\left\{A_k, \overline{A_k}\right\}$ équivaut à maximiser l'inertie inter $\mathrm{B}\left\{A_k, \overline{A_k}\right\}$ et, par conséquence, le critère discriminant $D(\Psi)$.

Comme dans la méthode de Ward, la « *hiérarchie supérieure* » (Mirkin, 2005) à la partition P_K est indexée par l'indice h de la classe C_K , définie par son inertie inter comme suit:

(13) $$h(C_k) = B\left(A_k, \overline{A_k}\right) = \frac{\mu(A_k)\mu\left(\overline{A_k}\right)}{\mu(A_k) + \mu\left(\overline{A_k}\right)} d^2\left(g(A_k), g\left(\overline{A_k}\right)\right)$$

L' algorithme DIVCLUS-T scinde la classe C_K^* qui maximise $h(C_K)$, assurant que la partition suivante $P_{K+1} = P_K \cup \left\{A_K, \overline{A_K}\right\} - C_K^*$ présente la valeur minimum de l'inertie intra, conformément à l'équation suivante

(14) $$W(P_{K+1}) = W(P_K) - h(C_K^*).$$

3. La hiérarchie divisive des estimations de coûts

Nous analysons les résultats obtenus en coûts spécifiques pour le porc (figure 1), l'un des produits sélectionnés dans le cadre du projet FACEPA, sur la base 2006 du

réseau européen d'information comptable agricole (UE-RICA). Les coûts spécifiques du porc comprennent principalement les intrants alimentaires, vétérinaires, et énergétiques.

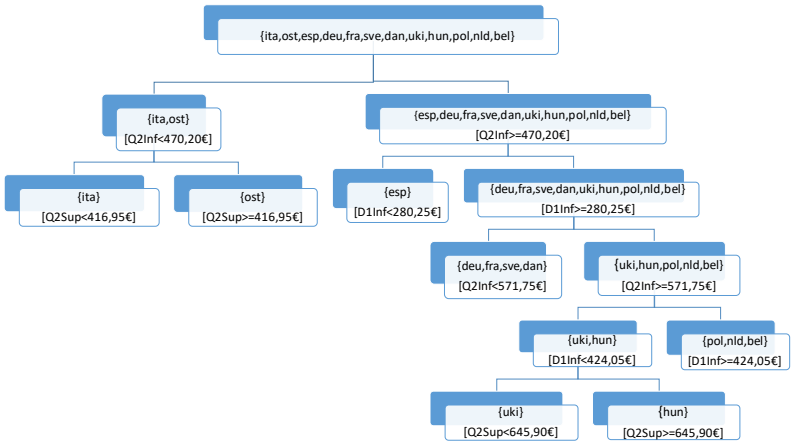


FIG. 1 - Classification de 12 pays européens sur la base de coûts spécifiques pour 1 000 € de produit brut porcin. Source : traitement de l'auteur, selon UE-RICA 2006.

Au sommet de la hiérarchie divisive, la procédure de regroupement permet d'identifier deux modèles contrastés de distributions empiriques des coefficients techniques du porc pour des coûts de production spécifiques : d'une part, l'Autriche (OST), caractérisée par le supremum de la médiane (Q2Sup $\geq$ 416,95 €), est la distribution la plus plate représentant le modèle à *translation simple* basé sur l'hypothèse de producteurs homogènes en leurs coûts spécifiques ; d'autre part, l'Italie (ITA), séparée par le supremum médian (Q2Sup < 416,95 €), présente la deuxième distribution la plus pentue illustrant le modèle à *translation-échelle*, formalisant l'hypothèse de producteurs hétérogènes en leurs coûts spécifiques.

4. Conclusions

L'analyse des intervalles d'estimation à l'aide d'une classification hiérarchique divisive permet d'identifier différents types de distributions nationales de coûts spécifiques pour le porc. Les différences entre les groupes de pays sont délimitées par des seuils exprimés selon les quantiles conditionnels en termes unitaires du produit brut. Ces analyses identifient deux modèles d'échelle de répartition des coûts, celui de la translation simple opposé à celui de la translation-échelle. Ces seuils peuvent être utilisés pour segmenter les populations d'exploitations agricoles afin d'analyser ultérieurement les impacts différentiels des mesures de politique agricole. L'application de cette méthodologie est envisagée au deuxième niveau de la Nomenclature européenne des unités territoriales statistiques (NUTS 2), soit 281 régions.

Références

Billard L., Diday E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*, 321 p.

Chavent, M., *et al*. (2007). DIVCLUS-T: A monothetic divisive hierarchical clustering method. *Comput. Statist. Data Anal.*, doi: 10.1016/j.csda.2007.03.013.

Desbois D. (2015). *Estimation des coûts de production agricoles : approches économétriques*. Thèse ABIES-AgroParisTech, dirigée par J.C. Bureau et Y. Surry, 249 p.

Desbois D., Butault J.-P., Surry Y. (2017). Distribution des coûts spécifiques de production dans l'agriculture de l'Union européenne : une approche reposant sur la méthode de régression quantile, *Economie rurale*, n° 361, pp. 3-22.

Divay J.F., Meunier F. (1980). Deux méthodes de confection du tableau entrées-sorties. *Annales de l'INSEE*, n°37, pp. 59-109.

D'Haultfœuille X., Givord P. (2014). La régression quantile en pratique. *Économie et Statistique*, n°471, pp. 85-111.

He X., Hu F. (2002). Markov Chain Marginal Bootstrap, *Journal of the American Statistical Association*, n°97, pp. 783-795.

Koenker R., Bassett G. (1978). Regression quantiles. *Econometrica*, vol. 46, pp. 33-50.

Mirkin, B. (2005*). Clustering for Data Mining. A Data Recovery Approach*. Chapman & Hall, CRC Press, London, Boca Raton, FL, 366 p.

Summary

This communication analyses the similarities between distributions of conditional quantile estimates, applying it to the problem of cost allocation in agriculture. The first part presents the empirical model, the quantile regression approach and the interval data clustering technique used. The second part presents the comparative analysis of the results between twelve European Member States.

---

[i] Cette communication développe certains travaux de l'auteur réalisés lors de la préparation de sa thèse (Desbois, 2015), co-dirigée par Y. Surry et JC. Bureau, dans le cadre du projet FACEPA (*Farm Accountancy Cost Estimation and Policy Analysis*) du 7e programme-cadre de la Communauté européenne (FP7 / 2007 2013, approbation n ° 212292). Cette mention n'implique aucune approbation par les personnes et organismes cités du texte placé sous l'entière responsabilité de l'auteur.

Impact des mesures de similarité sémantique dans un algorithme de partitionnement : d'un cas biomédical à la détection de comportements de consommation

Jocelyn Poncelet*, Pierre-Antoine Jean*, François Trousset*,
Sebastien Harispe*, Nicolas Pecheur**, Jacky Montmain*

*LGI2P - IMT Mines Alès - Université de Montpellier, Alès, France
{prénom}.{nom}@mines-ales.fr
**TRF retail - 116 Allée Norbert Wiener - Nîmes, France
{prénom}.{nom}@trfretail.com

Résumé. Les approches permettant de regrouper/segmenter des objets partageant ou non des caractéristiques similaires sont nombreuses. Des distances classiques sont utilisées selon l'hypothèse que ces caractéristiques sont indépendantes et définissent un espace métrique. Cependant, lorsque ces caractéristiques sont organisées dans une représentation des connaissances, ces métriques deviennent discutables. Cet article propose la comparaison de distances et de mesures de similarité au sein d'une approche de partitionnement ascendant hiérarchique. L'étude cherche à mettre en évidence l'intérêt d'approches sémantiques permettant de détecter des comportements de consommation. Un parallèle avec le domaine biomédical a été réalisé pour pallier le manque de données dans le domaine de la grande distribution et valider notre approche.

1 Introduction et état de l'art

Dans les secteurs industriels tels que la grande distribution, le partitionnement de clients est une problématique récurrente. Cela permet de recueillir des informations essentielles pour contrôler et définir des stratégies commerciales et marketing orientées autour d'indicateurs, *e.g.* RFM pour *Recency, Frequency, Monetary* ou CLV pour *Customer Lifetime Value* (Chen et al., 2009; Ching-Hsue et You-Shyang, 2009). Les approches de partitionnement traditionnelles considèrent une représentation vectorielle des clients où les dimensions sont essentiellement des mesures monétaires des comportements de consommation (fréquence, volume d'achats, etc.). Elles conduisent donc plus à une segmentation des clients orientée « valeur monétaire » qu'à une véritable caractérisation du client. Il existe pourtant un réel besoin d'analyses dirigées par la caractérisation des produits achetés pour identifier précisément les typologies de comportements de consommation afin de mieux comprendre et anticiper les changements dans les habitudes d'achats. Dans ce cadre, les dimensions de l'espace à partitionner pourraient correspondre aux produits en vente dans un supermarché avec pour chaque composante une

valeur booléenne (acheté ou pas) [1], une fréquence d'achats, etc. Ainsi, si trois clients achètent respectivement dans une boulangerie : des bonbons, des chewing-gums et une baguette ; ils seront considérés équidistants si l'on utilise, par exemple, une distance euclidienne dans cet espace de dimension 3. Pourtant, de façon intuitive, les personnes qui achètent des bonbons ou des chewing-gums ont des comportements proches et distincts du client qui achète du pain. Cette intuition repose sur l'idée qu'il existe un lien sémantique qui « rapproche » bonbons et chewing-gums : ce sont des friandises. Pour formaliser cette intuition, les mesures de similarité sémantique constituent une solution intéressante pour introduire la connaissance a priori associée à un domaine. Par ailleurs, la représentation vectorielle des clients à l'échelle d'une grande surface ($1,5 \times 10^6$ produits dans le calcul de voisinages) conduit à une projection très peu dense et difficilement interprétable en termes de segments. De nouveau, les mesures de similarité sémantique vont permettre d'introduire une notion d'abstraction qui résout ce problème.

Cette étude s'inscrit dans ce contexte et propose une comparaison des performances d'algorithmes de segmentation en fonction des distances et mesures de similarité utilisées. A des fins de validation, cette problématique est transposée dans le domaine du biomédical où jeux de données et structurations de la connaissance sont davantage disponibles et formalisés. Le protocole expérimental [2] décrivant le corpus, la métrique d'évaluation et la méthodologie élaborée sont exposés en section 2. Les résultats et la discussion sont détaillés dans la section 3.

2 Protocole expérimental

2.1 Description du corpus et métrique d'évaluation

À notre connaissance, aucun jeu de données public issu de la grande distribution n'est disponible pour réaliser des expérimentations. Toutefois, des jeux de données provenant du domaine biomédical proposent des similarités fortes avec les données de la grande distribution. Le jeu de données recherché doit représenter un ensemble d'objets (*i.e.* nos clients) au travers d'un ensemble de concepts hiérarchisés au sein d'un ordre partiel (*i.e.* nos produits) pour se conformer à la particularité de notre problématique (l'ordre partiel n'est autre qu'une hiérarchie d'abstraction dans laquelle les produits vendus seraient les concepts les plus spécifiques). En outre, la validation de nos approches dans un contexte multi-classes implique également qu'une étiquette unique soit associée à chacun de nos objets (*i.e.* l'étiquette de leur segment).

Les travaux de Zhou et al. (2014) proposent une liste de maladies/symptômes désambiguïsés dans la taxonomie MeSH [3]. L'analogie avec notre problématique du retail s'opère de la façon suivante : un symptôme est à la maladie ce qu'un produit est au consommateur. Ainsi, chaque maladie peut être vue comme un vecteur de symptômes, et un client peut être perçu comme un vecteur de produits achetés. La désambiguïsation des maladies et des symptômes dans la taxonomie du MeSH (c'est-à-dire la transposition des maladies et des symptômes dans

1. Ce type de représentation est, par exemple, utilisé dans le cadre de la recherche de règles d'association où les colonnes correspondent aux produits du magasin et les lignes aux différents clients.

2. Afin d'assurer la reproductibilité des évaluations, elles sont réalisées sur un jeu de données public et le code développé est mis à disposition à l'adresse suivante : https ://github.com/PAJEAN/diseases_segmentation.

3. Le *MeSH* pour *Medical Subject Headings*, est le thésaurus de référence dans le domaine biomédical. Il est notamment utilisé pour indexer les articles de PubMed.

la structure hiérarchique du MeSH) permet à la fois d'appliquer les mesures de similarité sémantique pour analyser la ressemblance/différence entre deux maladies et d'attribuer une étiquette unique aux maladies par le biais de leurs concepts plus abstraits. Ainsi, l'étiquetage des maladies sous un même concept abstrait permet d'évaluer la pertinence du partitionnement qui a été réalisé : deux maladies partageant plusieurs symptômes seront déclarées proches et devraient porter la même étiquette abstraite (comme deux consommateurs qui achètent des produits similaires devraient être classés dans un même segment de clientèle).

Le jeu de données finalement utilisé contient 1517 maladies et 223 symptômes. Les différentes méthodes de partitionnement sont évaluées et comparer à l'aide de la F_1-mesure. Cette mesure d'évaluation se base sur la moyenne harmonique entre la précision et le rappel calculés sur toutes les paires de maladies du jeu de données (Hatzivassiloglou et McKeown, 1993).

2.2 Description des expérimentations

Les expérimentations réalisées ont pour objectif d'étudier les performances de mesures sémantiques dans le cadre d'une problématique de segmentation. Le protocole expérimental mis en place s'appuie sur un partitionnement ascendant hiérarchique qui tient compte des similarités sémantiques. Leur regroupement est réalisé par la méthode de Ward qui minimise la distance à l'intérieur des groupes (distance *intra*-groupes) tout en maximisant la distance entre les groupes (distance *inter*-groupes) (Murtagh, 2014). Dans ce protocole, les performances obtenues avec ces partitionnements sont comparées à deux méthodes de référence : le *K-means* et l'utilisation de métriques sur des espaces vectoriels couplés au partitionnement ascendant hiérarchique.

***K-means* et mesures vectorielles** Dans les travaux de Zhou et al. (2014) chaque objet (maladie) est représenté par un vecteur de réels sur l'ensemble des concepts (symptômes). Chaque composante de ce vecteur mesure la force d'association du concept avec l'objet. Cette force d'association réelle est calculée sur la base d'un TF-IDF (Sparck Jones, 1972). À partir de ces vecteurs d'observation, les auteurs calculent la distance entre les objets avec une distance cosinus. Pour des résultats plus exhaustifs, la distance euclidienne est également expérimentée et une restriction exploitant uniquement des vecteurs binaires (la force d'association existe ou non) pour un objet donné est aussi proposée.

Mesures de similarité sémantique Les similarités sémantiques entre les objets sont calculées à partir d'une structuration taxonomique. Ces similarités comparent des groupes de concepts associés aux objets par le biais de mesures dites *groupwise* (Harispe et al., 2015). Ces mesures se basent elles-mêmes sur des mesures dites *pairwise* permettant de calculer la similarité entre deux concepts au sein de la taxonomie. Certaines d'entre elles exploitent le contenu informationnel (*Information Content*, IC) associé aux concepts, c'est-à-dire, la quantité d'information associée à un concept (plus un concept est spécifique, plus son contenu informationnel est grand). Il existe plusieurs mesures *groupwise*, comme il existe plusieurs mesures *pairwise* et plusieurs définitions pour l'IC. L'objectif de cette étude est de présenter les performances et l'intérêt d'approches sémantiques.

Les mesures *groupwise* permettent la comparaison des ensembles de concepts rattachés à des objets. Il en existe deux catégories principales : les mesures directes et indirectes. Les me-

sures *groupwise* directes comparent les ensembles de concepts sans tenir compte de leur position dans la taxonomie. Pour cette étude, la distance de Jaccard a été mise en place. Concernant les mesures *groupwise* indirectes, elles agrègent les similarités obtenues des mesures de similarités *pairwise*. Cette étude exploite la BMA pour *Best Match Average* (Pesquita et al., 2007). Pour les mesures de similarité *pairwise*, la littérature en relate deux principaux types : les mesures basées sur le contenu informationnel (IC) et, celles basées sur la notion de plus court chemin dans la taxonomie (Harispe et al., 2015). Dans cette étude, les mesures de similarité *pairwise* mises en place sont respectivement la mesure de Resnik (Resnik, 1995) et la mesure de Wu & Palmer (Wu et Palmer, 1994). Enfin, concernant le contenu informationnel (IC), nous distinguons les IC intrinsèques et extrinsèques. Les IC intrinsèques prennent en compte uniquement les propriétés topologiques de la structure taxonomique du graphe sémantique. Ce type d'IC est généralement lié à la position d'un concept dans la taxonomie. L'IC intrinsèque utilisé pour cette étude est celui de Seco (Seco et al., 2004). Pour les IC extrinsèques, introduits par Resnik (Resnik, 1995), ils étendent l'approche intrinsèque en prenant également en considération la fréquence d'un concept dans une base d'observation (*e.g.* corpus). L'IC extrinsèque de Resnik est utilisé pour cette étude. Dans le cadre des expérimentations, les objets sont les maladies et les symptômes, les concepts. Pour plus d'information concernant les mesures de similarité sémantique mises en œuvre grâce à la *Semantic Measures Library*, le lecteur est invité à se référer aux travaux de Harispe et al. (2014).

3 Résultats et discussion

Le tableau 1 présente les résultats obtenus sur le corpus établi pour cette étude (cf. sous-section 2.1). A noter que les vecteurs d'observation du *K-means* et les matrices de distance dans le cadre du partitionnement ascendant hiérarchique sont soit, exploités tels quels, soit, *normalisés* pour observer l'impact de la normalisation sur le processus de segmentation.

	F_1-mesure	F_1-mesure (*normalisées*)
K-Means		
Vecteurs binaires	0.114	0.156
Vecteurs TF-IDF	0.113	0.136
Mesures vectorielles		
Vecteurs binaires, distance euclidienne	0.078	0.074
Vecteurs TF-IDF, distance euclidienne	0.104	0.094
Vecteurs binaires, distance cosinus	0.104	0.109
Vecteurs TF-IDF, distance cosinus	0.123	0.140
Mesures de similarité sémantique		
Jaccard	0.086	0.083
Wu & Palmer, BMA	0.108	0.124
IC Seco, Resnik, BMA	0.104	0.127
IC Resnik, Resnik, BMA	0.127	**0.182**

TAB. 1 – *Résultats des partitionnements.*

La F_1-mesure permet de mesurer la performance d'une configuration donnée à regrouper des maladies à partir de leurs symptômes sous une même étiquette abstraite (classe de maladies). A partir des résultats obtenus, nous pouvons dresser deux constats. Le premier repose sur la normalisation des vecteurs d'observation et de la matrice des distances. Dans la majorité des cas, le processus de normalisation a un impact positif et significatif sur la F_1-mesure. Ces résultats démontrent l'importance d'un tel processus lors d'une phase de segmentation. Le second constat, quant à lui, porte sur la pertinence des mesures de similarité sémantique au sein d'un processus de partitionnement appliqué avec des objets caractérisés par un ensemble de concepts structurés au sein d'un ordre partiel. Les résultats montrent que les mesures de similarité sémantique sont plus performantes avec la BMA associée à la mesure *pairwise* de Resnik et l'IC de Resnik avec une amélioration de la F_1-mesure de 16% au regard de ceux obtenus avec le *K-means*. Nous sommes conscients que ces résultats sont spécifiques aux particularités du jeu de données (typologie du MeSH, nombre de symptômes associés aux maladies, nombre de symptômes différents dans le jeu de données). Toutefois, ils permettent d'apporter une réponse objective à l'intérêt des mesures de similarité sémantiques dans un processus de segmentation où les données sont structurées par un ordre partiel.

Pour reprendre l'analogie avec le domaine du retail, l'utilisation de cette méthodologie permettra de regrouper les clients en fonction de leurs habitudes d'achats. Les mesures de similarité sémantique dans le processus de partitionnement apporteront des informations plus intuitives permettant la proposition d'assortiments et de services adaptés à un segment de clientèle.Ce partitionnement dépendra alors des produits achetés et de la structuration de la taxonomie de produits (*e.g.* chewing-gums et bonbons sont des friandises). Ces informations permettront, pour un preneur de décision, de mieux connaître sa clientèle et de proposer des stratégies de fidélisation adéquates.

4 Conclusion et perspectives

Dans cet article, nous proposons une alternative potentielle aux méthodes de segmentation de clients, qui se basent sur des métriques appliquées à des espaces vectoriels, en considérant les relations de similarité pouvant exister entre les dimensions de ce même espace. Ainsi, nous mettons en avant l'intérêt d'employer une connaissance *a priori* au travers des mesures de similarité sémantique afin d'exploiter les liens existants entre les caractéristiques des objets comparés. Un parallèle avec le domaine biomédical qui offre des jeux de données structurées nous a permis d'amorcer la validation de la pertinence de notre partitionnement sémantique. Les expérimentations nous ont permis d'observer de meilleures performances associées aux mesures de similarité comparativement aux méthodologies de segmentation *classiques*. En terme de perspective, nous allons employer les mesures de similarité sémantique sur des données réelles issues de la grande distribution pour identifier les habitudes d'achats des consommateurs : par exemple, différencier les consommateurs qui viennent principalement pour des produits alimentaires de ceux qui achètent des produits ménagers. Pour aller plus loin, nous envisageons également de travailler sur l'évolution des segments pour être en mesure d'identifier les changements d'habitudes grâce à l'analyse des leurs trajectoires (Gaffney et Smyth, 1999) et prédire les nouvelles tendances (*e.g* vegan).

Références

Chen, Y. L., M. H. Kuo, S. Y. Wu, et K. Tang (2009). Discovering recency, frequency, and monetary (rfm) sequential patterns from customers' purchasing data. *Electronic Commerce Research and Applications 8*(5), 241–251.

Ching-Hsue, C. et C. You-Shyang (2009). Classifying the segmentation of customer value via rfm model and rs theory. *Expert Systems With Applications 36*(3), 4176–4184.

Gaffney, S. et P. Smyth (1999). Trajectory clustering with mixtures of regression models. *KDD 99*(2), 63–72.

Harispe, S., S. Ranwez, S. Janaqi, et J. Montmain (2015). Semantic similarity from natural language and ontology analysis. *Synthesis Lectures on Human Language Technologies*.

Harispe, S., D. Sánchez, S. Ranwez, S. Janaqi, et J. Montmain (2014). A framework for unifying ontology-based semantic similarity measures : a study in the biomedical domain. *Journal of Biomedical Informatics 48*, 38–53.

Hatzivassiloglou, V. et K. R. McKeown (1993). Towards the automatic identification of adjectival scales: Clustering adjectives according to meaning. *Association for Computational Linguistics*, 172–182.

Murtagh, F. (2014). Ward's hierarchical agglomerative clustering method: Which algorithms implement ward's criterion? *Journal of Classification 31*, 274–295.

Pesquita, C., D. Faria, H. Bastos, A. Falcao, et F. Couto (2007). Evaluating go-based semantic similarity measures. *Proc 10th Annual Bio-Ontologies Meeting 37*(40).

Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. *Proceedings of IJCAI-95*, 448—-453.

Seco, N., T. Veale, et J. Hayes (2004). An intrinsic information content metric for semantic similarity in wordnet. *16th European Conference on Artificial Intelligence*, 1–5.

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation 28*, 11–21.

Wu, Z. et M. Palmer (1994). Verb semantics and lexical selection. *32nd. Annual Meeting of the Association for Computational Linguistics*, 133—-138.

Zhou, X., J. Menche, A. L. Barabási, et A. Sharma (2014). Human symptoms–disease network. *Nature communications 5*, 4212.

Summary

Segmenting approaches used to group objects that share similar features are numerous. On the hypothesis that characteristics are independent and defined in a metric space, conventional distances are used. When these characteristics are instead organized in a representation of knowledge, these metrics become questionable. This article proposes the comparison of distances within a hierarchical ascending partitioning approach. The study aims to highlight the interest of semantic approaches to detect customers behavior. A parallel with the biomedical field was made to overcome the lack of data related to the retail sector.

Nouvelles bases des règles d'association non-redondantes

Parfait Bemarisika*, André Totohasina*

*Laboratoire de Mathématiques et Informatique, Université d'Antsiranana, Madagascar.
bemarisikap7@yahoo.fr, andre.totohasina@gmail.com

Résumé. Dans ce papier, nous proposons des nouvelles bases des règles d'association positives et négatives les plus informatives non-redondantes.

1 Introduction et Motivations

Notre étude se situe dans le cadre de la science des données. Nous nous intéressons au concept des bases des règles d'association informatives, qui est un sous-ensemble des règles non redondantes à partir duquel, on peut retrouver l'ensemble de toutes les règles d'association valides. Etant donné un ensemble $\mathcal{I}$ de motifs de la base de données $\mathcal{B}$, une règle d'association est une implication logique entre deux motifs disjoints X et Y de $\mathcal{I}$, de la forme $X \rightarrow Y$, où X est la prémisse et Y le conséquent. Une règle *informative* est celle qui, à partir de la plus petite prémisse, fournit le plus grand conséquent. Intuitivement, la règle r_1 est dite redondante moins informative que r_2 si (i) elle partage la même information que r_2, et (ii) sa prémisse est sur-ensemble de la prémisse de r_2 et son conséquent est sous-ensemble du celui de r_2.

Le problème de redondance a été abordé par plusieurs approches (Guigues et Duquenne, 1986; Pasquier, 2000; Kryszkiewicz, 2002) dites bases des règles non redondantes. Toutefois, ces approches ne concernent que des règles positives, et ce, avec le couple moins sélectif, support-confiance (Agrawal et Srikant, 1994) comme nous démontrerons dans la section 2. Les règles négatives [1] sont donc dans l'ombre. Or, cette limite ne permet pas d'appréhender tous les besoins de la science des données (big data), il faut aussi des règles négatives.

Afin de pallier ces limites, nous proposons des nouvelles bases informatives des règles d'association positives et négatives non redondantes, et ce, à l'aide d'une mesure plus sélective, M_{GK} (Feno et al., 2006). En s'appuyant sur ces modèles, nous proposons aussi un algorithme, NON-REDUNDANT-RULES, qui retourne les règles d'association non-redondantes. Le reste de ce papier est organisé comme suit. Le cadre formel est donné en section 2. La section 3 détaille l'approche proposée. Une conclusion et des perspectives sont données dans la section 4.

2 Préliminaires

Dans ce travail, nous nous plaçons dans un contexte transactionnel. Un contexte transactionnel (cf. tableau 1) est un triplet $\mathcal{B} = (\mathcal{T}, \mathcal{I}, \mathcal{R})$, où $\mathcal{T}, \mathcal{I}, \mathcal{R}$ sont des ensembles finis et non vides. Un élément de $\mathcal{I}$ est appelé *item* (ou motif), un élément de $\mathcal{T}$ est appelé une *transaction* représentée par un identifiant `TID`, et $\mathcal{R}$ est une relation binaire entre $\mathcal{T}$ et $\mathcal{I}$. On note par $i\mathcal{R}t$

1. C'est une implication logique de la forme $X \rightarrow \overline{Y}$, $\overline{X} \rightarrow Y$ et $\overline{X} \rightarrow \overline{Y}$, $\forall X, Y \subseteq \mathcal{I}$, où $\overline{\mathcal{A}} = \neg\mathcal{A} = \mathcal{I} \backslash \mathcal{A}$.

TID	Items	Items globaux	Equivalent binaire
1	ACD	A$\overline{B}$CD$\overline{E}$	10110
2	BCE	$\overline{A}$BC$\overline{D}$E	01101
3	ABCE	ABC$\overline{D}$E	11101
4	BE	$\overline{A}$B$\overline{C}$ $\overline{D}$E	01001
5	ABCE	ABC$\overline{D}$E	11101
6	BCE	$\overline{A}$BC$\overline{D}$E	01101

TAB. 1: Contexte transactionnel $\mathcal{B}$

si un item i est présent dans $t \in \mathcal{T}$. $X \subseteq \mathcal{I}$ est appelé itemset, et $Y \subseteq \mathcal{T}$ est appelé tidset.

Les deux applications t et i ci-après définissent la connexion de Galois entre $\mathcal{P}(\mathcal{I})$ et $\mathcal{P}(\mathcal{T})$, où $\mathcal{P}(S)$ est l'ensemble des parties de S (Ganter et Wille, 1999). $t : \mathcal{I} \mapsto \mathcal{T}, t(X) = \{y \in \mathcal{T} | \forall x \in \mathcal{I}, x\mathcal{R}y\}$ et $i : \mathcal{T} \mapsto \mathcal{I}, i(Y) = \{x \in \mathcal{I} | \forall y \in \mathcal{T}, x\mathcal{R}y\}$. L'opérateur de fermeture de la connexion de Galois est la composition $iot(X) = i(t(X)) = \gamma(X)$. X est alors dit fermé si $X = \gamma(X)$, et est générateur s'il est minimal (au sens de l'inclusion) dans sa classe d'équivalence : $[X] = \{Y \subseteq \mathcal{I} | \gamma(Y) = \gamma(X)\}$. Du tableau 1, on a $\gamma(AB) = \gamma(ABC) = \gamma(ABE) = \gamma(ABCE)$. Il apparaît que AB est un générateur et $ABCE$ est un fermé.

Le support de X s'écrit $supp(X) = \frac{|t(X)|}{|\mathcal{T}|} = P(t(X))$, où $|\mathcal{A}|$ désigne la cardinalité de $\mathcal{A}$ et P est la probabilité discrète dans $(\mathcal{T}, \mathcal{P}(\mathcal{T}))$. Par souci de simplification, écrirons $P(t(S)) = P(S), \forall S$. Pour tous $X, Y \subseteq \mathcal{I}$, le support, la confiance, la mesure M_{GK} de $X \to Y$ s'écrivent respectivement $supp(X \cup Y) = \frac{|t(X \cup Y)|}{|\mathcal{T}|}$, $conf(X \to Y) = P(Y|X)$ et

$$M_{GK}(X \to Y) = \begin{cases} \frac{P(Y|X) - P(Y)}{1 - P(Y)}, & \text{si } P(Y|X) > P(Y),\ P(Y) \neq 1 \\ \frac{P(Y|X) - P(Y)}{P(Y)}, & \text{si } X\ P(Y|X) \leq P(Y),\ P(Y) \neq 0. \end{cases} \quad (1)$$

Si $P(Y|X) > P(Y)$, on a $0 < M_{GK}(X \to Y) \leq 1$, alors X et Y sont positivement dépendants (X favorise Y), donc $X \to Y$ pourraît être intéressante. Si $P(Y|X) \leq P(Y)$, on a $-1 \leq M_{GK}(X \to Y) \leq 0$, alors X et Y sont négativement dépendants (X défavorise Y), donc $X \to Y$ n'est pas intéressante, mais $X \to \overline{Y}$ et $Y \to \overline{X}$ pourraient être intéressantes. Une règle $X \to Y$ est valide si $supp(X \cup Y) \geq minsup$ et $M_{GK}(X \to Y) \geq minmgk$, où $minsup$ et $minmgk$ seuils minima de support et de M_{GK} respectivement, choisis arbitrairement dans $]0, 1]$. Les exemples du tableau 2 ci-après illustrent les limites du couple support-confiance, ainsi que l'intérêt de M_{GK}. Les 4 premières colonnes correspondent aux caractéris-

	A	$\neg A$	$\sum$		café	$\neg$café	$\sum$
B	72	18	90	thé	20	5	25
$\neg B$	8	2	10	$\neg$thé	70	5	75
$\sum$	80	20	100	$\sum$	90	10	100

TAB. 2: Tableau de contingence de deux items

tiques de l'achat des produits A et B, les 4 dernières indiquent celles de l'achat du café et du thé. On a $supp(A \cup B) = 0.72$, $P(B|A) = 0.9$, $supp(\text{thé} \cup \text{café}) = 0.2$ et $P(\text{café}|\text{thé}) = 0.8$. Ces valeurs raisonnablement élevées nous invitent à penser que $A \to B$ et `thé` $\to$ `café` sont intéressantes. Cependant, $M_{GK}(A \to B) = \frac{0.9 - 0.9}{1 - 0.9} = 0$ et $M_{GK}(\text{thé} \to \text{café}) = \frac{0.8 - 0.9}{1 - 0.9} < 0$

siginifient respectivement une indépendance entre A et B, et une dépendance négative entre l'achat du thé et du café, donc $A \rightarrow B$ et $\texttt{thé} \rightarrow \texttt{café}$ sont non-intéressantes. D'où l'intérêt majeur de M_{GK} d'élaguer systématiquement les règles non-intéressantes du fait qu'elle prend en considération la probabilité du conséquent, ce n'est pas le cas pour la mesure Confiance.

3 Nouvelles Bases des Règles Positives et Négatives

Considérer à la fois les règles positives et négatives augmente exponentiellement le nombre de règles, ce qui nous va restaurer 8 règles, à savoir $X \rightarrow Y$, $Y \rightarrow X$, $\overline{X} \rightarrow \overline{Y}$, $\overline{Y} \rightarrow \overline{X}$, $X \rightarrow \overline{Y}$, $\overline{X} \rightarrow Y$, $\overline{Y} \rightarrow X$ et $Y \rightarrow \overline{X}$ dont nombreuses sont inintéressantes et redondantes. Heureusement, avec M_{GK}, les types non-intéressants étant systématiquement élagués. Il reste à étudier dans ce qui suit les règles redondantes. Formellement, $r_1 : X_1 \rightarrow Y_1$ est redondante moins informative par rapport à $r_2 : X_2 \rightarrow Y_2$ ssi : (i) $(supp(r_1) = supp(r_2)$ et $M_{GK}(r_1) = M_{GK}(r_2))$, (ii) $(X_2 \subseteq X_1$ et $Y_1 \subset Y_2)$. Pour ce faire, nous avons montré dans (Bemarisika et Totohasina, 2017) que si X favorise Y, alors $X \rightarrow Y$ (resp. $\overline{X} \rightarrow \overline{Y}$) est équivalente à $\overline{Y} \rightarrow \overline{X}$ (resp. à $Y \rightarrow X$). Si X défavorise Y, alors $X \rightarrow \overline{Y}$ (resp. $\overline{X} \rightarrow Y$) est équivalente à $Y \rightarrow \overline{X}$ (resp. $\overline{Y} \rightarrow X$). Grâce à ces propriétés, nous ne retenons que 4 règles, à savoir $X \rightarrow Y$, $\overline{X} \rightarrow \overline{Y}$, $X \rightarrow \overline{Y}$ et $\overline{X} \rightarrow Y$, soit la moitié de l'ensemble, et n'étudions que 2 règles, à savoir $X \rightarrow Y$ et $X \rightarrow \overline{Y}$ (1/4 des candidats), soit un taux de réduction 75% de l'espace.

Ainsi, notre premier modèle concerne la base des règles d'association positives exactes du type $X \rightarrow Y$ telles que $M_{GK}(X \rightarrow Y) = 1$. Les approches comparables sont celles qui se sont développées dans (Guigues et Duquenne, 1986; Feno et al., 2006). Sans rentrer dans les détails de leur calcul, ces approches ne sont pas informatives. En effet, elles sélectionnent les prémisses à partir des pseudo-fermés (Guigues et Duquenne, 1986; Pasquier, 2000) qui retournent les maximaux, icompatibles à la minimalité des prémisses. Pour cela, nous proposons une nouvelle base informative, *EPAR-Exact-Positive-Association-Rules* (cf. théorème 1), en sélectionnant les prémisses (resp. conséquents) à partir des générateurs (resp. fermés).

Théorème 1. *Soit $\mathcal{F}_F$ l'ensemble des fermés fréquents et, pour chaque motif fermé F, $\mathcal{G}_F$ désigne l'ensemble de ses générateurs minimaux, alors*

$$EPAR = \{R : G \rightarrow F \backslash G \mid G \in \mathcal{G}_F \ \wedge \ F \in \mathcal{F}_F\}. \tag{2}$$

Faute de place, toutes les preuves sont ommises, et seront expliquées de façon intuitive. Du tableau 1, nous voyons que A et AC font partie d'une même classe d'équivalence que A est générateur et AC fermé (i.e. $[AC] = \{A, AC\}$), ce qui donne la règle induite $A \rightarrow C$. Puisque $P(C|A) = \frac{|t(A \cup C)|}{|t(A)|} = \frac{|t(A)|}{|t(A)|} = 1 \Leftrightarrow M_{GK}(A \rightarrow C) = 1$, d'où $A \rightarrow C \in EPAR$.

Le deuxième modèle repose sur la base des règles positives approximatives $X \rightarrow Y$ telles que $0 < M_{GK}(X \rightarrow Y) < 1$. Les bases existantes (Guigues et Duquenne, 1986; Feno et al., 2006) utilisent encore les pseudo-fermés, incompatibles à la minimalité des prémisses. A cela, nous proposons une nouvelle base informative, appelée *APAR-Approximate-Positive-Association-Rules* (cf. théorème 2), qui sélectionne les prémisses à partir des générateurs, et les conséquents dans des fermés d'une autre classe d'équivalence contenant le fermé courant.

Théorème 2. *Soit $minmgk \in]0, 1]$. Soient $\mathcal{F}_F$ l'ensemble des fermés fréquents et, pour chaque fermé F, $\mathcal{G}_F$ l'ensemble de ses générateurs, $\widetilde{F}$ un fermé d'une autre classe, alors*

$$APAR = \{r : G_F \rightarrow \widetilde{F} \backslash G_F | (F, \widetilde{F}) \in \mathcal{F}_F^2, F \subset \widetilde{F}, G_F \in \mathcal{G}_F, M_{GK}(r) \geq minmgk\}. \tag{3}$$

Du tableau 1, admettons $minsup = 0.2$ et $minmgk = 0.4$. On trouve, $\gamma(B) = \gamma(BE) \neq \gamma(ABCE)$. Pourtant $BE \subset ABCE$, ce qui donne la règle induite $B \rightarrow ACE$, candidate. Par suite, $P(ACE|B) = 0.4$, d'où $M_{GK}(B \rightarrow ACE) = 0.4 \Rightarrow B \rightarrow ACE \in APAR$.

Le troisième modèle est la base des règles négatives exactes du type $X \rightarrow \overline{Y}$ tel que $M_{GK}(X \rightarrow \overline{Y}) = 1$. Une démarche comparable a été définie dans (Feno et al., 2006). Toutefois, cette approche sélectionne les prémisses à partir de la bordure positive (Mannila et Toivonen, 1997) qui retourne intuitivement les maximaux, incompatibles au concept de minimalité. Nous proposons pour cela une nouvelle base informative, *ENAR-Exact-Negative-Association-Rules* (cf. théorème 3), sélectionnant les prémisses à partir des générateurs de la bordure positive, $Bd^+(\mathcal{F}) = \{X \in \mathcal{F} \mid \nexists Y \supset X, Y \in \mathcal{F}\}$ et les conclusions à partir d'un transversal minimal, noté $\overline{Bd^+(\mathcal{F})}$, où $\mathcal{F}$ est l'ensemble des motifs fréquents d'une base de données.

Théorème 3. *Soit $Bd^+(\mathcal{F})$ l'ensemble de bordure positive et, pour chaque fermé X de $Bd^+(\mathcal{F})$, $\mathcal{G}_X$ désigne l'ensemble de ses générateurs minimaux, alors*

$$ENAR = \{G_X \rightarrow \{\overline{y}\} \mid G_X \in \mathcal{G}_X,\ X \in Bd^+(\mathcal{F}),\ \forall y \in \overline{Bd^+(\mathcal{F})}\}. \quad (4)$$

Par exemple, du Tableau 1, si $minsup = 0.2$, on obtient $Bd^+(\mathcal{F}) = \{ABCE\}$ donnant $\overline{Bd^+(\mathcal{F})} = \{\overline{ABCE}\} = \{D\}$. Et nous avons trouvé que $\gamma(AB) = \gamma(AE) = \gamma(ABCE)$ donnant par la suite des règles induites $AB \rightarrow \overline{D}$ et $AE \rightarrow \overline{D}$. Par suite, $supp(AB\overline{D}) = supp(AB) - supp(ABD) = 0.3 - 0$, d'où $P(\overline{D}|AB) = \frac{0.3}{0.3} = 1$ équivaut à $M_{GK}(AB \rightarrow \overline{D}) = 1$ implique que la règle $AB \rightarrow \overline{D}$ appartient à ENAR. De même pour la règle $AE \rightarrow \overline{D}$.

Le 4e et dernier modèle s'articule sur la base des règles d'association négatives approximatives $X \rightarrow \overline{Y}$ telles que $M_{GK}(X \rightarrow \overline{Y}) < 1$. Une approche naïve (Feno et al., 2006) sélectionne les prémisses et conséquents dans les fermés, donc incompatible au concept d'informativité. Pour cela, nous proposons une nouvelle base informative, appelée *ANAR-Approximate-Negative-Association-Rules* (cf. théorème 4), qui sélectionne les prémisses et conséquents dans les générateurs des fermés incomparables. Il est immédiat, pour tous X et Y, que $M_{GK}(X \rightarrow \overline{Y}) = -M_{GK}(X \rightarrow Y)$ (Bemarisika et Totohasina, 2017). Ce qui nous permet de dériver directement toutes les candidates du type $X \rightarrow \overline{Y}$ dont $M_{GK}(X \rightarrow Y)$ est négative.

Théorème 4. *Soit $minmgk \in]0, 1]$ un minimum de M_{GK}. Soit $\mathcal{F}_F$ l'ensemble des fermés fréquents, pour chaque fermé F, $\mathcal{G}_F$ désigne l'ensemble de ses générateurs minimaux et, $\mathcal{G}_{\widetilde{F}}$ l'ensemble des générateurs d'un autre fermé $\widetilde{F}$ tels que F et $\widetilde{F}$ soient incomparables, alors*

$$ANAR = \{r : G_F \rightarrow \overline{G}_{\widetilde{F}} | (F, \widetilde{F}) \in \mathcal{F}^2_{\widetilde{F}}, F \not\subseteq \widetilde{F}, (G_F, G_{\widetilde{F}}) \in \mathcal{G}_F \times \mathcal{G}_{\widetilde{F}}, M_{GK}(r) \geq minmgk\} \quad (5)$$

Si $minsup = 0.2$, on a, du tableau 1, $[AC] = \{A, AC\}$ et $[BE] = \{B, E, BE\}$. Puisque les deux fermés AC et BE sont non comparables et AC défavorise BE (i.e. AC favorise $\overline{BE}$), ce qui nous donne les règle candidates $A \rightarrow \overline{B}$ et $A \rightarrow \overline{E}$. Admettons par suite $minmgk = 0.2$, on a $M_{GK}(A \rightarrow \overline{B}) = M_{GK}(A \rightarrow \overline{E}) = 0.2 \Rightarrow \{A \rightarrow \overline{B}, A \rightarrow \overline{E}\} \in ANAR$.

Pour énumérer efficacement ces nouvelles bases, nous proposons une méthode en partitionnant les candidats en 2 sous-ensembles disjoints, notés $\mathcal{L}^+_{XY} = \{X, Y \subseteq \mathcal{I} | M_{GK}(X \rightarrow Y) > 0\}$ et $\mathcal{L}^+_{X\overline{Y}} = \{X, Y \subseteq \mathcal{I} | M_{GK}(X \rightarrow \overline{Y}) > 0\}$, qui seront parcourus récursivement, i.e. si X favorise Y, la méthode parcourt uniquement $\mathcal{L}^+_{XY}$, elle n'a plus à parcourir $\mathcal{L}^+_{X\overline{Y}}$, et inversement. Ce qui réduit notablement l'espace de recherche que le théorème 5 l'explique.

Théorème 5. *Toute règle de $\mathcal{L}^+_{XY}$ (resp. $\mathcal{L}^+_{X\overline{Y}}$) est dérivable de $X \rightarrow Y$ (resp. de $X \rightarrow \overline{Y}$).*

A l'intérieur des classes, nous poursuivons encore la réduction de l'espace de recherche, en termes d'élagage des redondances. Cette restriction découle du calcul économique des M_{GK} de toutes les règles d'association candidates que les théorèmes 6 et 7 ci-dessous le montrent.

Théorème 6 (Elagage d'espace dans $\mathcal{L}^{+}_{XY}$)**.** $\forall\ X \rightarrow Y \in \mathcal{L}^{+}_{XY}$ *et tout* $a \subseteq \mathcal{I}$ *tels que* $X \cap \{a\} = \emptyset$*, on a* $M_{GK}(X \rightarrow Y \backslash X) \leq M_{GK}(X \cup \{a\} \rightarrow Y \backslash (X \cup \{a\}))$.

Il est donc inutile de considérer $\widetilde{X} \rightarrow Y \backslash \widetilde{X}, \forall\ \widetilde{X} \subseteq X$, lorsque $X \rightarrow Y \backslash X$ est non valide.

Théorème 7 (Elagage d'espace dans $\mathcal{L}^{+}_{X\overline{Y}}$)**.** $\forall\ X \rightarrow Y \in \mathcal{L}^{+}_{X\overline{Y}}$ *et tout* $a \subseteq \mathcal{I}$ *tels que* $X \cap \{a\} = \emptyset$*, on a* $M_{GK}(X \cup \{a\} \rightarrow \overline{Y} \backslash (X \cup \{a\})) \geq M_{GK}(X \rightarrow \overline{Y} \backslash X)$.

Ce qui relève que, si $X \backslash \overline{\widetilde{Z}} \rightarrow \overline{\widetilde{Z}}$ est valide, alors $X \backslash \overline{Z} \rightarrow \overline{Z}$ est aussi valide, $\forall\ \widetilde{Z} \subseteq Z \subseteq \mathcal{I}$. Ces résultats seront synhétisés dans les algorithmes 1 et 2 ci-après. L'algorithme 1 prend en entrée la famille des motifs fermés fréquents $\mathcal{F}_F$, des générateurs $\mathcal{G}_F$, des bordures positives $Bd^{+}(\mathcal{F})$ et, des $minsup$ et $minmgk$, il retourne la base informative des règles positives et négatives non redondantes, dénotée $\mathcal{B}_{N2R}$. Particulièrement, la fonction NONREDRULES

Algorithm 1 NON-REDUNDANT-RULES

Require: $\mathcal{F}_F$, $\mathcal{G}_F$, $Bd^{+}(\mathcal{F})$, $minsup$ and $minmgk$.
Ensure: $\mathcal{B}_{N2R}$, *Informative Base of Non-Redundant Rules.*

1: $\mathcal{B}_{N2R} = \emptyset$;
2: **for all** $(F \in \mathcal{F}_F)$ **do**
3: **for all** $(G_F \in \mathcal{G}_F)$ **do**
4: **if** $(G_F, F \subseteq \mathcal{L}^{+}_{G_F})$ **then**
5: **if** $(\gamma(G_F) = \gamma(F))$ **then**
6: **if** $(G_F \neq \gamma(G_F)$ && $supp(G_F \cup F) \geq minsup)$ **then**
7: $\mathcal{B}_{N2R} \leftarrow \mathcal{B}_{N2R} \cup \{G_F \rightarrow F \backslash G_F\}$; /* Exact Positive Rules-EPAR */
8: **end if**
9: **else**
10: **for all** $(\widetilde{F} \in \mathcal{F}_F \mid \widetilde{F} \supset F)$ **do**
11: **if** $(supp(G_F \cup \widetilde{F}) \geq minsup$ && $M_{GK}(G_F \rightarrow \widetilde{F} \backslash G_F) \geq minmgk)$ **then**
12: $\mathcal{B}_{N2R} \leftarrow \mathcal{B}_{N2R} \cup \{G_F \rightarrow \widetilde{F} \backslash G_F\}$; /* Approximate Positive Rules-APAR */
13: **end if**
14: **end for**
15: **end if**
16: **else**
17: **for all** $(X \in Bd^{+}(\mathcal{F}))$ **do**
18: $\mathcal{G}_X = generator(X)$;
19: **for all** $(G_X \in \mathcal{G}_X)$ **do**
20: **for all** $(y \in \overline{Bd^{+}(\mathcal{F})})$ **do**
21: **if** $(supp(G_X \cup \{\overline{y}\}) \geq minsup)$ **then**
22: $\mathcal{B}_{N2R} \leftarrow \mathcal{B}_{N2R} \cup \{G_X \rightarrow \{\overline{y}\} \backslash G_X\}$; /* Exact Negative Rules-ENAR */
23: **end if**
24: **end for**
25: **end for**
26: **end for**
27: **for all** $(G_{\widetilde{F}} \in \mathcal{G}_{\widetilde{F}} | F \not\subseteq \widetilde{F} \vee \widetilde{F} \not\subseteq F)$ **do**
28: **if** $(supp(G_F \cup \overline{G_{\widetilde{F}}}) \geq minsup$ && $M_{GK}(G_F \rightarrow \overline{G_{\widetilde{F}}}) \geq minmgk)$ **then**
29: $\mathcal{B}_{N2R} \leftarrow \mathcal{B}_{N2R} \cup \{G_F \rightarrow \overline{G_{\widetilde{F}}} \backslash G_F\}$; /* Approximate Negative Rules-ANAR */
30: **end if**
31: **end for**
32: **end if**
33: **end for**
34: **end for**
35: NONREDRULES($\mathcal{B}_{N2R}$)
36: **return** $\mathcal{B}_{N2R}$

(Algo.1 line 35) permet d'élaguer les règles redondantes. L'Algorithme 2 quant à lui retourne

les règles dérivées. En effet, si $X \rightarrow Y$ (resp. $X \rightarrow \overline{Y}$) est valide, alors $\overline{X} \rightarrow \overline{Y}$ (resp. $\overline{X} \rightarrow Y$) l'est aussi, c'est-ce que nous l'avons expliqué au début de la section 3 ci-dessus.

Algorithm 2 DERIVE-NON-REDUNDANT-RULES

Require: $\mathcal{B}_{N2R}$ *(Base of Non-Redundant Rules), and* $minsup$ *(support threshold).*
Ensure: $\mathcal{D}_{N2R}$ *(Derive Non-Redundant Rules).*
1: $\mathcal{D}_{N2R} = \mathcal{B}_{N2R}$;
2: **for all** $(X \rightarrow Y \backslash X \in \mathcal{B}_{N2R})$ **do**
3: **if** $(supp(\overline{X} \cup \overline{Y}) \geq minsup)$ **then**
4: $\mathcal{D}_{N2R} \leftarrow \mathcal{D}_{N2R} \cup \{\overline{X} \rightarrow \overline{Y} \backslash \overline{X}\}$
5: **end if**
6: **end for**
7: **for all** $(X \rightarrow \overline{Y} \backslash Y \in \mathcal{B}_{N2R})$ **do**
8: **if** $(supp(\overline{X} \cup Y) \geq minsup)$ **then**
9: $\mathcal{D}_{N2R} \leftarrow \mathcal{D}_{N2R} \cup \{\overline{X} \rightarrow Y \backslash \overline{X}\}$
10: **end if**
11: **end for**
12: **return** $\mathcal{D}_{N2R}$

4 Conclusion

Nous avons proposé une nouvelle approche des bases des règles d'association positives et négatives non redondantes et informatives. Malgré son efficacité, nous n'avons pas pu mener les expérimentations, faute de place. Les travaux futurs pourraient porter sur cette question. Une autre perspective serait d'étendre ce travail au problème des règles généralisées.

Références

Agrawal, R. et R. Srikant (1994). Fast Algorithms for Mining Association Rules. In *Proceedings of 20th VLDB Conference*, Santiago, Chile, pp. 487–499.

Bemarisika, P. et A. Totohasina (2017). Optimisation de l'extraction des règles d'association positives et négatives. In *Actes des 24èmes Rencontres de la SFC*, Lyon, France, pp. 25–28.

Feno, D. R., J. Diatta, et A. Totohasina (2006). Galois Lattices and Based for M_{GK}-valid Association Rules. In *Proc. of the Fourth International Conference on Concept Lattices and their Applications, CLA'06*, pp. 127–138.

Ganter, B. et R. Wille (1999). *Formal concept analysis: Mathematical foundations*. Springer Verlag.

Guigues, J. L. et V. Duquenne (1986). Familles minimales d'implications informatives résultant d'un tableau de donnés binaires. In *Mathématiques et Sciences Humaines*, Volume 95, pp. 5–18.

Kryszkiewicz, M. (2002). Concise representations of association rules. In *D. J. Hand, N.M. Adams et R.J. Bolton, éditeurs*, pp. 92–103. Springer Verlag.

Mannila, H. et H. Toivonen (1997). Levelwise Search and Borders of Theories in Knowledge Discovery. In *Data Mining Knowledge Discovery*, pp. 241–258.

Pasquier, N. (2000). *Extraction de Bases pour les Règles d'Association à partir des Itemsets Fermés Fréquents*. Laboratoire d'Informatique (LIMOS).

Summary

In this paper, we propose new bases of positive and negative valid non-redundant and informative association rules in the context of data science.

Alignement de Structures Argumentatives et Discursives par Fouille de Graphes et de Redescriptions.

Laurine Huber*, Yannick Toussaint*
Charlotte Roze*, Mathilde Dargnat**,***, Chloé Braud*

* Université de Lorraine, CNRS, Inria, LORIA (UMR 7503), F-54000 Nancy, France
firstname.lastname@loria.fr
** ATILF, Université de Lorraine, CNRS (UMR 7118), Nancy, France
*** Institut des Sciences Cognitives Marc Jannerod, CNRS (UMR 5304), Bron, France
mathilde.dargnat@univ-lorraine.fr

Résumé. Dans cet article, nous étudions la similarité entre structures argumentatives et discursives en alignant des sous-arbres dans un corpus annoté en RST et en structure argumentative. Contrairement aux travaux précédents, nous ne nous intéressons pas uniquement à un alignement relation à relation, mais à un alignement de sous-structures. À l'aide de méthodes de fouille de données, nous montrons que des similitudes existent entre l'argumentation et le discours. L'annotation multiple du corpus permet également de proposer un alignement entre les structures. De plus, cette approche permet de mettre en évidence les différences d'expressivité des deux formalismes.

1 Introduction

La représentation sémantique d'un texte en Traitement Automatique des Langues se fait à différents niveaux. Le niveau discursif représente sous forme d'un graphe étiqueté les relations sémantico-pragmatiques qui existent entre les segments (i.e clauses ou phrases) d'un texte. Il existe différents cadres théoriques permettant de relier ces segments. L'annotation en RST (Rhetorical Structure Theory, Stede (2008)) représente l'organisation textuelle par le biais de relations de cohérence entre les segments de texte et peut être appliquée à n'importe quel genre textuel. L'annotation de la macro-structure argumentative (notée ARG) proposée par Peldszus et Stede (2013) repose sur les travaux de Freeman (1992). Celle-ci permet de représenter la manière dont les prémisses et les conclusions sont liées au sein d'un texte argumentatif pour former un ensemble cohérent menant à une conclusion principale. Ces deux cadres théoriques ont pour objectif de représenter l'intention de l'auteur par rapport au lecteur mais elles utilisent des ensembles de relations distincts et suivent des règles de construction différentes. Comprendre les liens entre ces deux formalismes permettrait alors de construire des ponts entre les théories et de mieux saisir le pouvoir expressif de chacun d'entre eux. Cet article présente les résultats préliminaires sur l'alignement de ces structures en utilisant la fouille de graphes et la fouille de redescriptions. Nous proposons d'appliquer la fouille de redescriptions sur un corpus de textes annotés en ARG et en RST. Chacune des annotations nous permet de

construire un arbre initial dont nous extrayons les sous-arbres. Ces sous-arbres permettent de construire une vue ARG et une vue RST, e.g. des tables représentant la relation binaire entre l'ensemble des textes et l'ensemble des attributs ARG ou RST respectivement.

Peldszus et Stede (2016) ont montré qu'un alignement deux à deux entre les relations n'est pas toujours possible. Certaines non-correspondances sont expliquées par des différences d'expressivité entre les deux types d'annotation. Nous proposons donc d'étudier un alignement entre sous-structures permettant ainsi la combinaison de relations.

Cabrio et al. (2013) proposent une étude manuelle pour la mise en correspondance de schémas argumentatifs (selon les *Argumentation Schemes* (AS) de Walton et al. (2008)) avec les relations du *Penn Discourse TreeBank* (PDTB) de Prasad et al. (2008) : des correspondances sont conjecturées et 2 annotateurs évaluent leur pertinence. Le coefficient Cohenś Kappa calculé sur les annotations montre un accord significatif qui valide leur hypothèse. Leur approche est basée sur une appréciation et une annotation humaine. Contrairement à eux, nous proposons une approche automatique basée sur la fouille de données. C'est à notre connaissance la première approche automatique et systématique pour l'alignement de structures ARG et RST.

2 Méthodologie

Le processus en trois étapes vise à trouver un alignement exhaustif et systématique dans le corpus entre des "parties" des représentations RST et des "parties" des représentations ARG. Pour toutes les représentations $t \in T$, nous transformons chacune des structures ARG et RST en un arbre A et R (voir Fig. 1) et nous extrayons les sous-arbres $a \in A$ et $r \in R$, produisant ainsi deux ensemble d'attributs distincts. Les deux vues sont ensuite définies par les relations binaires $R_{arg} \subseteq T \times A$ et $R_{rst} \subseteq T \times R$, où $aR_{arg}t$ et $rR_{rst}t$ définissent l'appartenance d'un sous-arbre a à un texte t dans leur représentations ARG et RST respectivement. La fouille de redescriptions permet ensuite d'extraire des paires (q_{Arg}, q_{Rst}) de requêtes, où q_{Arg} est une formule logique construite à partir des attributs de A et q_{Rst} à partir de ceux de R.

Encodage des arbres RST et ARG. Les représentations RST et ARG sont différentes sur plusieurs points (e.g contraintes d'attachement, notion de nucléarité (Stede, 2008)). Cependant, Peldszus et Stede (2016) ont montré que les segments correspondant au noyau principal [1] en RST et la conclusion principale en ARG correspondent dans 85% des textes au même segment. Nous représentons donc pour chaque texte deux arbres initiaux distincts, où la racine (étiquetée `CC`) représente le noyau principal dans la RST et la conclusion dans l'ARG. Pour chaque relation entre la prémisse p et la conclusion c de l'ARG et entre le noyau n et le satellite s de la RST, les relations discursives ou argumentatives correspondent au label de la branche correspondante, et la notion parent-enfant est définie comme suit :

— en ARG, c est le père et p le fils,
— en RST, n est le père et s le fils.

Une particularité existe dans les arbres ARG, où la relation d'`undercut` est dirigée vers un arc et non vers un noeud. Nous nous inspirons de Wachsmuth et al. (2017) et la modifions afin que la cible de la relation `undercut` devienne la prémisse de la relation. La Fig. 1 illustre les annotations ARG et RST et leurs arbres initiaux.

1. L'unité la plus centrale. (Stede, 2008)

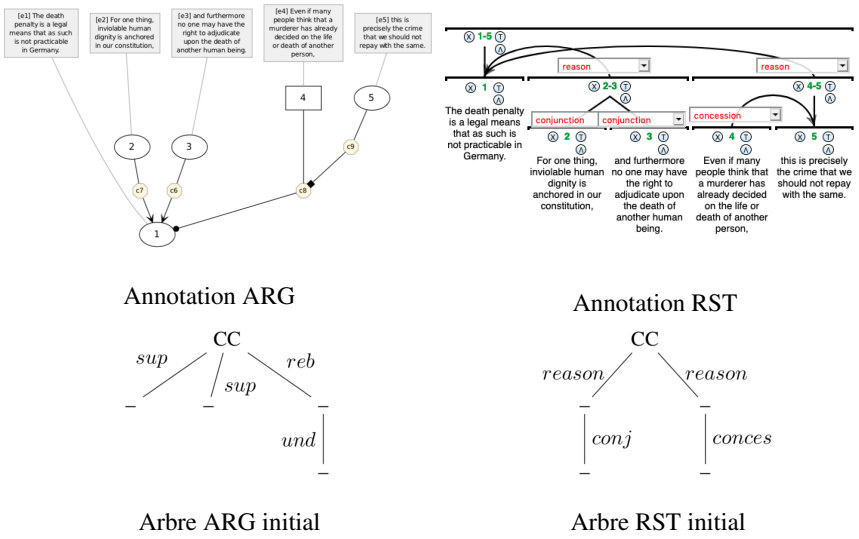


FIG. 1: Texte *micro_b006* annoté en ARG et en RST et arbres initiaux associés.

Construction des deux vues. Nous extrayons indépendamment les sous-arbres de l'ensemble des arbres RST et ARG. Nous donnons à chacun un identifiant unique, il devient alors un attribut : un texte t possède un attribut $a \in A$ et $r \in R$ si l'arbre correspondant à l'attribut est un sous-arbre de l'arbre initial A de l'ARG ou R de la RST respectivement. Chaque texte possède donc un ensemble d'attributs $a \in A$ et un ensemble d'attributs $r \in R$. Les sous-arbres sont extraits avec gSpan (*Graph-Based Substructure Pattern Mining*) (Yan et Han, 2002), un algorithme qui, étant donné un ensemble de graphes $\mathbb{GS}$, en extrait les sous-graphes fréquents. De façon informelle, un graphe h est un *sous-graphe* de g si h est contenu dans g, et h est *fréquent* si au moins s graphes de $\mathbb{GS}$ contiennent h, s étant un seuil fixé par l'utilisateur [2]. À partir de la sortie de GSpan, nous représentons les attributs booléens dans deux tables binaires, où les lignes correspondent aux textes et les colonnes correspondent aux attributs.

Fouille de redescriptions. En analyse de données, la fouille de redescriptions (Galbrun et Miettinen, 2017) consiste à trouver deux caractérisations différentes d'un même ensemble d'objets (i.e. textes dans notre contexte). L'objectif est de trouver deux expressions $q1$ et $q2$ (des requêtes), où $q1$ et $q2$ sont des formules logiques constituées à partir des attributs $a \in A$ et $r \in R$ respectivement, et où l'ensemble de textes décrits par $q1$ et $q2$ est suffisamment similaire. Cette similarité est mesurée par l'indice de Jaccard J(q1,q2) = $\frac{supp(q1 \wedge q2)}{supp(q1 \vee q2)}$ où $supp(q)$ est le nombre de textes pour lesquels q est vraie. L'indice de Jaccard représente la manière dont se recoupent les objets vrais dans $q1$ et ceux vrais dans $q2$.

2. Nous fixons ce seuil à 2 pour considérer tous les sous-arbres qui sont présents dans au moins deux textes.

La stratégie d'exploration de ReReMi est basée sur la mise à jour atomique. Premièrement, l'algorithme calcule l'indice de Jaccard pour toutes les paires possibles de requêtes atomiques, autrement dit toutes les redescriptions qui peuvent être construites à partir d'un attribut de chaque vue. Les n meilleures paires sont conservées. A partir de ces paires, l'algorithme applique des opérations d'addition sur l'une et l'autre des requêtes afin d'améliorer les redescriptions candidates jusqu'à ce que l'indice de Jaccard ne puisse plus être amélioré. Nous utilisons l'algorithme ReReMi (Galbrun et Miettinen, 2012) implémenté dans l'outil Siren (Galbrun et Miettinen, 2018) avec les paramètres prédéfinis par l'outil. Les conjonctions et les disjonctions sont autorisées dans les requêtes mais la longueur des requêtes est limitée à quatre attributs.

3 Expérimentation

Données. Le corpus est composé de 112 textes répondant à une question controversée (e.g *"Should Germany introduce the death penalty ?"*). Les arbres RST sont annotés avec 28 relations distinctes, chacun des textes ayant entre 2 et 12 relations par arbre. Les relations RST les plus fréquentes sont : `reason` (180), `concession` (65), `list` (63), `conjunction` (44), `antithesis` (32), `elaboration` (37), et `cause` (20). Cinq relations distinctes servent à annoter les arbres ARG, et chaque texte possède entre 2 et 9 relations. Les relations ARG les plus fréquentes sont : `support` (263), `rebut` (108) et `undercut` (63). Avec gSpan, nous avons extrait 311 sous-arbres RST et 98 sous-arbres ARG. L'attribut RST le plus fréquent apparaît dans 105 textes alors que l'attribut ARG le plus fréquent apparaît dans 94 textes. Seuls 22 attributs RST sont partagés par plus de 10 textes, et 18 attributs ARG sont partagés par plus de 13 textes.

	q1	q2	J(q1,q2)	# texts
$Rd1$	`a57`	`r40` ∨ `r65` ∨ `r123`	0.691	54
$Rd2$	`a58`	`r61` ∨ `r119` ∨ `r125`	0.351	13
$Rd3$	`a23` ∨ `a59`	`r125`	0.3	8

TAB. 1: 3 des 31 redescriptions. `aX` et `rX` correspondent resp. aux sous-arbres ARG et RST.

Résultats. Pour des raisons de place, nous ne commenterons que 3 des 35 redescriptions obtenues (voir Tab. 1). Nous choisissons $Rd1$ car elle a l'indice de Jaccard le plus haut, $Rd2$ car c'est une spécialisation de $Rd1$ et $Rd3$ car celle-ci contient une disjonction du côté ARG. Les attributs de chacune des redescriptions sont représentés en Fig. 2, Fig. 3 et Fig. 4.

Les 54 textes décrits par $Rd1$ contiennent tous l'attribut `a57` en ARG, mais la disjonction côté RST met au jour une différence de granularité entre les deux formalismes. Plus précisément, parmi les 54 arbres qui ont `a57`, 30 contiennent `r123`, 22 contiennent `r65`, 2 contiennent `r40` dans leur représentation RST. En d'autres termes, près de la moitié du corpus contient deux relations de `support` dirigées vers la CC, et ces textes ont dans leur arbre ARG soit une relation `reason` avec 2 éléments en `list`, soit deux relations `reason`, soit une `motivation` dirigée vers la `CC`. Les objets décrits par $Rd2$ et $Rd3$ sont aussi décrits par $Rd1$ donc $Rd2$ et $Rd3$ peuvent être vus comme des spécialisations de $Rd1$. $Rd2$ peut être lue de la même manière que $Rd1$: parmi les 23 textes qui contiennent `a58`, 13 sont alignés avec soit `r61` (3), soit `r119` (3), soit `r125` (7).

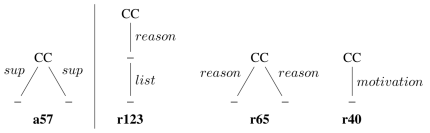


FIG. 2: Sous-arbres correspondants aux attributs de $Rd1$

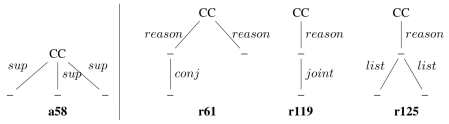


FIG. 3: Sous-arbres correspondants aux attributs de $Rd2$

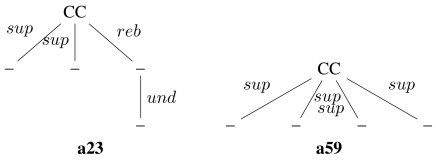


FIG. 4: Sous-arbres correspondants aux attributs de $Rd3$

Ces deux premières redescriptions confirment qu'une relation de `support` en ARG peut correspondre à différentes relations en RST (Peldszus et Stede (2016)). Notre approche permet cependant de révéler les attributs RST qui apparaissent en vis-à-vis d'une structure ARG et d'en quantifier le nombre d'occurences. Contrairement à $Rd1$ et $Rd2$, la disjonction du côté de l'ARG dans $Rd3$ suggère que l'attribut `r125` (qui apparaît dans 8 textes) peut être aligné avec deux différentes structures ARG : `a59` (dans 2 textes), et `a23` (dans 5 textes). Bien qu'ayant un faible indice de Jaccard, cette redescription est pertinente si il existe une relation `undercut` entre X et Y, et une `rebut` entre Y et la CC, alors X est en fait un argument en `support` de la CC. Néanmoins, l'approche engendre aussi des alignements incohérents dus à l'anonymisation des segments de texte et aux paramètres de ReRemi.

4 Conclusion

La fouille de redescriptions peut être appliquée sur des sous-arbres pour aligner différentes structures textuelles. Ce processus automatique vise à proposer une comparaison systématique de différents formalismes. Appliquée à un corpus annoté en ARG et RST, cette expérience préliminaire permet de mettre en exergue des différences de granularité et d'encodage entre les formalismes. Les prochains travaux doivent permettre de préserver l'ancrage sur les segments textuels et d'étendre les expériences à d'autres formalismes (par exemple la SDRT).

Références

Cabrio, E., S. Tonelli, et S. Villata (2013). From Discourse Analysis to Argumentation Schemes and Back : Relations and Differences. In D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, J. Leite, T. C. Son, P. Torroni,

L. van der Torre, et S. Woltran (Eds.), *Computational Logic in Multi-Agent Systems*, Volume 8143, pp. 1–17. Berlin, Heidelberg : Springer.

Freeman, J. B. (1992). *Dialectics and the Macrostructure of Argument*. Berlin : Foris.

Galbrun, E. et P. Miettinen (2012). From black and white to full color : extending redescription mining outside the Boolean world. *Statistical Analysis and Data Mining : The ASA Data Science Journal 5*(4), 284–303.

Galbrun, E. et P. Miettinen (2017). *Redescription Mining*. Springer International Publishing.

Galbrun, E. et P. Miettinen (2018). Mining redescriptions with Siren. *ACM Transactions on Knowledge Discovery from Data (TKDD) 12*(1), 6 :1–6 :30.

Peldszus, A. et M. Stede (2013). From Argument Diagrams to Argumentation Mining in Texts : A Survey. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI) 7*(1), 1–31.

Peldszus, A. et M. Stede (2016). Rhetorical structure and argumentation structure in monologue text. In *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, Berlin, Germany, pp. 103–112. Association for Computational Linguistics.

Prasad, R., N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. Joshi, et B. Webber (2008). The Penn discourse treebank 2.0. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Stede, M. (2008). Rst revisited : Disentangling nuclearity. *'Subordination' versus 'Coordination' in Sentence and Text*, 33–59.

Wachsmuth, H., G. Da San Martino, D. Kiesel, et B. Stein (2017). The impact of modeling overall argumentation with tree kernels. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, pp. 2379–2389. Association for Computational Linguistics.

Walton, D., C. Reed, et F. Macagno (2008). *Argumentation Schemes*. Cambridge : Cambridge University Press.

Yan, X. et J. Han (2002). gSpan : graph-based substructure pattern mining. In *2002 IEEE International Conference on Data Mining, 2002. Proceedings.*, Maebashi City, Japan, pp. 721–724. IEEE.

Summary

In this paper, we investigate similarities between discourse and argumentation structures by aligning subtrees in a corpus containing both annotations. Contrary to previous works, we focus on comparing sub-structures and not only relation matches. Using data mining techniques, we show that discourse and argumentation most often align well, and the double annotation allows to derive a mapping between structures. Moreover, this approach enables the study of similarities between discourse structures and differences in their expressive power.

Construction de variables pour la classification par échantillonnage de motifs

Lamine Diop[*,**], Cheikh Talibouya Diop[**], Arnaud Giacometti[*], Dominique Li[*], Arnaud Soulet[*]

[*]Université de Tours, France
{arnaud.giacometti, dominique.li, arnaud.soulet}@univ-tours.fr
[**]Université Gaston Berger de Saint-Louis, Sénégal
{diop.lamine3,cheikh-talibouya.diop}@ugb.edu.sn

Résumé. La découverte de motifs est une méthode intéressante pour extraire des variables représentatives d'un jeu de données à des fins de classification. Il est possible d'obtenir un nombre raisonnable de motifs complémentaires qui décrivent bien le jeu de données en recourant à l'échantillonnage de motifs. Cette technique récente consiste à tirer aléatoirement des motifs proportionnellement à leur support. Cet article synthétise nos résultats concernant la construction de variables par échantillonnage pour les itemsets et les sous-séquences. Nous montrons l'importance de la norme pour focaliser l'échantillonnage sur les motifs les plus représentatifs et ainsi, améliorer la précision des classifieurs.

1 Introduction

Ces dernières années, de nouvelles méthodes de classification très performantes, comme les réseaux de neurones (LeCun et al., 2015), ont été mises en oeuvre pour traiter les données complexes comme le texte ou les images. Néanmoins, ces méthodes sont moins adaptées aux données symboliques très structurées et leur interprétabilité est plus ardue. Pour de telles données, il peut s'avérer judicieux de construire des variables binaires en amont de l'apprentissage (Liu et Yu, 2005). Plus précisément, on dispose d'un jeu de données $\mathcal{D}$ dont chaque instance $d \in \mathcal{D}$ est issue d'un langage $\mathcal{L}$ muni d'une relation de spécialisation $\preceq$ (Mitchell, 1982; Mannila et Toivonen, 1997). Plus un motif est spécifique, moins il couvre des instances du jeux de données. Pour construire un classifieur, il est alors tentant de choisir, comme variables, les motifs du langage $\mathcal{L}$ apparaissant suffisamment souvent dans $\mathcal{D}$ pour être représentatifs.

L'extraction de motifs fréquents (Agrawal et al., 1994; Agrawal et Srikant, 1995) vise à énumérer tous les motifs du langage $\mathcal{L}$ qui apparaissent dans une large proportion du jeu de données $\mathcal{D} : \{\varphi \in \mathcal{L} : supp(\varphi, \mathcal{D}) \geq \sigma\}$ (où $supp$ est la proportion d'instances du jeu de données plus spécifiques que φ). Malheureusement, il n'est pas possible d'utiliser directement ces motifs fréquents car ils sont bien trop nombreux et trop redondants si le seuil de support minimal σ est petit. A l'opposé, si σ est trop grand, certaines instances seront peu ou pas décrites. Pour obtenir un nombre raisonnable de motifs complémentaires qui décrivent bien le jeu de données $\mathcal{D}$, il est possible de recourir à l'échantillonnage de motifs (Boley et al., 2011).

L'échantillonnage de motifs selon le support vise à tirer aléatoirement un motif φ parmi le langage $\mathcal{L}$ avec une probabilité $\pi = supp(\varphi, \mathcal{D})/Z$ (où Z est une constante de normalisation) et nous noterons ce tirage : $\varphi \sim \pi(\mathcal{L})$. Avec une telle approche, si un motif φ_1 est 2 fois plus fréquents que φ_2 (i.e., $supp(\varphi_1, \mathcal{D}) = 2 \times supp(\varphi_2, \mathcal{D})$), alors φ_1 aura deux fois plus de chance d'être tiré.

Cet article synthétise nos résultats concernant la construction de variables par échantillonnage pour les itemsets et les sous-séquences. Plus précisément, nous nous sommes intéressés à un verrou majeur de l'échantillonnage de motifs à savoir la malédiction de la longue traine. Dans notre cas, la longue traine du langage $\mathcal{L}$ est l'énorme portion de motifs qui ont une fréquence très faible. Même si le tirage est biaisé pour tirer les motifs les plus fréquents, les motifs rares sont tellement nombreux que les motifs fréquents ne seront quasiment jamais tirés. Ce phénomène s'observe tout particulièrement avec l'explosion combinatoire de $\mathcal{L}$ pour les données complexes comme les séquences. Concrètement, cela signifie que dans ce cas, les variables extraites seront peu utiles avec un risque de sur-apprentissage car elles décrivent peu d'instances. Pour lever ce verrou, nous proposons de ne concentrer le tirage que sur les motifs les plus généraux du langage (i.e., les plus courts et souvent, les plus fréquents). Après quelques définitions et notations, nous présentons l'algorithme d'échantillonnage utilisé pour construire les variables dans la section 3. Ensuite, nous présentons les résultats de cette approche sur différents jeux de données de la littérature pour les itemsets et les sous-séquences dans la section 4.

2 Préliminaires

Nous disposons d'un ensemble de motifs, appelé langage $\mathcal{L}$, muni d'une relation de spécialisation $\preceq$. Plus précisément, cette relation de spécialisation est un ordre partiel où $\varphi_1 \preceq \varphi_2$ signifie que φ_1 est plus général que φ_2 (de manière duale, φ_2 est plus spécifique que φ_1). Par exemple, pour un ensemble de littéraux $\mathcal{I}$, le langage des itemsets correspond à $2^{\mathcal{I}}$ et la relation d'inclusion est une relation de spécialisation. φ_2 est une spécialisation immédiate de φ_1, dénoté par $\varphi_1 \dot{\prec} \varphi_2$, (inversement, φ_1 est une généralisation immédiate de φ_2) s'il n'existe pas de motif φ' tel que $\varphi_1 \prec \varphi' \prec \varphi_2$. Si un motif φ ne possède pas de généralisation immédiate, sa norme est 0 et on note : $||\varphi|| = 0$. Sinon, la norme d'un motif est égale à la plus petite norme de ses généralisations immédiates plus 1 : $||\varphi|| = \arg\min_{\varphi' \dot{\prec} \varphi} ||\varphi'|| + 1$ s'il existe un motif $\varphi' \prec \varphi$. Par exemple, pour les itemsets, l'ensemble vide ne dispose pas de généralisation et sa norme est 0. Plus généralement, la norme d'un itemset correspond à sa longueur.

L'ensemble de tous les motifs de norme inférieure à M est dénotée par $\mathcal{L}_{\leq M} = \{\varphi \in \mathcal{L} : ||\varphi|| \leq M\}$. Ainsi, pour éviter de tirer les motifs trop rares et trop spécifiques de la longue traine, notre proposition et problème consiste à tirer des motifs de $\mathcal{L}_{\leq M}$ selon leur support : $\varphi \sim supp(\mathcal{L}_{\leq M}, \mathcal{D})$.

3 Construction de variables par échantillonnage

La sous-section 3.1 présente un algorithme pour tirer aléatoirement un motif de norme inférieure à M proportionnellement au support. Ensuite, la sous-section 3.2 détaille la construction du classifieur à partir de ces variables extraites.

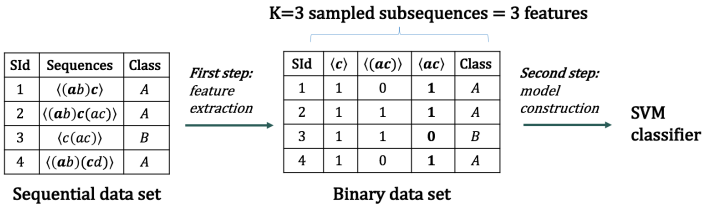


FIG. 1 – *Classification en deux phases*

3.1 Echantillonnage de motifs avec une contrainte de norme

Nous reformulons l'algorithme de tirage en deux étapes de Boley et al. (2011) dédié à la mesure d'aire pour la fréquence $supp(\varphi, \mathcal{D})$ mais en ajoutant une contrainte de la norme maximale M afin de tirer les motifs au sein de $\mathcal{L}_{\leq M}$. L'algorithme 1 s'applique sur une base de données $\mathcal{D}$ et une norme maximale M. *Etape 1 :* Nous calculons la pondération de chaque instance d de $\mathcal{D}$ en comptant le nombre de motifs φ plus généraux que d et de norme inférieure à M (ligne 1). Ensuite, une instance d est tirée au hasard proportionnellement à son poids $\omega(d)$ (ligne 2). *Etape 2 :* La ligne 3 détaille le poids $\omega(d)$ afin de connaître pour chaque norme ℓ, le poids de tous les motifs de d qui ont exactement cette norme ℓ. La ligne 4 utilise cette distribution pour tirer au hasard une norme ℓ proportionnellement à $\omega_\ell(d)$. Enfin, il suffit de retourner un motif tiré uniformément parmi ceux qui ont une norme ℓ (ligne 5).

Algorithm 1 Tirage d'un motif dans une base de données avec une norme inférieure à M

Input: Une base de données $\mathcal{D}$ et un seuil de norme maximale M
Output: Un motif tiré aléatoirement $\varphi \sim supp(\mathcal{L}_{\leq M}, \mathcal{D})$
// Etape 1 : tirage d'une instance
1: Soient les poids ω définis par $\omega(d) := |\{\varphi \in \mathcal{L} : \varphi \preceq d \wedge ||\varphi|| \leq M\}|$ pour tout $d \in \mathcal{D}$
2: Tirer une instance d proportionnellement à ω : $d \sim \omega(\mathcal{D})$
// Etape 2 : tirage d'un motif
3: Soient les poids définis par $\omega_\ell(d) := |\{\varphi \in \mathcal{L} : \varphi \preceq d \wedge ||\varphi|| = \ell\}|$ pour tout $\ell \in [0..M]$
4: Tirer une norme ℓ proportionnellement à $\omega_\ell(d)$: $\ell \sim \omega_{[0..M]}(d)$
5: **return** un motif de norme ℓ de d : $\varphi \sim unif(\{\varphi \preceq d : ||\varphi|| = \ell\})$

Cet algorithme générique peut être implémenté efficacement pour différents langages $\mathcal{L}$ comme les itemsets (Diop et al., 2019) ou les sous-séquences (Diop et al., 2018). L'idée clé est de dénombrer $\omega_\ell(d)$ sans énumérer tous les motifs au sein de l'instance d qui ont une norme ℓ.

3.2 Utilisation de motifs pour la classification

Cette sous-section décrit une méthode en 2 phases pour utiliser un échantillon de motifs afin de construire un classifieur (Boley et al., 2011). Pour un jeu de données $\mathcal{D}$ et un échantillon de motifs $F = \{f_1, \ldots, f_K\}$, la méthode de classification en deux phases consiste à transformer $\mathcal{D}$ en un jeu de données étiquetées de variables binaires $\mathcal{D}^b$ à partir de F et appliquer un algorithme de classification sur $\mathcal{D}^b$. Plus précisément :

TAB. 1 – *Précision des classifieurs avec 1000 itemsets en fonction de M*

Jeu de données	M=1	M=2	M=3	M=4	M=5	M=6	M=7	Best
Auto	0.820	**0.842**	0.835	0.827	0.821	0.821	0.808	0.842
Congres	**0.952**	0.938	0.941	0.945	0.941	0.938	0.937	0.952
CylBands	0.757	0.762	**0.771**	**0.771**	0.770	0.770	0.691	0.771
Hepatitis	**0.863**	0.820	0.807	0.801	0.799	0.795	0.792	0.863
HorseColic	**0.803**	0.778	0.773	0.765	0.760	0.757	0.737	0.803
Ionosphere	0.870	0.882	0.895	**0.900**	0.891	0.865	0.849	0.900
Mushroom	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	0.999	1.000
Soybean-large	**0.930**	0.928	0.925	0.908	0.889	0.874	0.856	0.930
Waveform	**0.796**	0.746	0.733	0.731	0.726	0.722	0.720	0.796
Zoo	0.966	**0.969**	0.964	0.963	0.968	0.967	0.962	0.969
Moyenne	0.876	0.867	0.864	0.861	0.857	0.851	0.835	0.883

1. **Binarisation :** Pour chaque instance de $d \in \mathcal{D}$ étiquetée c_i, $\mathcal{D}^b$ contient un tuple t_i de $K+1$ valeurs où $t_i[j] = 1$ si $f_j \preceq d_i$ (0 sinon) pour $j \in [1..K]$, et $t_i[K+1] = c_i$. La figure 1 illustre ce principe où un jeu de données séquentielles avec 4 séquences dans les classes A ou B est transformé en un jeu de données binaires en utilisant un échantillon de $K = 3$ sous-séquences. Par exemple, si on considère la sous-séquence $\langle ac \rangle$, nous avons $t_1[3] = 1$ parce que la séquence s_1 contient la sous-séquence $\langle ac \rangle$, alors que $t_3[3] = 0$ car la séquence s_3 ne contient pas la sous-séquence $\langle ac \rangle$.
2. **Classification :** Nous créons un classifieur C à partir du jeu de données d'apprentissage transformé $\mathcal{D}^b$ en utilisant une méthode qui fonctionne sur des variables booléennes (par exemple, un arbre de décision, un réseaux de neurones, etc). Notez que dans nos expériences, nous utilisons l'algorithme SMO fourni par Weka 3.8 et ses options par défaut pour construire des classifieurs SVM.

Afin de prédire la classe d'une instance non-étiquetée, nous transformons d'abord cette séquence en un vecteur binaire en utilisant les K motifs de l'échantillon F. Ensuite, nous utilisons le classificateur SVM C pour prédire la classe de la séquence précédemment transformée en vecteur binaire.

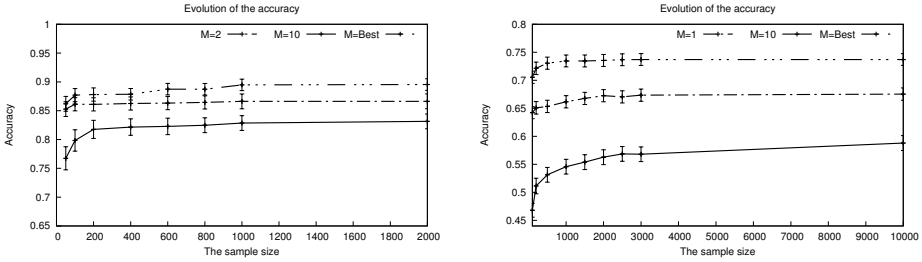
4 Expérimentations

Cette section évalue la qualité des classifieurs obtenus à partir d'échantillons de motifs sous contraintes de norme maximale. Nous reportons une partie des expérimentations menées dans Diop et al. (2019) pour les itemsets et dans Diop et al. (2018) pour les séquences. En particulier, ces articles décrivent les caractéristiques des jeux de données, évaluent la rapidité de l'extraction et comparent les approches avec les méthodes de l'état de l'art.

Impact de la norme Les tableaux 1 et 2 reportent respectivement les précisions obtenues avec 1000 itemsets et 10000 séquences sur différents jeux de données en variant la norme maximale M. Les meilleurs scores de précisions sont notés en gras. La contrainte de norme maximale est primordiale pour éviter une chute de la précision. Pour $M = 10$, on constate que la précision est souvent inférieure et ce phénomène s'accentue lorsque M augmente. Cela montre l'intérêt d'utiliser une contrainte de norme pour focaliser l'extraction de variables générales.

TAB. 2 – *Précision des classifieurs avec 10 000 sous-séquences en fonction de M*

Jeu de données	M=1	M=2	M=3	M=5	M=7	M=10	Best
aslbu	**0.649**	0.601	0.608	0.539	0.396	0.373	0.649
aslgt	0.668	**0.688**	0.680	0.634	0.505	0.364	0.688
auslan	0.230	0.250	0.320	0.320	**0.330**	**0.330**	0.330
blocks	0.857	**1.000**	0.995	0.995	0.995	0.995	1.000
context	**0.984**	0.984	0.971	0.975	0.967	0.959	0.984
pioneer	**1.000**	0.975	0.969	0.858	0.691	0.656	1.000
skater	0.883	0.930	**0.944**	0.919	0.889	0.874	0.944
speed	0.257	0.281	0.306	**0.366**	0.326	0.301	0.366
reuters	**0.949**	0.901	0.765	0.531	0.523	0.519	0.949
Moyenne	0.720	**0.734**	0.729	0.682	0.625	0.597	0.734



(a) itemsets (b) sous-séquences

FIG. 2 – *Comparaison de la précision des classifieurs*

Ensuite, il est clair que certains jeux de données bénéficient de motifs avec une norme strictement supérieure à 1. Ce phénomène est plus net avec les séquences car la taille de l'échantillon est plus grand et surtout, il faut au moins 2 items pour capturer la notion de séquentialité.

Impact de la taille de l'échantillon La figure 4 reporte l'évolution de la précision moyenne pour différentes normes maximales M selon la taille de l'échantillon aussi bien pour les itemsets (à gauche) que pour les sous-séquences (à droite). Aussi bien pour les itemsets que pour les sous-séquences, la précision s'améliore avec le nombre de motifs extraits. De manière intéressante, la précision augmente très rapidement avec les premiers motifs extraits. Clairement, cette progression est plus fulgurante avec les itemsets car les phénomènes à capturer sont probablement moins complexes.

5 Conclusion

Nos expériences de classification nous ont permis de tirer plusieurs leçons sur la construction de variables par échantillonnage. La norme est efficace pour éviter de concentrer le tirage sur la longue traine et ainsi, extraire des variables représentatives. Il est cependant pertinent de considérer des motifs dont la norme est supérieure à 1 notamment pour décrire les relations fines au sein des langages structurés. De manière intéressante, si la qualité de la prédiction

augmente avec le nombre de variables, un échantillon de taille réduite (quelques milliers) est déjà très efficace. Enfin, un langage plus complexe nécessitera des échantillons plus gros. Plusieurs pistes d'améliorations sont envisagées. Nous pourrions contrôler la norme des motifs en les tirant suivant une autre mesure (par exemple, le support multiplié par un facteur à décroissance exponentielle). Il serait aussi intéressant d'appliquer cette approche sur d'autres langages comme les graphes ou d'utiliser des mesures de contrastes plutôt que le support pour choisir des motifs spécifiques à une classe. A plus long terme, nous voudrions utiliser directement l'échantillon de motifs pour faire de la classification associative à la CBA (Ma et al., 1998).

Références

Agrawal, R. et R. Srikant (1995). Mining sequential patterns. In *Proc. of ICDE 95*, pp. 3–14.

Agrawal, R., R. Srikant, et al. (1994). Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, Volume 1215, pp. 487–499.

Boley, M., C. Lucchese, D. Paurat, et T. Gärtner (2011). Direct local pattern sampling by efficient two-step random procedures. In *Proc. of the 17th ACM SIGKDD*, pp. 582–590.

Diop, L., C. T. Diop, A. Giacometti, D. Li, et A. Soulet (2018). Sequential pattern sampling with norm constraints. In *2018 IEEE International Conference on Data Mining (ICDM)*, pp. 89–98. IEEE.

Diop, L., C. T. Diop, A. Giacometti, D. Li, et A. Soulet (2019). Echantillonnage de motifs ensemblistes selon une utilité fondée sur la taille. In *ConféreNce sur la Recherche en Informatique et ses Applications, CNRIA'2019*, pp. 104–115.

LeCun, Y., Y. Bengio, et G. Hinton (2015). Deep learning. *nature 521*(7553), 436.

Liu, H. et L. Yu (2005). Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge & Data Engineering* (4), 491–502.

Ma, B. L. W. H. Y., B. Liu, et Y. Hsu (1998). Integrating classification and association rule mining. In *Proceedings of the fourth international conference on knowledge discovery and data mining*, pp. 24–25.

Mannila, H. et H. Toivonen (1997). Levelwise search and borders of theories in knowledge discovery. *Data mining and knowledge discovery 1*(3), 241–258.

Mitchell, T. M. (1982). Generalization as search. *Artificial intelligence 18*(2), 203–226.

Summary

To build a classifier, pattern mining is an interesting method for extracting features representative of a dataset. It is possible to obtain a reasonable number of complementary patterns that describe the dataset by using pattern sampling. This recent technique randomly draws patterns proportionally to their support. This paper summarizes our results concerning feature construction by sampling itemsets or subsequences. We show the importance of the norm to focus sampling on the most representative patterns and thus, improve the accuracy of classifiers.

Classification croisée de données tensorielles

Rafika Boutalbi [*,**] Lazhar Labiod [*], Mohamed Nadif [*]

[*] Lipade, Université de Paris, 75006, France Paris
[**]Trinov
<prénom.nom>@parisdescartes.fr

Résumé. Pour atteindre l'objectif de la classification croisée de données se présentant sous forme d'un tenseur, nous proposons une extension du modèle de *Poisson Latent Block Model*. Les paramètres de ce modèle sont estimés par un algorithme de type *Variationnel EM*. L'évaluation de notre approche est réalisée sur des tenseurs de données réelles.

1 Introduction

La classification croisée (*co-clustering*) est une méthode permettant de regrouper simultanément les lignes et les colonnes d'une matrice de données. Elle conduit de ce fait à une réorganisation des données en blocs homogènes (après permutations appropriées). Le *co-clustering* joue un rôle important dans une grande variété d'applications où les données sont généralement organisées dans des tableaux à double entrée (Govaert et Nadif, 2013). Cependant si on considère l'exemple du *clustering* d'articles, nous pouvons collecter plusieurs informations liées aux articles tels que les termes en commun, les co-auteurs et les citations, qui conduisent naturellement à une représentation tensorielle. L'exploitation d'un tel tenseur permettrait d'améliorer les résultats de clustering d'un des ensembles. Ainsi, deux articles qui partagent un ensemble important de mots en commun, qui ont des auteurs en commun et qui se citent sont très susceptibles de traiter du même sujet. Dans la suite nous nous intéresserons à de tels tenseurs.

Malgré le grand intérêt pour le *co-clustering* et la représentation tensorielle, peu de travaux portent sur le *co-clustering* de tenseurs. Nous pouvons néanmoins citer le travail basé sur l'information Minimum Bregman (MBI) (Banerjee et al., 2005) ou encore la méthode de co-clustering de tenseurs non négatifs GTSC (General Tensor Spectral Co-Clustering)(Wu et al., 2016). Cependant, la majorité des auteurs ne considèrent pas le co-clustering à partir d'un tenseur selon une approche probabiliste mais plutôt selon des méthodes de factorisation tensorielles. Nous présentons dans ce papier un modèle probabiliste *Tensor Latent Block Model* (Tensor LBM) pour le co-clustering de tenseurs s'appuyant sur une simple extension de LBM.

2 Modèle des blocs latents (LBM)

Le modèle des blocs latents (Govaert et Nadif, 2003) en $g \times m$ blocs est défini de la manière suivante. Étant donnée une matrice $\mathbf{X}$ de taille $n \times d$, on suppose qu'il existe un couple de partitions $(\mathbf{z}, \mathbf{w})$ où $\mathbf{z}$ est la partition en g classes sur l'ensemble des lignes I et $\mathbf{w}$ la partition

en m classes sur l'ensemble des colonnes J, tel que chaque élément x_{ij} appartenant au bloc $k\ell$ est généré selon une distribution de probabilitié, où k représente la classe de la ligne i, tandis que ℓ représente la classe de la colonne j. La partition $\mathbf{z}$ peut être représentée par un vecteur de labels ou par $\mathbf{z} = (z_{ik})$ de taille $n \times g$ où $z_{ik} = 1$ si la ligne i appartient à la classe k, et $z_{ik} = 0$ sinon. De la même manière, la partition $\mathbf{w}$ peut être représentée par un vecteur de labels ou par une matrice de classification des colonnes $\mathbf{w} = (w_{j\ell})$ de taille $d \times m$ où $w_{j\ell} = 1$ si la colonne j appartient à la classe ℓ, et $w_{j\ell} = 0$ sinon. Sous l'hypothèse d'indépendance $p(\mathbf{z}, \mathbf{w}) = p(\mathbf{z})p(\mathbf{w})$ et en notant $\mathcal{Z}$ et $\mathcal{W}$ les ensembles de toutes les partitions possibles $\mathbf{z}$ et $\mathbf{w}$, la vraisemblance des données observées s'écrit :

$$f(\mathbf{X}; \mathbf{\Omega}) \quad = \quad \sum_{(\mathbf{z},\mathbf{w}) \in \mathcal{Z} \times \mathcal{W}} \prod_{i,k} \pi_k^{z_{ik}} \prod_{j,\ell} \rho_\ell^{w_{j\ell}} \times \prod_{i,j,k,\ell} \left(\Phi(x_{ij}; \lambda_{k\ell})\right)^{z_{ik} w_{j\ell}}, \tag{1}$$

où $\mathbf{\Omega} = (\pi, \rho, \lambda)$ correspond aux paramètres inconnus de LBM avec $\pi = (\pi_1, \ldots, \pi_g)$ et $\rho = (\rho_1, \ldots, \rho_m)$ où $(\pi_k = p(z_{ik} = 1), k = 1, \ldots, g)$, $(\rho_\ell = p(w_{j\ell} = 1), \ell = 1, \ldots, m)$ sont les proportions des clusters et $\lambda_{k\ell}$ représentre les paramaètres de la distribution dans le bloc $k\ell$. Avec ce modèle, la log-vraisemblance classifiante prend la forme suivante :

$$L_C(\mathbf{z}, \mathbf{w}, \mathbf{\Omega}) = \sum_{i,k} z_{ik} \log \pi_k + \sum_{j,\ell} w_{j\ell} \log \rho_\ell + \sum_{i,j,k\ell} z_{ik} w_{j\ell} \log(\Phi(x_{ij}; \lambda_{k\ell})). \tag{2}$$

3 Modèle TLBM

Dans ce papier, on s'intéresse à un tenseur présentant une succession de tables de contingences. Nous pouvons donc nous appuyer le modèle des blocs latents Poisssonnien (Poisson LBM) (Govaert et Nadif, 2018) et proposer son extension aux données tensorielles (Poisson Tensor LBM/Poisson TLBM). Dans ce cas le paramètre $\lambda_{k\ell}$ est un vecteur de taille v et comme il dépendera également des marges des lignes et des colonnes i et des j, nous considérons cette paramétrisation $\lambda_{ij}^a = x_{i.}^a x_{.j}^a \sum_{k,\ell} z_{ik} w_{j\ell} \gamma_{k\ell}^a$. Nous assumons une indépendance conditionnelle par bloc, donc $\Phi(\mathbf{x}_{ij}; \lambda_{k\ell})$ est définie par $\prod_{a=1}^{v} \frac{e^{\lambda_{ij}^a} {\lambda_{ij}^a}^{x_{ij}^a}}{x_{ij}^a!}$ et L_C s'écrit

$$L_C(\mathbf{z}, \mathbf{w}, \mathbf{\Omega}) = \sum_{i,k} z_{ik} \log(\pi_k) + \sum_{j,\ell} w_{j\ell} \log \rho_\ell + \sum_{i,j,k,\ell} z_{ik} w_{j\ell} \sum_a \left(-x_{i.}^a x_{.j}^a \gamma_{k\ell}^a + x_{ij}^a \log(\gamma_{k\ell}^a)\right).$$

Pour estimer les parametres de $\mathbf{\Omega}$ nous optons pour une approche s'appuyant sur la maximisation de la log-vraisemblance même si celle-ci requière une approximation variationnelle ; pour plus de détails voir par exemple (Govaert et Nadif, 2005). Cette approche maximise la borne inférieure de la log-vraisemblance : $F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}; \mathbf{\Omega}) = L_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \mathbf{\Omega}) + H(\tilde{\mathbf{z}}) + H(\tilde{\mathbf{w}})$ où $L_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}, \mathbf{\Omega})$ est la vraisemblance classifiante floue avec $\tilde{\mathbf{z}} \in [0,1]^{n \times g}$), $\tilde{\mathbf{w}} \in [0,1]^{d \times m}$, $H(\tilde{\mathbf{z}}) = -\sum_{i,k} \tilde{z}_{ik} \log \tilde{z}_{ik}$ et $H(\tilde{\mathbf{w}}) = -\sum_{j,\ell} \tilde{w}_{j\ell} \log \tilde{w}_{j\ell}$. L'algorithme *Variational EM* sur tenseur (`VEM-T`) alterne deux étapes. Dans L'étape E nous calculons les probabilités a posteriori $\tilde{z}_{ik}$ et $\tilde{w}_{j\ell}$. Dans l'étape M, nous mettons à jour les estimations des paramètres du modèle en maximisant la fonction objectif $F_C(\tilde{\mathbf{z}}, \tilde{\mathbf{w}}; \mathbf{\Omega})$. Les parametres estimés sont :

$$\pi_k = \frac{\sum_i \tilde{z}_{ik}}{n} = \frac{\tilde{z}_{.k}}{n}, \rho_\ell = \frac{\sum_j \tilde{w}_{j\ell}}{d} = \frac{\tilde{w}_{.\ell}}{d}, \gamma_{k\ell}^a = \frac{\sum_{ij} \tilde{z}_{ik} \tilde{w}_{j\ell} x_{ij}^a}{\sum_i \tilde{z}_{ik} x_{i.}^a \sum_j \tilde{w}_{j\ell} x_{.j}^a} = \frac{x_{k\ell}^a}{x_{k.}^a x_{.\ell}^a}$$

où $x_{k.}^a = \sum_i \tilde{z}_{ik} x_{i.}^a, x_{.\ell}^a = \sum_j \tilde{w}_{j\ell} x_{.j}^a, x_{k\ell}^a = \sum_{i,j} \tilde{z}_{ik} \tilde{w}_{j\ell} x_{ij}^a$.

Algorithme 1 : VEM−T

Input : $\mathbf{X}$, g, m.
Initialization : $(\mathbf{z}, \mathbf{w})$ at random, compute $\mathbf{\Omega}$
repeat
  E-Step
  — **Compute** $\tilde{z}_{ik}$ **using** $\tilde{z}_{ik} \propto \pi_k \exp\left(\sum_{j,\ell} \tilde{w}_{j\ell} \log\left(\Phi(\mathbf{x}_{ij}; \lambda_{k\ell})\right)\right)$.
  — **Compute** $\tilde{w}_{j\ell}$ **using** $\tilde{w}_{j\ell} \propto \rho_\ell \exp\left(\sum_{i,k} \tilde{z}_{ik} \log\left(\Phi(\mathbf{x}_{ij}; \lambda_{k\ell})\right)\right)$.
  M-Step
    Update $\mathbf{\Omega}$
until *convergence*;
return z, w, π**,** ρ**,**γ

4 Expériences

Dans cette partie nous illustrons les performences de VEM−T en termes de clustering, en montrant l'impact de la combinaison de differentes informations. Nous utilisons pour cela trois jeux de données textuelles, DBLP1, DBLP2 et PubMed Diabetes [1]. DBLP1$(2223 \times 2223 \times 4)$ et DBLP2 $(1949 \times 1949 \times 4)$ sont construits à partir de DBLP [2] en selectionnant troix revues pour chaque base. Les revues sélectionnées pour DBLP1 sont SIGMOD, STOC, et SIGIR et ceux sélectionnées pour DBLP2 sont Discrete Applied Mathematics, IEEE software, et SIGIR. Ces revues représentent les vraies partitions. Pour la base PubMed Diabetes $(4354 \times 4354 \times 4)$ les articles sont classés en trois catégories portants sur différents types de la maladie du diabète. Nous faisons l'extraction de plusieurs informations à partir de ces trois bases à savoir : (i) La matrice des co-termes dans le titre, (ii) la matrice de co-termes dans le résumé, (iii) la matrice de co-auteurs et (iv) la matrice de citation entre articles.

On compare VEM−T avec Spherical K−means, Itcc (Dhillon et al., 2003), et VEM appliqué à chaque couche a du tenseur et deux autres algorithmes appliqués aux tenseurs à savoir PARAFAC et GTSC (Wu et al., 2016). Notons que PARAFAC est utilisé avec un nombre de rangs égale à 10, K−means est appliqué sur la réduction de la dimension obtenue. Nous réalisons 50 initialisations aléatoires, et nous calculons les mesures l'Indice de Rand ajusté (ARI) et l'information mutuelle normalisée (NMI) pour évaluer la qualité de la partition au regard de la partition connue. Les résultats obtenus sont présentés dans la figure 1, Nous pouvons remarquer qu'en termes de NMI et ARI, VEM−T permet de donner des résultats encourageants et donc de trouver le meilleur compromis en considérant toutes les couches même celles qui sont les plus mélangées.

5 Conclusion

Ce travail présente une extension du modèle des blocs latents Poissonien pour traiter un tenseur de données. L'estimation des paramètres du modèle est réalisée à l'aide d'un algorithme de type *variational EM*. Les exemples traités permettent d'illustrer l'intérêt de l'approche qui peut être également utilisée pour la classification croisée de graphes multiples.

1. https ://linqs.soe.ucsc.edu/data
2. https ://aminer.org/citation

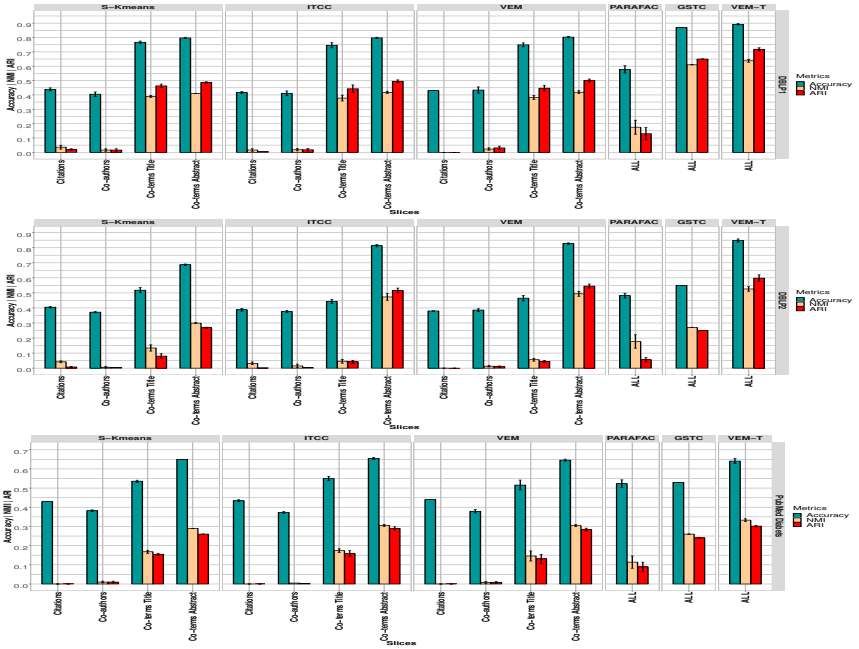


FIG. 1 – *Comparaison entre* `S-Kmeans`, `ITCC`, `VEM`, `PARAFAC`, `GSTC` *et* `VEM-T` *sur* `DBLP1`, `DBLP2` *et* `PubMed`.

Références

Banerjee, A., C. Krumpelman, J. Ghosh, S. Basu, et R. J. Mooney (2005). Model-based overlapping clustering. In *Proceedings of the Eleventh ACM SIGKDD*, pp. 532–537.

Dhillon, I. S., S. Mallela, et D. S. Modha (2003). Information-theoretic co-clustering. In *Proceedings of the Ninth ACM SIGKDD*, pp. 89–98.

Govaert, G. et M. Nadif (2003). Clustering with block mixture models. *Pattern Recognition 36*, 463–473.

Govaert, G. et M. Nadif (2005). An EM algorithm for the block mixture model. *IEEE Transactions on Pattern Analysis and machine intelligence 27*(4), 643–647.

Govaert, G. et M. Nadif (2013). *Co-clustering*. Wiley-IEEE Press.

Govaert, G. et M. Nadif (2018). Mutual information, phi-squared and model-based co-clustering for contingency tables. *Advances in Data Analysis and Classification 12*(3), 455–488.

Wu, T., A. R. Benson, et D. F. Gleich (2016). General tensor spectral co-clustering for higher-order data. In *Advances in Neural Information Processing Systems 29*, pp. 2559–2567.

Summary

To achieve the objective of co-clustering of data in the form of a tensor, we propose an extension of the *Poisson Latent Block Model*. The parameters of this model are estimated by an algorithm of type *variational EM*. The assessment of our approach is performed on real data tensors.

Towards a Constrained Clustering Algorithm Selection

Guilherme Alves, Miguel Couceiro, Amedeo Napoli

Université de Lorraine, CNRS, Inria Nancy G.E., LORIA
{guilherme.alves-da-silva, miguel.couceiro}@inria.fr, amedeo.napoli@loria.fr

Abstract. The success of machine learning approaches to solving real-world problems motivated the plethora of new algorithms. However, it raises the issue of algorithm selection, as there is no algorithm that performs better than all others. Approaches for predicting which algorithms provide the best results for a given problem become useful, especially in the context of building workflows with several algorithms. Domain knowledge (in the form of constraints, preferences) should also be considered and used to guide the process and improve results. In this work, we propose a meta-learning approach that characterizes sets of constraints to decide which constrained clustering algorithm should be employed. We present an empirical study over real datasets using three clustering algorithms (one unsupervised and two semi-supervised), which shows improvements in cluster quality when compared to existing semi-supervised methodologies.

1 Introduction

Novel machine learning algorithms are constantly being proposed. As we do not have a single algorithm that performs better than all other algorithms in all of the cases, it raises the issue of algorithm selection need to design learning workflows. Several approaches for automating algorithm selection become useful to e with this issue (Brazdil et al., 2008). For supervised machine learning tasks, e.g. classification algorithms, plenty of research works are available (Cachada et al., 2017)(Wang et al., 2014). For clustering tasks, which is traditionally an unsupervised task, few methods have been proposed (Pimentel and de Carvalho, 2019).

In some cases, we have domain knowledge available, for instance, a set of constraints. A well-known approach to specify constraints is in the form of instance-level constraints, which are composed of two types: *must-links* and *cannot-links*. A must-link means that two instances should be assigned to the same cluster. A cannot-link implies that the instances cannot belong to the same cluster. Constraints are used to guide the clustering process to improve the quality of the obtained clusters. Research works have extended classical (unsupervised) clustering algorithms to be able to deal with constraints, for instance, COP-KMEANS is the first extension of K-MEANS that can process a set of instance-level constraints (Wagstaff et al., 2001). Some years later, Bilenko et al. (2004) has integrated metric learning proposing the algorithm MPCK-MEANS. Apart from partitional methods, the widely used density-based clustering algorithm DBSCAN has also been extended in the semi-supervised clustering method C-DBSCAN (Ruiz et al., 2007).

Despite the many constrained clustering algorithms proposed in the literature, we are not aware of any contribution towards the automated selection of *constrained* clustering algorithms. To our knowledge, only one research work has proposed a constraint-based metric, Constraint Based Overlap (CBO), to decide which clustering algorithm should be employed (Adam and Blockeel, 2017). CBO is based on how the set of constraints are overlap. The authors argue that CBO captures how difficult is the data to be separated based on a set of constraints. Nevertheless, they do not get improvements when CBO is combined with the metrics based on the unsupervised setting.

Moreover, getting constraints is costly without guarantees of improvements in terms of quality of obtained clusters. Additionally, selecting constraints improperly may deteriorate the constrained clustering algorithm performance (Davidson et al., 2006). In order to cope with this issue, in the active clustering literature, different strategies have been proposed to select informative constraints based on uncertainty (Mallapragada et al., 2008), (Xiong et al., 2014) and k-nearest neighbor graph (Vu et al., 2010).

In this research work, we combine CBO with features based on heuristics for selecting constraints and our proposed feature based on constraints' neighbourhood to predict which constrained clustering algorithm should be used. The main hypothesis of this paper is *that combining CBO with other semi-supervised features along with our proposed feature can help on providing accurate predictions in a constrained clustering algorithm selection*.

This paper is organized as follows. Section 2 introduces the main concepts underlying this work. Section 3 explains our approach. Section 4 presents the experimental setup and discusses the results. The conclusions and future work are discussed in Section 5.

2 Background

A meta-learning system exploits knowledge obtained from previous experiences (Brazdil et al., 2008). In order to represent the previous experiences, we build a dataset named meta-dataset. Each instance of meta-dataset is a meta-instance, which is composed of features extracted from the original dataset and from the associated set of constraints. The extracted features in a meta-dataset are called meta-features. We assign to each meta-instance one class that represents the sequence of recommended algorithms based on criteria of clustering quality. The main problem is to extract meta-features, particularly extract them from a set of constraints. The first proposed meta-feature to characterize the set of constraints is CBO. It summarizes how the clusters overlap based on a given set of constraints by aggregating two components. The first component measures the overlap among short cannot-links and the second measures the overlap among pairs of must-link and cannot-link close to each other.

Let k be a positive integer, let $d(\cdot,\cdot)$ be a distance function and $\mathcal{D} = \{x_i\}_{i=1}^{n}$ a dataset. Given the sets of constraints $ML = \{c_t\}_{t=1}^{m}$ and $CL = \{c_t\}_{t=1}^{m'}$, where $c_t = (x_i, x_j), i \neq j$, let ϵ_i be the distance between instance x_i and the k-th nearest neighbour of x_i. The CBO over $\mathcal{D}$ w.r.t. ML and CL is defined as follows

$$CBO(\mathcal{D}, ML, CL) = \frac{\sum_{c \in CL} score(c) + \sum_{c_i \in CL, c_j \in ML} score(c_i, c_j)}{\sum_{c \in CL \cup ML} score(c) + \sum_{c_i \in CL, c_j \in CL \cup ML} score(c_i, c_j)} \quad (1)$$

Name	Heuristic	Reference
Min-Max	Uncertainty	(Mallapragada et al., 2008)
Ability to Separate between Clusters (ASC)	k-nearest neighbor graph	(Vu et al., 2010)
Normalized Point-based Uncertainty (NPU)	Uncertainty	(Xiong et al., 2014)

TAB. 1: Strategies for selecting constraints employed in this research work.

where $score(c) = s(x_i, x_j)$ and $score(c_i, c_j) = s(x_{i1}, x_{j1}) \times s(x_{i2}, x_{j2})$ for

$$s(x_i, x_j) = \begin{cases} 1 - \frac{d(x_i, x_j)}{\max(\epsilon_i, \epsilon_j)} & \text{if } d(x_i, x_j) \leq \max(\epsilon_i, \epsilon_j) \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

We argue that we can employ the heuristics for selecting constraints in the algorithm selection. In our problem, instead of using the before-mentioned strategies for selecting constraints, we employ the functions underlying the same approaches for estimating how informative is a given set of constraints. We thus select three well-known heuristics available in the literature to be used for constrained clustering algorithm selection. Table 1 shows the selected approaches. Each approach employs a distinct function to estimate the informativeness. For instance, ASC is designed for density-based constrained clustering algorithms as it can treat datasets with different cluster densities. Min-Max employes the radial basis function kernel to estimate the uncertainty of pairs of constraints. The last strategy, NPU, integrates a constrained clustering algorithm to refine iteratively the process of uncertainty estimation.

3 Proposed Approach

In this section, we explain our approach to building a meta-learning system for constrained clustering algorithm selection. We start from the assumption that a well-spread set of constraints can provide holistic information about the dataset in comparison to the density located set of constraints. Therefore, we propose a meta-feature that measures the distribution of shared k-nearest data instances from the set of constraints. In order to do that, we represent this meta-feature using a histogram, not only as a real number. A histogram can express the most knowledge possible about the dataset being characterized. In our case, the histogram characterizes the proportion of reachable data instances from the set of constraints. If constraints are close to each other, most of data instances share the same neighbourhood and the remaining data instances do not be computed in the histogram. On the other hand, if the set of constraints is well distributed in the data space, the overlap of neighbourhoods tend to be minimized and more data instances are considered, increasing the proportion of k-nearest data instances from the set of constraints.

Algorithm 1 presents how the histogram is built. The algorithm only requires the number of neighbours k and the set of constraints. It then builds a histogram of k bars in which each bar represents the proportion of shared k-nearest instances reachable from the set of constraints. We build a histogram for each set of constraints C, i.e., one histogram for ML and another one for CL. Each data instance that belongs to the set of constraints is processed in order to discover its k nearest neighbours. The algorithm adds the i-th neighbour to i-th bar and it counts the data instance only once. For example, Fig. 1 shows two examples of the obtained

histograms over different set of constraints w.r.t. the same dataset. One notes that, in the first example from top to bottom, the data instance b_2 is the shared neighbour of b and e. With our approach we have a global view of the number of data instances that is affected by the attraction power of must-links and the number of data instances is affected by the repulsion power of cannot-links.

We employ the Euclidean distance for extracting meta-features which depend on a distance function. We also compute the same distance between each pair of instances (x_i, x_j). We build three different histograms based on these distances and concatenate them afterwards. The first histogram is computed only from unconstrained pairs, the second is built only from pairs involved in a *must-link*, and the last one considers only *cannot-link* pairs.

Algorithm 1 Constraint neighbourhood-based histogram

```
E ← {}, h ← [0, ..., 0]
for c ∈ C do
    for i ∈ [0, k] do
        for x ∈ c do
            N ← Nearestneighbours(x, i)
            h[i] ← h[i] + |N−E| / n
            E ← E ∪ N
return h
```

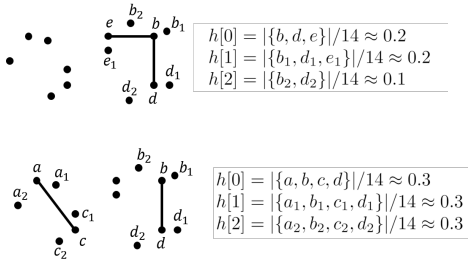


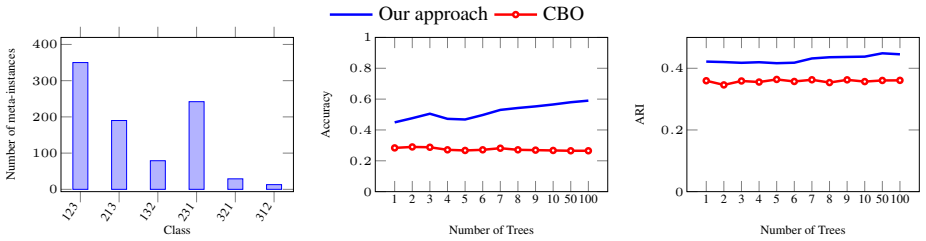
FIG. 1: The constraints arrangement in the dataset changes the obtained histogram ($k = 2$).

4 Experiments and Results

Experimental setup. In order to evaluate our approach, the experiments were conducted using 23 real datasets covering different domains from Open Machine Learning, an open scientific platform for standardizing and sharing datasets and empirical results. They are a subset from the datasets used by Pimentel and de Carvalho (2019) and Adam and Blockeel (2017). All these datasets have labeled data instances, which allows us to run our experiments. For each dataset, 5 different set of constraints were sampled according to a uniform distribution until the number of data instances were reached (0%,25%,50%,75%,100%). We also repeat the execution of each algorithm over each pair (unlabeled dataset, set of constraints) 5 times in order to catch the general behavior of the algorithms in each problem.

We compare the state of the art meta-learning system (that only uses CBO as meta-feature) with our approach, which comprises CBO, Min-Max, ASC, NPU, the constraint neighbourhood-based histogram, and the distance-based histograms. In order to evaluate both predictions, we use the leave-p-out protocol, where p is the number of meta-instances yielded from one dataset. The idea is to avoid sharing information among meta-instances that comes from the same dataset. The pool of considered algorithms were: the constrained clustering algorithms COP-KMEANS (1) and K-MEANS (2), and the traditional clustering algorithm K-MEANS (3).

Following the research works in algorithm selection, we adopted Random Forest (RF) (Breiman, 2001) as meta-learner. Therefore, we run RF over our meta-dataset where meta-instances were composed of the above-mentioned meta-features and were labeled according to Adjusted Rand Index (ARI). For example, given partitions obtained from a dataset and its set of constraints, if we have the following values of ARI: COP-KMEANS = 0.6, MPCK-MEANS



(a) (b) (c)

FIG. 2: (a) Class distribution, (b) Classification assessment, (c) Partition quality assessment.

= 0.7, and K-MEANS = 0.5, the assigned class to the associated meta-instance is "213". It means that the recommended order for employing the available algorithms is MPCK-MEANS, COP-KMEANS, and then K-MEANS. Fig. 2a shows the class distribution. Note that we do not have the best algorithm in all scenarios, which matches with we have mentioned earlier.

Empirical Results. We designed a set of experiments intending to evaluate how would be the variation of accuracy when we change the number of trees of RF. Figure 2b shows the assessment in terms of accuracy of the built meta-model using only CBO and using our approach. We can observe that for the smaller number of trees, the two approaches are competitive, yielding results with minor differences. However, the main advantages of our approach over CBO can be noted at the larger number of trees, as we have more meta-features for describing the same clustering problems.

Furthermore, we can also observe improvements in terms of ARI (see Figure 2c) . ARI is calculated based on the first position indicated in the predicted class. For instance, if the predicted class is "213", it means that algorithm 2 (MPCK-MEANS) is highly recommended and thus we run MPCK-MEANS over the dataset to compute its ARI afterwards. Therefore, the increase in the average of ARI corroborates that our meta-features contribute to a better decision of which clustering algorithm should be employed.

5 Conclusion

In this paper, we proposed an approach for constrained clustering algorithm selection using the set of meta-features: CBO, heuristics for selecting constraints, and our proposed constraint neighbourhood-based histogram. We evaluate our approach over real datasets and we achieved results that indicate improvements with respect to existing state of the art.

This work opens several avenues for future research. Our work could be extended to select the most informative meta-instances. Another interesting directions are to deal with this problem as a learning of ranking task and extend it to the online setting.

References

Adam, A. and H. Blockeel (2017). Constraint-based measure for estimating overlap in clustering. In *Benelux Conference on Machine Learning*, Volume 6, pp. 54–61.

Bilenko, M., S. Basu, and R. J. Mooney (2004). Integrating constraints and metric learning in semi-supervised clustering. In *ICML*, pp. 11. ACM.

Brazdil, P., C. G. Carrier, C. Soares, and R. Vilalta (2008). *Metalearning: Applications to data mining*. Springer Science & Business Media.

Breiman, L. (2001). Random forests. *Machine learning 45*(1), 5–32.

Cachada, M., S. M. Abdulrahman, and P. Brazdil (2017). Combining feature and algorithm hyperparameter selection using some metalearning methods. In *AutoML@PKDD/ECML*, pp. 69–83.

Davidson, I., K. L. Wagstaff, and S. Basu (2006). Measuring constraint-set utility for partitional clustering algorithms. In *PKDD*, pp. 115–126. Springer.

Mallapragada, P. K., R. Jin, and A. K. Jain (2008). Active query selection for semi-supervised clustering. In *ICPR*, pp. 1–4. IEEE.

Pimentel, B. A. and A. C. de Carvalho (2019). A new data characterization for selecting clustering algorithms using meta-learning. *Information Sciences 477*, 203–219.

Ruiz, C., M. Spiliopoulou, and E. Menasalvas (2007). *C-DBSCAN: Density-Based Clustering with Constraints*, Volume 4482 of *LNCS*. Berlin, Heidelberg: Springer.

Vu, V., N. Labroche, and B. Bouchon-Meunier (2010). Boosting Clustering by Active Constraint Selection. In *ECAI*, Lisbon, Portugal.

Wagstaff, K., C. Cardie, S. Rogers, S. Schrödl, et al. (2001). Constrained k-means clustering with background knowledge. In *ICML*, Volume 1, pp. 577–584.

Wang, G., Q. Song, X. Zhang, and K. Zhang (2014). A generic multilabel learning-based classification algorithm recommendation method. *ACM TKDD 9*(1), 7.

Xiong, S., J. Azimi, and X. Z. Fern (2014). Active Learning of Constraints for Semi-Supervised Clustering. *IEEE TKDE 26*(1), 43–54.

Résumé

Le succès des approches d'apprentissage automatique pour résoudre les problèmes du monde réel a motivé une pléthore de nouveaux algorithmes. Cependant, cela soulève le problème de la sélection des algorithmes, puisqu'il n'ya pas un seule algorithme qui soit toujours plus performant que tous les autres. Les approches permettant de prédire quels algorithmes fournissent les meilleurs résultats pour un problème donné deviennent utiles, en particulier dans le cadre des workflows avec plusieurs algorithmes. Les connaissances du domaine (sous forme de contraintes et de préférences) doivent également être prises en compte et utilisées pour guider le processus et pour améliorer les résultats. Dans ce travail, nous proposons une approche de méta-apprentissage qui caractérise des ensembles de contraintes pour décider quel algorithme de clustering contraint doit être utilisé. Nous présentons une étude empirique sur des ensembles de données réels utilisant trois algorithmes de clustering (un non supervisé et deux semi-supervisés) et qui montre l'amélioration de la qualité des clusters obtenus par rapport aux méthodologies semi-supervisées existantes.

Modélisation stochastique et spectrale de l'occupation du sol

Jean François Mari*, Odile Horn**

*Université de Lorraine, Loria, UMR 7503, Vandœuvre-lès-Nancy, F-54506, France
jfmari@loria.fr
** LCOMS,ISEA, Université de Lorraine, 7, rue Marconi, Metz, F- 57070, France
odile.horn@univ-lorraine.fr

Résumé. Nous proposons une approche stochastique pour découvir des items périodiques dans un processus stationnaire. Le passage d'une représentation sous forme d'une série temporelle d'items à une représentation sous forme d'une série temporelle de tenseurs multi-dimensionnels nous permet d'utiliser les techniques de traitement de signaux multi-dimensionnels. A l'aide des coefficients d'auto corrélation croisés, nous montrons sur des données artificielles qu'une analyse spectrale permet de faire apparaître des comportements périodiques.

1 Introduction

L'activité humaine est source de données temporelles de différentes natures : données numériques telles que les prix de l'essence à la pompe ou données catégorielles telles les majorités parlementaires de la cinquième république.

Dans toutes ces séries de données, l'observation de comportements périodiques présente un intérêt pour l'extraction de connaissances. Nous nous plaçons dans le cas d'une source de données stationnaire, c'est à dire dont la statistique sur une fenêtre temporelle glissante est indépendante du temps et dont les réalisations sont des séries temporelles de catégories appelées items.

En ce qui nous concerne, nous nous intéressons aux occupations annuelles d'une parcelle agricole sous la forme des cultures qui y sont portées. Tant que l'ensemble des opportunités ou contraintes économiques et climatiques imposées aux cultivateurs ne changent pas, on peut faire l'hypothèse que les séries de cultures dans les différentes parcelles se ressembleront tout en acceptant une certaine variabilité inhérente à toute activité humaine.

La recherche de périodes dans des séries temporelles d'items est principalement faite à l'aide de méthodes combinatoires Elfeky et al. (2005); Tatavarty et al. (2007); Galbrun et al. (2018). Toutes utilisent des seuils fixés *a priori* pour décider si la répétition d'un comportement est significatif et périodique.

Les séries de données échantillonnées à partir d'un signal réel continu sont traitées par une analyse spectrale des fonctions d'auto corrélation depuis les travaux de Vlachos et al. (2005) et Li et al. (2012).

L'originalité de notre travail est double : d'une part, nous proposons une méthode de traitement de données catégorielles employant les formalismes de l'analyse spectrale. Cela nécessite

une représentation tensorielle de ces données afin d'utiliser les techniques éprouvées d'analyse des signaux numériques.

D'autre part, pour pallier la trop courte durée des séries temporelles de données disponible, nous traitons plusieurs séries d'items simultanément en les considérant comme des échantillons d'une source probabiliste de séries temporelles dont les moments peuvent être calculés efficacement.

Cet article est structuré de la façon suivante. Après avoir fixé le cadre de la modélisation stochastique d'un champ de tenseurs, nous proposons d'utiliser les coefficients d'auto corrélation croisée entre dimensions des tenseurs pour faire apparaître par analyse spectrale des comportements périodiques. Nous décrivons ensuite un générateur de séquences périodiques à l'aide de modèles de Markov cachés (HMM) et utilisons les données produites pour les analyser et extraire leurs périodes. Enfin, dans une conclusion / discussion, nous esquissons une méthode pour dépasser la recherche de périodes sur un seul item en la généralisant à la détection de motifs périodiques impliquant plusieurs items.

2 Processus Stochastique

Considérons une séquence temporelle $x_1, x_2, \dots x_T$ de T items issus d'un ensemble $\mathcal{E} = \{e_1, e_2, \dots e_K\}$ de K différentes catégories comme la série temporelle des T occupations du sol (LU comme *Land Use*), observées aux instants $1, 2, \dots T$ sur une parcelle agricole dans un territoire donné. Chaque LU appartient à l'ensemble $\mathcal{E}$. Les x_t sont les réalisations d'une variable aléatoire $X_t(\omega)$ aux instants t sur une parcelle agricole représentée par ω.

Supposons avoir enquêté une mosaïque parcellaire pour relever toutes ses LU pendant T années. Ces données définissent une matrice M dans laquelle la ligne i représente les T LU de la parcelle i observées aux années $1, 2, \dots T$. La colonne t représente les LU enquêtées dans tout le territoire à l'instant t. Cette matrice est un échantillon de la série temporelle des variables aléatoires $X_1(\omega), X_2(\omega), \dots X_T(\omega)$.

Afin de pouvoir utiliser les méthodes de traitement du signal, nous représentons ces données comme un champ de tenseurs de $\mathcal{R}^K$. La série temporelle de LU sur une parcelle ω sera représentée par une séquence de T vecteurs appartenant à $\mathcal{R}^K$. Le vecteur x ayant toutes ses composantes à 0 exceptée la i^e égale à 1

$$x = \begin{bmatrix} 0 \\ \dots \\ 1 \\ \dots \\ 0 \end{bmatrix} \leftarrow i$$

représentera la catégorie e_i. La composante i du vecteur x_t est la fonction Dirac qui représente l'observation de e_i comme fonction de t. Le champ de tenseurs remplace la série temporelle d'items par une série de fonctions de Dirac multidimensionnelles permettant ainsi l'utilisation des techniques associées aux signaux aléatoires multidimensionnels.

2.1 Moment d'ordre deux croisé

Nous définissons le second moment croisé par l'espérance :

$$\begin{aligned} C_X(\tau) &= E\left[X_t(\omega)X^*_{t+\tau}(\omega)\right] \\ &= \frac{1}{T}\sum_t x_t x^*_{t+\tau} Prob(x_t, x_{t+\tau}) \end{aligned} \quad (1)$$

dans lequel x^* représente la transposé du vecteur x.

Chaque terme $x_t x^*_{t+\tau}$ dans la somme de l'équation 1 est une matrice $K \times K$. Quand $(X_t, X_{t+\tau})$ prend les valeurs (e_i, e_j) aux instants $(t, t+\tau)$ le produit $x_t x^*_{t+\tau}$ est une matrice nulle avec un 1 à l'indice (i, j).

$$x_t x^*_{t+\tau} = \begin{bmatrix} 0 & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & \cdots & 1 & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix} \begin{matrix} \\ \\ \leftarrow i \\ \\ \\ \end{matrix}$$
$$\begin{matrix} & & \uparrow & & \\ & & j & & \end{matrix}$$

Le terme général (i, j) de $C_X(\tau)$ est :

$$C_X(\tau)(i, j) = \frac{1}{T}\sum_t Prob(x^i_t, x^j_{t+\tau}) \quad (2)$$

x^i_t est la composante i du vecteur x_t et signifie qu'à l'instant t l'item e_i a été observé. La probabilité $Prob(x^i_t, x^j_{t+\tau})$ peut être estimée à l'aide du nombre d'occurrences dans la matrice M du couple (e_i, e_j) dans les colonnes t et $t+\tau$ respectivement.

2.2 La fonction d'auto covariance

La fonction d'auto covariance est définie à partir de l'équation 1 mais avec des variables centrées.

$$\begin{aligned} R_{XX}(\tau) &= E\left[(X_t(\omega) - E\left[X_t(\omega)\right])(X^*_{t+\tau}(\omega) - E\left[X^*_{t+\tau}(\omega)\right])\right] \\ &= E\left[X_t(\omega)X^*_{t+\tau}(\omega)\right] - E\left[X_t(\omega)\right] E\left[X^*_{t+\tau}(\omega)\right] \end{aligned} \quad (3)$$

Dans un processus stationnaire, l'équation 3 devient

$$R_{XX}(\tau) = E\left[X_t(\omega)X^*_{t+\tau}(\omega)\right] - E^2\left[X(\omega)\right] \quad (4)$$

Le terme général (i, j) est égal à :

$$R_{XX}(\tau)(i, j) = \frac{1}{T}\sum_t Prob(x^i_t, x^j_{t+\tau}) - E_i E_j \quad (5)$$

E_i est la composante i du vecteur $E\left[X(\omega)\right]$.

3 Expérimentation sur des données artificielles

3.1 Génération de données périodiques artificielles

Afin de démontrer l'intérêt de $R_{XX}(\tau)$ pour révéler des items périodiques, nous avons construit des séquences présentant des comportements périodiques à l'aide d'un modèle de Markvov caché (HMM) (*cf.* figure 1).

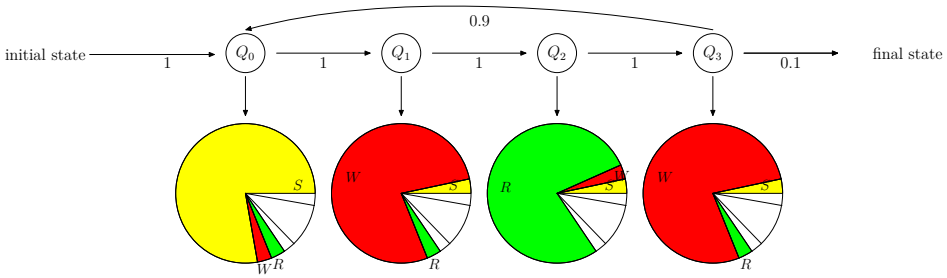


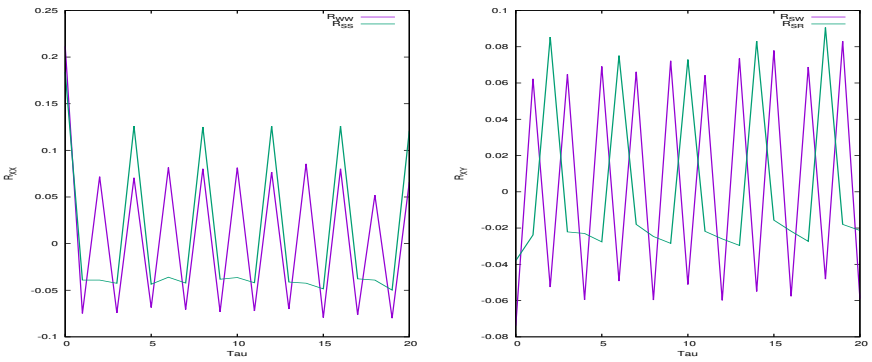
FIG. 1 – *HMM pour simuler des répétitions du patron [S-W-R-W]. Les Q_i sont les états du HMM. Les camemberts de secteurs colorés représentent les probabilités utilisées pour générer les différents items à chaque état. Tous les items ne sont pas représentés*

Nous avons défini tout d'abord un ensemble $\mathcal{E} = \{S, W, R, \ldots\}$ de 11 labels représentant les 11 LU rencontrés dans les parcelles : Tournesol (*Sunflower*), Blé (*Wheat*), Colza (*Rapeseed*),... Nous avons ensuite construit 8 répétitions du patron [S−W−R−W] pour obtenir une séquence de 32 symboles représentant la suite des LU sur une parcelle pendant 32 années. Afin de bruiter cette séquence, des substitutions de symboles ont été effectuées. Ce processus a été répété pour construire 50 séquences de 32 symboles chacune. Le but de cette expérience est de retrouver une période de 4 pour les symboles R et S ainsi qu'une période de 2 pour le symbole W.

Pour simuler ces séquences, le HMM se comporte comme un automate dans lequel un jeton se déplace aléatoirement depuis l'état initial jusqu'à l'état final en fonction des transitions possibles entre états. A chaque état, un label issu de $\mathcal{E}$ est aléatoirement produit à l'aide de la densité de probabilité (pdf) associée à cet état. Le processus est répété jusqu'à produire une séquence de la longueur désirée.

Différentes pdf de différentes entropies sont utilisées pour simuler différents niveaux de bruit dans les séquences. L'entropie est une mesure de l'imprédictibilité d'une pdf. On la mesure en nombre de bits nécessaires pour numéroter et distinguer les évènements. Elle est maximale pour la loi uniforme – tous les évènements doivent être numérotés – et nulle pour une loi certaine : il n'y a rien à numéroter. Pour une pdf possédant n symboles de probabilité $p_i, i = 1, n$, elle est définie par :

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i \tag{6}$$



(a) Périodes de S (=4) et W (=2) (b) Décalages (lags) entre S-W (=1) et S-R (=2)

FIG. 2 – *Signal d'auto corrélation calculé sur des séquences générées par le* HMM *décrit Fig. 1 construit avec des pdf d'entropie 2,65 bits*

3.2 Analyse spectrale des fonctions d'auto corrélation

En traitement du signal, la fonction réelle d'auto corrélation $R_{XX}(\tau)(i,i)$, $\tau = 0, T-1$ est utilisée pour révéler les comportements périodiques à l'aide d'une analyse spectrale donnée par une transformée de Fourier discrète rapide (FFT). Dans le cas particulier de signaux artificiels, l'observation de la courbe $R_{XX}(\tau)(i,i)$ peut suffire pour révéler des comportements périodiques. La période est le petit décalage – ou retard – en temps (*lag* en anglais) où la fonction d'auto corrélation atteint un maximum. La figure 2(a) montre que le symbole "W" (courbe mauve) a une période de 2 – ou un temps de retour de 2 dans le langage des agronomes – , alors que le symbole "S" (courbe verte) a une période de 4. Nous développons aussi une analyse spectrale qui pourra servir à détecter ces mêmes motifs sur des signaux réels bruités. Dans ce sens, nous cherchons une représentation fréquentielle par une transformée de Fourier rapide.

$R_{XX}(\tau)(i,i)$ est tout d'abord fenêtré l'aide d'une fenêtre de Hamming de longueur 32 afin d'atténuer les problèmes liés au caractère limité de la séquence.

Lorsque nous notons $f_i(\tau) = R_{XX}(\tau)(i,i)$ une fonction de τ paramétrée par i, la FFT discrète $\mathcal{F}_i(k)$ sur les $N = 32$ points est définie par :

$$\mathcal{F}_i(k) = \sum_{\tau=0}^{N-1} f_i(\tau) \exp -j\frac{2\pi k\tau}{N}, \quad k = 0, \cdots N-1 \tag{7}$$

Les valeurs $\frac{k}{N}$ sont appelées les fréquences alors que $\frac{N}{k}$ représentent les périodes. La représentation graphique du module $\parallel \mathcal{F}_i(k) \parallel$ vu comme une fonction de k/N est appelée un spectrogramme et vu comme une fonction de N/k un périodogramme. Une valeur de $N = 32$ permet une résolution fréquentielle raisonnable pour le genre de signaux simulés. Leur pendants réels doivent être des résultats d'enquêtes annuelles de terrain. Il est irréaliste d'envisager des séquences de longueur supérieure à 32. La figure 3 montre le spectrogramme $\parallel \mathcal{F}_i(k) \parallel$ pour les items "S" et "R".

Afin de détecter les pics dans un périodogramme, nous avons suivi la stratégie développée par (Li et al., 2012; Vlachos et al., 2005). Chaque série artificielle est réarrangée aléatoirement

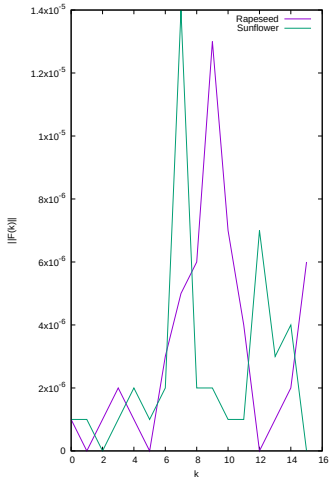


FIG. 3 – *Spectrogramme des items "R" et "S". Les ordonnées représentent* $\| \mathcal{F}_i(k) \|$ *pour les items* $i = Rapeseed$ *et* $i = Tournesol$*. L'axe des abscisses représente le* k *de la fréquence* $\frac{k}{N}$*. Dans une fenêtre de longueur (*$N = 32$*) chaque item montre une fréquence de* $\frac{8}{32}$ *soit une période de* $\frac{32}{8} = 4$*. Les pics de part et d'autre de* $k = 8$ *correspondent à la fréquence 1/4. L'imprécision du résultat provient du faible nombre de valeurs* $N = 32$ *de la série*

pour faire disparaître tout comportement périodique. Après le calcul de la fonction d'auto corrélation, son fenêtrage et sa FFT, le maximum du périodogramme est stocké. Cent réarrangements sont effectués pour calculer moyenne et écart type des différents maximums. Le seuil qui détermine si un pic est significatif est placé à la moyenne plus deux fois l'écart type ce qui correspond à une confidence de 95%.

$R_{XX}(\tau)(i,j), i \neq j$ donne aussi une indication sur la co-occurrence des symboles e_i et e_j. Fig. 2(b) montre que la corrélation entre les symboles "S" et "W" au décalage (*lag*) de 1 est plus important qu'à 2 ce qui signifie que le couple "S-W" est plus fréquent que le patron "(S- ?-W)" (le caractère joker " ?" représente n'importe quel autre symbole appartenant à $\mathcal{E}$).

Une analyse similaire peut être faite sur le signal $R_{XX}(\tau)(i,j)$ lorsque (i,j) représente les symboles "S" et "R" et montre un maximum au décalage 2 signifiant que "S- ?-R" doit être suspecté. Donc, $R_{XX}(\tau)(i,j)$ peut être utilisé comme un trait pour hypothétiser des successions de symboles dans des séries temporelles bruitées.

4 Conclusion

Cet article décrit une méthode hybride symbolique et numérique pour analyser un ensemble de séries temporelles d'items afin d'en extraire des comportements périodiques. Le passage d'une représentation sous forme d'une série temporelle d'items à une représentation sous forme d'une série temporelle de tenseurs multidimensionnels nous permet d'utiliser les techniques de traitement de signaux multidimensionnels.

Utilisation de réseau de neurones siamois en clustering : application aux événements du réseau électrique français

Laure Crochepierre*,**, Antoine Marot**, Vincent Barbesant**,
Benjamin Donnot**, Lydia Boudjeloud-Assala*

* Université de Lorraine, CNRS, LORIA, F-57000 Metz
prénom.nom@univ-lorraine.fr
Rte R&D, Paris, France
**prénom.nom@rte-france.com

Résumé. Cet article propose d'étudier l'utilisation d'un réseau neuronal siamois dans le cadre de la labellisation des événements du réseau électrique français. Après une première étape de labellisation partielle à partir de règles expertes, nous entraînons un réseau de neurones siamois pour définir une mesure de similarité spécifique aux données. Là où d'autres mesures telles que le Dynamic Time Warping ne fournissaient pas de résultats pertinents, l'application de cette approche aux données de la région de Lyon sur l'année 2017 permet de mettre en évidence l'utilité des réseaux siamois pour l'exploration de données et l'identification de nouvelles classes et sous-classes.

1 Introduction

Rte ("Réseau de Transport d'Electricité") dispose de 10 ans d'historique sur les actions de ses composants électriques télécommandables, à l'origine mesurées pour l'exploitation en temps réel du réseau électrique. Bien que ces données soient de bonne qualité, les causes des actions prises sur ces composants n'ont pas été sauvegardées au moment de leur acquisition. Un enjeu important pour Rte aujourd'hui est donc de pouvoir labelliser a posteriori ces données afin de les exploiter ensuite pour de l'apprentissage supervisé. Du fait du coût élevé et de l'expertise nécessaire pour labelliser manuellement, l'entreprise souhaiterait réaliser cette étape par des méthodes automatiques ou très faiblement supervisées. Dans un premier temps, nous restreignons notre labellisation à trois classes d'événements récurrents de la vie du réseau électrique (notées par la suite A, B et C) :

- la classe A : les consignations (des événements de maintenance d'ouvrages du réseau électrique)
- la classe B : les événements de fiabilisation du matériel en exploitation
- la classe C : les événements lors desquels une opération de fiabilisation est mise en oeuvre simultanément avec une consignation.

Bien qu'il existe de nombreuses autres catégories (et sous-catégories) d'événements sur le réseau, nous nous intéressons à ces trois catégories afin de construire notre méthode et de la valider sur des données pour lesquelles nous disposons de labels.

Dans ce papier, nous présentons l'utilisation d'un réseau de neurones siamois comme une alternative à l'utilisation du *Dynamic Time Warping* (DTW) (Sakoe et Chiba, 1978) en tant que mesure de similarité entre les séquences d'actions. Nous comparerons les résultats de clustering obtenus et présentons les possibilités d'exploration de l'espace de projection créé par le réseau siamois.

2 Description des données

Pour reconstituer les événements survenus sur le réseau électrique, nous étudions ici l'historique des actions effectuées sur les composants commandables du réseau. Mesurée sous forme de signal binaire discret, une action a_i peut être définie par un quadruplet tel que $a_i = (t_i, \delta p_i, \delta t_i, c_i)$ avec t_i l'horodate de sa mesure, δp_i la valeur du changement de position binaire ainsi que δt_i sa durée de persistance et c_i le type de composant manoeuvré. Afin de capturer les dépendances spatio-temporelles entre nos données, nous associons les actions mesurées sous forme de séquences temporelles $S = (a_i)_{i \in N}$ de longueur variable telle qu'une séquence corresponde à l'ensemble des actions sur une zone spatiale cohérente pendant une même journée. Selon ce découpage nous obtenons 40485 séquences spatio-temporelles de longueur variable et c'est ensuite au sein de ces séquences que nous cherchons les classes A,B et C présentées précédemment.

Afin d'obtenir un premier ensemble de données labellisées sur lequel travailler, nous avons construit, de manière itérative en coopération avec des experts, un ensemble de règles permettant de discriminer des exemples au sein des trois classes. L'application de ces règles a pu ensuite être validée partiellement grâce à des fichiers fournis par les opérateurs. Nous avons ainsi pu découvrir 6907 séquences dont 3044 de la classe A, 3389 de la classe B et 474 de la classe C. Néanmoins, 80% de l'ensemble des séquences utilisables restent non-labellisées et l'exploration de nouvelles méthodes est nécessaire pour trouver plus de labels ainsi que de nouvelles classes.

3 Méthode proposée

Du fait de l'hétérogénéité des variables et de l'importance différence de longueur entre les séquences, la définition explicite d'une mesure de similarité adaptée et calculable avec un budget computationnel limité s'avère être une tâche complexe. Ainsi, bien que des mesures classiques telles que DTW soient utilisables sur des séquences temporelles multivariées de longueur variable, le temps de calcul nécessaire pour traiter l'intégralité des séquences rend difficile son utilisation dans notre contexte. L'approche que nous allons maintenant présenter s'appuie sur un processus d'apprentissage : plutôt que de définir une mesure, il est possible de l'apprendre spécifiquement pour un problème donné à partir d'exemples sélectionnés. Cet apprentissage permet en outre d'intégrer une expertise opérationnelle sur des labels existants et d'exploiter les similarités des séquences. Inspirés par les récentes performances atteintes par les réseaux de neurones siamois (Bromley et al., 1994) dans les tâches de clustering, notamment pour l'exploration et la découverte de classes inconnues (Bahaadini et al., 2018), nous avons choisi de l'explorer sur nos données.

Un réseau de neurones siamois est composé de deux sous-réseaux identiques partageant les mêmes poids et travaillant en parallèle sur des entrées distinctes. Chaque entrée est traitée par l'un des deux sous-réseaux afin d'obtenir un vecteur de projection de même taille. Les deux vecteurs obtenus par les deux sorties des sous-réseaux sont ensuite comparés, au moyen d'une norme prédéfinie avant l'entraînement, afin de d'obtenir en sortie du réseau complet une mesure de similarité entre les deux entrées. L'objectif que l'on cherche à atteindre est de minimiser (resp. maximiser) cette mesure pour deux objets que l'on sait être semblables (resp.dissimilaires). Ceci est mis en oeuvre au moyen d'une perte particulière appelée contrastive loss (Hadsell et al., 2006).

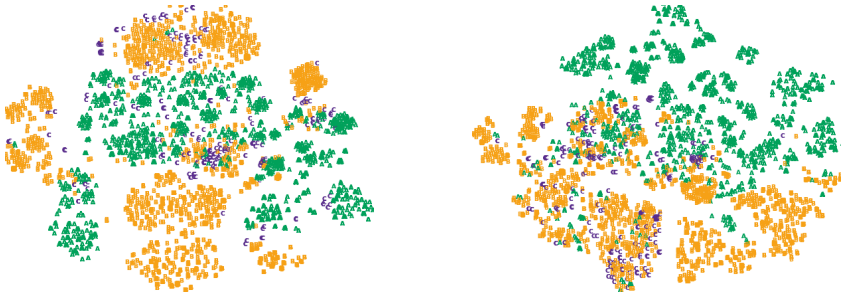
Ce type de réseau est entraîné à partir de paires de séquences S_1 et S_2 labellisées, chaque séquence pouvant appartenir à la classe A, B ou C : le label de la paire (noté y), utilisé en sortie du réseau, valant 0 pour deux séquences de même classe et 1 sinon. Le réseau projette chaque entrée sous forme de vecteur dont la taille a été préalablement choisie. Nous avons choisi ici la taille la plus petite permettant la convergence de l'entraînement du réseau. La fonction non linéaire de projection créée par l'entraînement permet de comparer les deux séquences projetées $P(S_1)$ et $P(S_2)$ directement via une norme choisie avant à l'entraînement (L_2 dans notre cas). La mesure en sortie est alors minimale pour deux séquences de même classe et maximale sinon. Le réseau de neurones siamois permet donc d'obtenir à la fois une mesure de similarité entre les exemples mais également un projection des données dans un nouvel espace ayant inclus la connaissance des labels. Une fois la projection obtenue, nous pouvons désormais travailler dans ce nouvel espace pour réaliser un clustering ou de l'exploration de données.

4 Expérimentations

Exploration de la projection Le réseau permet d'obtenir une projection des données dans un espace facilement explorable après réduction de dimension par des algorithmes tels que la t-SNE (Maaten et Hinton, 2008). Une exploration de cette projection permet de déduire des regroupements sur les données (de manière interactive ou algorithmique) qui n'auraient pas été envisagés dans l'étude des données sous un format de séquences. Un exemple de projection obtenu par t-SNE avec une perplexité de 30 et un learning rate de 200 est présenté dans la Figure 1, illustrant une meilleure séparabilité apparente des données dans l'espace de projection.

Comparaison avec la mesure de DTW sur une tâche de clustering Dans une optique de minimisation de l'effort de labellisation, nous souhaiterions nous appuyer sur la structure des données pour les regrouper en clusters : le label attribué au cluster serait alors le label majoritaire parmi ceux connus dans le cluster. Avant d'appliquer un clustering à l'ensemble des 40485 séquences majoritairement non-labellisées, nous allons éprouver cette approche sur les séquences dont les labels ont été validés, afin de tenter de retrouver les 3 classes connues.

En s'inspirant du protocole décrit dans Sardá-Espinosa (2017), nous avons cherché dans un premier temps à comparer les performances des algorithmes de clustering sur les 6907 séquences multivariées préalablement labellisées. A ces fins, nous avons utilisé en parallèle la mesure de DTW et la sortie du réseau de neurones siamois comme mesures de similarité entre séquences. Les calculs ont été menés sur différents algorithmes de clustering tel que la



(a) similarités avec DTW sur les données brutes (b) similarités apprises sur les données projetées

FIG. 1: Représentation des données réduite par t-SNE. Les couleurs vert, orange et violet représentent respectivement des séquences de type A, B et C.

Classification Ascendante Hiérarchique (CAH) (S. Michalski et E. Stepp, 1983) avec différents critères d'aggrégation (lien simple, complet, moyen ou critère de Ward) et les K-Medoids (MacQueen et al., 1967) en retenant deux critères pour la validation des clusters obtenus : l'indice de Calinski Harabaz (Caliński et Harabasz, 1974) et la silhouette (Rousseeuw, 1987). Une analyse croisée des scores des deux critères de validation à l'optimum, calculés pour différents algorithmes, nous a permis d'identifier 3 clusters avec les deux formats de données.

Pour approfondir plus en détail la compréhension des clusters obtenus, nous avons procédé à une analyse de la répartition des classes A, B et C dans les différents clusters trouvés par les deux algorithmes, en leur attribuant le label majoritaire. Sur la classification correspondante, nous obtenons les scores les plus élevés pour l'algorithme de CAH avec le critère de Ward. Les scores utilisés sont le F1-score pondéré par le nombre d'instances de chaque classes, ainsi que la proportion de séquences correctement labellisées parmi l'ensemble des 6907 séquences utilisées.

		validation métier					
		séquences			projections		
		A	B	C	A	B	C
prédiction	A	39,8	35,8	3,9	42	33	4
	B	4,2	13,4	2,9	2	16	3
	C	0	0	0	0	0	0
F1-score pondéré		0,57			**0,64**		
% labels correct		53			**66**		

TAB. 1: Matrices de confusion de l'approche non supervisée par méthode de Ward. Les résultats ont été calculés en pourcentages sur 6907 séquences sur les séquences et leur projection.

En comparant, dans la Table 1, les résultats de clustering obtenus par la distance DTW avec obtenus sur la projection, nous remarquons des scores plus élevés avec la mesure du réseau siamois où le F1-score pondéré est de $0,64$ contre $0,57$ avec la DTW. Ces résultats indiquent que le réseau a effectivement permis d'intégrer la connaissance des labels pour créer une représentation des données, où les données d'une même classe sont représentées de manière plus

uniforme. Ils suggèrent par ailleurs la possibilité d'utiliser la projection pour les données non-labellisées où la représentation des données serait plus similaire à celles des données de même classe. Enfin en intégrant ces données à des outils d'exploration tels que Tensoboard Projector (Smilkov et al., 2016), il sera possible de labelliser de nouvelles instances, proches dans l'espace d'instances déjà labellisées.

5 Conclusions et perspectives

Dans ce papier nous proposons l'utilisation de la sortie d'un réseau siamois entraîné comme mesure de similarité utilisable à des fins de labellisation d'événements du réseau électrique français. Nous comparons l'utilisation d'une métrique créée à partir des données par le réseau, à une mesure de DTW sur une tâche de clustering. Les résultats auxquels nous avons abouti montrent une représentation plus uniforme des données au regard des trois classes connues, tout en suggérant de nombreuses améliorations possibles au modèle depuis le choix de la structure initiale des données jusqu'au choix du modèle de réseau de neurones. Les projections construites faciliteront à l'avenir l'exploration des données sur des séquences non labellisées. Nous pourrons ainsi examiner les sous-groupes et nouveaux groupements identifiés par le clustering de la projection pour y découvrir de nouveaux événements. A terme, l'objectif sera d'intégrer des experts du métier dans un processus de labellisation itératif au cours duquel ils pourront proposer manuellement des labels, identifier de nouvelles classes, puis ré-entrainer le réseau afin d'obtenir une projection de plus en plus pertinente.

Références

Bahaadini, S., N. Rohani, A. K. Katsaggelos, V. Noroozi, S. Coughlin, et M. Zevin (2018). Direct : Deep discriminative embedding for clustering of ligo data. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pp. 748–752.

Bromley, J., I. Guyon, Y. LeCun, E. Säckinger, et R. Shah (1994). Signature verification using a" siamese" time delay neural network. In *Advances in neural information processing systems*, pp. 737–744.

Caliński, T. et J. Harabasz (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods 3*(1), 1–27.

Hadsell, R., S. Chopra, et Y. LeCun (2006). Dimensionality reduction by learning an invariant mapping. In *Computer Vision and Pattern Recognition*, pp. 1735–1742. IEEE.

Maaten, L. v. d. et G. Hinton (2008). Visualizing data using t-sne. *Journal of machine learning research 9*(Nov), 2579–2605.

MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proc. Fifth Berkeley Symp. on Math. Statist. and Prob.*, Volume 1.

Rousseeuw, P. J. (1987). Silhouettes : a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics 20*, 53–65.

S. Michalski, R. et R. E. Stepp (1983). Automated construction of classifications conceptual clustering versus numerical taxonomy. *Pattern Analysis and Machine Intelligence, IEEE Transactions on PAMI-5*, 396 – 410.

Sakoe, H. et S. Chiba (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing 26*(1), 43–49.

Sardá-Espinosa, A. (2017). Comparing time-series clustering algorithms in r using the dtwclust package. *R package vignette 2*.

Smilkov, D., N. Thorat, C. Nicholson, E. Reif, F. B. Viégas, et M. Wattenberg (2016). Embedding projector : Interactive visualization and interpretation of embeddings. *stat 1050*, 16.

Summary

This article uses the forward-pass of a siamese neural network as a similarity measure for labelling events on the French power system. After a first labelling step using logical rules, we train a siamese neural network to define a similarity measure from data. While Dynamic Time Warping couldn't show satisfactory results, the siamese neural network provide a new tool for data exploration and new class identification.

Détection en ligne de multiples changements dans un panel de données catégorielles

Milad Leyli-abadi*, Allou Samé*, Latifa Oukhellou*

*Université Paris-Est, IFSTTAR, COSYS, GRETTIA, F-77447 Marne-la-Vallée, France
{milad.leyli-abadi, allou.same, latifa.oukhellou}@ifsttar.fr

Résumé. Cet article présente une méthode de détection de changements communs à un ensemble de séquences catégorielles. La méthode proposée est basée sur un test séquentiel de rapport de vraisemblance généralisé fondé lui-même sur des chaînes de Markov non homogènes modélisant les données avant et après les changements.

1 Introduction

De nos jours, l'usage des données longitudinales devient de plus en plus répandu dans de nombreux domaines. Par exemple, dans les réseaux urbains (électricité ou eau), les compteurs intelligents permettent la collecte de telles données sur la consommation de multiples usagers. Dans ce travail, nous nous intéressons plus spécifiquement à l'analyse conjointe de multiples séquences catégorielles où chaque catégorie correspond à un mode d'usage dans le réseau.

Cet article propose une méthode en ligne basée sur le test séquentiel du rapport de vraisemblance généralisé (Basseville et al., 1993) pour détecter des changements dans de multiples séquences catégorielles. Ces changements pourront être interprétés comme des modifications du comportement des usagers du réseau. Nous proposons un seuil adaptatif permettant de décider d'éventuels changements de comportement. Dans le domaine des réseaux urbains notamment, la détection de changement permettra aux gestionnaires de réseau, de mieux répondre aux besoins évolutifs des usagers.

D'autre part, le comportement des usagers étant très souvent lié à des facteurs exogènes (température, précipitations, etc.) (House-Peters et al., 2010), nous modélisons les séquences catégorielles avant et après changement par une chaîne de Markov non homogène.

L'article est organisé de la manière suivante : les sections 2 et 3 décrivent respectivement les données et la méthodologie adoptée. Les résultats expérimentaux et l'évaluation de la méthode proposée sont détaillés dans la section 4.

2 Données

Le panel de données catégorielles analysé dans cet article, noté $(z_{it})_{1\leq i\leq n, 1\leq t\leq T}$, est relatif à n entités (ex. compteurs communicants) observées durant T instants (jours ou semaines). La figure 1 illustre ce type de données, chaque couleur faisant référence à une catégorie. En pratique, les catégories sont obtenues à partir d'une étape de discrétisation des profils journaliers

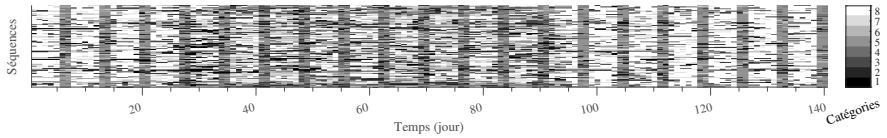


FIG. 1: Séquences catégorielles correspondant au comportement de 100 usagers d'un réseau durant 140 jours. Chaque ligne correspond à l'évolution du comportement d'un usager et chaque catégorie correspond à un profil d'usage journalier

de consommation s'appuyant sur la classification automatique de courbes (Samé et al., 2016). Dans notre cas, les données exogènes associées à ces séquences catégorielles, qui seront notée $\mathbf{u} = (u_{it})$, comprennent généralement la température, la précipitation et les données calendaires. Dans la suite de l'article, on utilisera de manière équivalente les notations $(\mathbf{z}_1, ..., \mathbf{z}_T)$, avec $\mathbf{z}_t = (z_{it})_{i=1,...,n}$.

3 Méthode proposée

L'approche proposée pour la détection de changement dans un ensemble de séquences catégorielles est basée sur le test séquentiel du rapport de vraisemblance généralisé. Les hypothèses du test sont définies comme suit :

$$\begin{cases} H_0 : & (\mathbf{z}_1, ..., \mathbf{z}_T) & \sim & P_{\tilde{\theta}_0} \\ H_A : & (\mathbf{z}_1, ..., \mathbf{z}_{\tau-1}) & \sim & P_{\theta_0} \\ & (\mathbf{z}_\tau, ..., \mathbf{z}_T) & \sim & P_{\theta_1} \end{cases} \tag{1}$$

où H_0 est l'hypothèse sous laquelle la séquence entière de données est distribuée suivant la même loi $P_{\tilde{\theta}_0}$ et l'hypothèse H_1 considère qu'à partir d'un instant τ, les données ne suivent plus le même modèle qu'avant cet instant. La distribution P_θ d'une séquence $(\mathbf{z}_a, \ldots, \mathbf{z}_b)$ donnée, avec $a < b$, est supposée être celle d'une chaîne de Markov non homogène dont les probabilités initiales et de transition sont définies comme suit :

$$P_\theta(z_{i,1} = k|\mathbf{u}_1) = \frac{e^{\boldsymbol{\alpha}_k^\top \mathbf{u}_{i,1}}}{\sum_{\ell=1}^K e^{\boldsymbol{\alpha}_\ell^\top \mathbf{u}_{i,1}}}, \tag{2}$$

$$P_\theta(z_{i,t} = k|z_{i,t-1} = \ell, \mathbf{u}_t) = \frac{e^{\boldsymbol{\beta}_{k,\ell}^\top \mathbf{u}_{i,t}}}{\sum_{h=1}^K e^{\boldsymbol{\beta}_{h,\ell}^\top \mathbf{u}_{i,t}}}, \tag{3}$$

où $\boldsymbol{\alpha}$ et $\boldsymbol{\beta}$ désignent respectivement les paramètres associés aux probabilités initiales et de transition. Afin de décider entre les deux hypothèses, nous nous appuyons sur le logarithme du rapport de vraisemblance, défini par :

$$\Lambda_1^T(\tau) = \log \left(\frac{\left(\prod_{i=1}^n P_\theta(z_{i1}|\mathbf{u}_{i1}) \prod_{t=2}^{\tau-1} P_\theta(z_{it}|z_{it-1}, \mathbf{u}_{it}) \times \right.}{\left. \prod_{i=1}^n P_\theta(z_{i\tau}|\mathbf{u}_{i\tau}) \prod_{t=\tau+1}^{T} P_\theta(z_{it}|z_{it-1}, \mathbf{u}_{it}) \right)} \right) \tag{4}$$

En développant cette équation, on obtient :

$$\begin{aligned}\Lambda_1^T(\tau) &= \sum_{i=1}^{n}\log P_{\theta_0}(z_{i1}|\mathbf{u}_{i1}) + \sum_{i=1}^{n}\sum_{t=2}^{\tau-1}\log P_{\theta_0}(z_{it}|z_{it-1},\mathbf{u}_{it}) \\ &+ \sum_{i=1}^{n}\log P_{\theta_1}(z_{i\tau}|\mathbf{u}_{i\tau}) + \sum_{i=1}^{n}\sum_{t=\tau+1}^{T}\log P_{\theta_1}(z_{it}|z_{it-1},\mathbf{u}_{it}) \\ &- \sum_{i=1}^{n}\log P_{\tilde{\theta}_0}(z_{i1}|\mathbf{u}_{i1}) - \sum_{i=1}^{n}\sum_{t=2}^{T}\log P_{\tilde{\theta}_0}(z_{it}|z_{it-1},\mathbf{u}_{it}).\end{aligned} \tag{5}$$

Les paramètres $(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_{0\ell}, \boldsymbol{\alpha}_1, \boldsymbol{\beta}_{1\ell}, \tilde{\boldsymbol{\alpha}}_0, \tilde{\boldsymbol{\beta}}_{0\ell}, \tau)$ sont estimés en utilisant la méthode de maximum de vraisemblance :

$$\Lambda_T = \max_{\tau,(\boldsymbol{\theta}_0,\boldsymbol{\theta}_1),\tilde{\boldsymbol{\theta}}_0} \Lambda_1^T(\tau). \tag{6}$$

La règle de décision suivante permet finalement de décider entre les deux hypothèses :

$$d = \{0 \text{ si } \Lambda_T < h;\ 1 \text{ si } \Lambda_T \geq h\}, \tag{7}$$

où h est le seuil de décision. Si la statistique de test dépasse cette valeur, un changement est détecté. La stratégie qui vient d'être décrite permet la détection d'un unique point de changement dans une séquence. Pour adapter cette stratégie à la détection de multiples changements, nous l'appliquons de manière séquentielle sur des fenêtres de taille croissante. Tant qu'aucun point de changement n'est détecté, la méthode de détection est appliquée sur une nouvelle fenêtre de taille plus grande que la précédente. Dès qu'un point de changement est détecté, la fenêtre est réinitialisée de même que le seuil de détection.

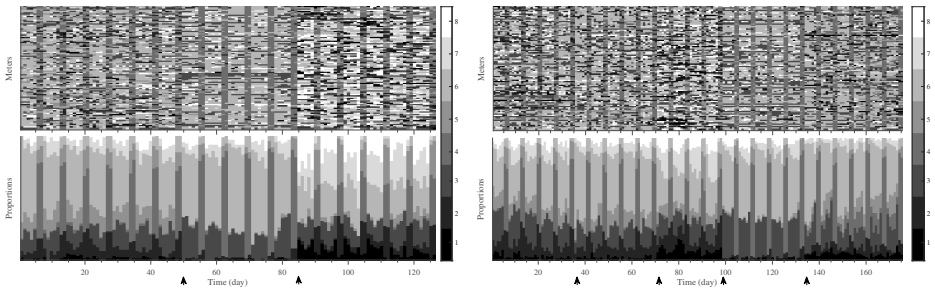
Il est évident que dans le scénario le plus défavorable où les séquences catégorielles sont de grande taille et ne comprennent aucun point de changement, la complexité algorithmique de la méthode proposée augmentera significativement compte tenu de la taille croissante de la fenêtre de détection. Pour résoudre ce problème, on peut utiliser une fenêtre dont la taille est majorée.

Estimation d'un seuil adaptatif. Pour avoir une estimation de la valeur du seuil, nous avons effectué des simulations de type Monte Carlo. À partir d'une séquence de données initiales ne présentant pas de changement (comportement nominal), un modèle de Markov non homogène est d'abord estimé par la méthode du maximum de vraisemblance. Ensuite, plusieurs séquences sont générées à partir de ce modèle et la statistique de test est évaluée pour chacune de ces séquences. Le fractile $(\mathcal{Q}_{1-p})$ de la distribution des statistiques de test est considéré comme la valeur du seuil. Le seuil est estimé de nouveau après chaque détection.

4 Expérimentation

Afin d'évaluer la performance de la méthode proposée, nous avons conçu deux bases de données en considérant un nombre de changements différent (voir Figure 2). Ces bases sont

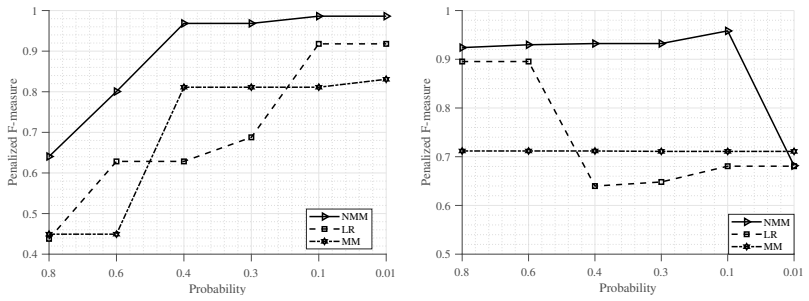
obtenues en générant aléatoirement des séquences à partir de paramètres réalistes. La méthode proposée est comparée à deux approches, l'une basée sur un modèle de Markov homogène (MM) et l'autre ne faisant pas l'hypothèse markovienne (équivalent à un modèle de régression logistique). Cette dernière méthode sera donc notée LR.



(a) Base 1 (2 changements) (b) Base 2 (4 changements)

FIG. 2: Représentation graphique des jeux données. Les figures du dessus représentent les bases de données générées, et les figures du dessous représentent la proportion des catégories au fils du temps. Les changements sont indiqués par des flèches sur l'axe des abscisses.

La comparaison est effectuée en termes de la F-mesure (voir la Figure 3) et trois autres critères (voir le tableau 1) qui sont l'aire sous la courbe ROC (AUC), le taux de vrais positifs (TPR) et le délai de détection (DD). La figure 3 montre pour chacune des bases, le calcul de la F-mesure en fonction de différentes valeurs de probabilité (p) associées au fractile (différentes valeurs de seuil). En observant ces deux graphiques et le tableau 1, on remarque que les meilleures performances sont obtenues pour la méthode proposée.



(a) Base 1 (b) Base 2

FIG. 3: Calcul de la F-mesure pour chacune des bases et en utilisant les 3 méthodes

TAB. 1: Tableau de comparaison des méthodes évaluées : MM : modèle de Markov homogène ; LR : régression logistique ; NMM : modèle de Markov non-homogène (modèle proposé). Les différents critères sont : AUC : aire sous la courbe ROC ; TPR : taux de vrais positifs ; DD : Délai de détection.

Modèle	**MM**			**LR**			**NMM**		
Critère	**AUC**	**TPR**	**DD**	**AUC**	**TPR**	**DD**	**AUC**	**TPR**	**DD**
Base 1	0.77	0.78	7.5	0.82	0.73	**5**	**0.92**	**0.96**	7.5
Base 2	0.70	0.65	**2.6**	0.83	0.76	6.3	**0.91**	**0.96**	8.7

5 Conclusion

Dans cet article, une méthode de détection de changement basée sur la rapport de vraisemblance généralisé est proposée. Un seuil adaptatif est calculé permettant d'ajuster ce dernier aux différents types de changements et de réduire le nombre de fausses alarmes. Les expérimentations réalisées sur deux bases de données simulées montrent de bonnes performances de la méthode proposée, qui recherche des points de changement communs à un ensemble de séquences catégorielles. Toutefois, si les changements varient légèrement selon les séquences, les détections obtenues par la méthode proposée pourraient refléter la moyenne de ces points de changement. L'étude de cette situation particulière constitue l'une des perspectives de ce travail.

Références

Basseville, M., I. V. Nikiforov, et al. (1993). *Detection of abrupt changes : theory and application*, Volume 104. Prentice Hall Englewood Cliffs.

House-Peters, L., B. Pratt, et H. Chang (2010). Effects of urban spatial structure, sociodemographics, and climate on residential water consumption in hillsboro, oregon 1. *JAWRA Journal of the American Water Resources Association 46*(3), 461–472.

Samé, A., Z. Noumir, N. Cheifetz, A.-C. Sandraz, et C. Féliers (2016). Décomposition et classification de données fonctionnelles pour l'analyse de la consommation d'eau. In *Conférence Internationale Francophone sur l'Extraction et la Gestion des Connaissances (EGC 2016), Atelier Clustering et Co-clustering (CluCo 2016)*, pp. 11p.

Summary

This article presents a change detection method performing on a set of categorical sequences. The proposed method is based on a generalized sequential likelihood ratio test, where the data are supposed to be distributed following a non homogeneous Markov model before and after potential change points.

Comparaison et Partitionnement de Séries Temporelles Basés sur la Forme des Séries

Brieuc Conan-Guez, Alain Gély, Lydia Boudjeloud-Assala, Alexandre Blansché

Université de Lorraine, CNRS, LORIA, F-57000 Metz, France
{brieuc.conan-guez, alain.gely, lydia.boudjeloud-assala, alexandre.blansche}
@univ-lorraine.fr

Résumé. Dans ce travail, nous nous intéressons à la classification non supervisée de séries temporelles. La méthode de partitionnement utilisée, dérivée des centres mobiles, s'appuie sur la forme des séries pour évaluer leurs ressemblances. Afin de comparer ces formes, nous proposons deux mesures de ressemblance entre séries temporelles invariantes par translation et par changement d'échelle. La première mesure est une adaptation de la mesure cosine pour laquelle l'alignement temporel optimal entre deux séries est obtenu en testant toutes les translations d'une série par rapport à l'autre. La seconde mesure est une version *soft* de la première : le calcul du min sur les différents alignements est effectué grâce à la fonction softmin.

1 Introduction

La classification de séries temporelles est une problématique majeure depuis de très nombreuses années. Dans ce travail, nous nous intéressons spécifiquement à des méthodes de partitionnement qui effectuent une comparaison des séries temporelles basée sur un critère de forme. Les mesures de ressemblance proposées sont doublement invariantes : elles ne dépendent pas de l'amplitude des séries (normalisation des valeurs des séries), et sont invariantes par décalage temporel. Contrairement à la mesure de déformation temporelle dynamique (DTW), qui effectue une déformation non linéaire du temps (le meilleur alignement est obtenu par contraction ou dilatation locale de l'axe temporel), les mesures de ressemblance auquelles on s'intéresse dans ce travail effectuent une translation uniforme de l'axe temporel. Ce type de mesures s'applique naturellement dans le cas où l'origine temporelle des séries est inconnue, mal définie ou différente entre séries : signaux périodiques, signaux pour lesquels l'outil de mesure peut induire un décalage sur l'axe des x (données spectrométriques), signaux pour lesquels la captation des données débute après l'origine du phénomène. Ce dernier cas se retouve par exemple lors de l'analyse du cycle de vie de hashtags apparaissant sur des réseaux socio-numériques : la période de captation des tweets citant le hashtag débute après l'apparition du premier tweet mentionnant le hashtag.

Plusieurs méthodes de partitionnement s'appuyant sur ce type de mesures ont déjà été proposées par le passé. On peut citer K-Spectral Centroïd (Yang et Leskovec (2011)) et K-Shape (Paparrizos et Gravano (2017)). Ces méthodes ont été éprouvées sur de nombreux jeux de données (Paparrizos et Gravano (2017)) et comparées avec de nombreuses méthodes comme

K-MeansDTW (DTW Barycenter Averaging). Les méthodes K-Spectral Centroïd (KSC) et K-Shape (KS) ont un coût de calcul assez important, évoluant cubiquement avec la longueur des séries L. Certaines astuces de calcul (Conan-Guez et al. (2018)) permettent malgré tout d'observer dans la pratique un comportement quadratique.

Dans ce travail, nous proposons deux nouvelles mesures de ressemblance entre séries temporelles. La première est similaire à celles de KSC et KS, et s'appuie sur la mesure cosine qui assure l'invariance par changement d'échelle sur l'axe Y. L'invariance par décalage temporel est obtenue en testant tous les décalages de la seconde série par rapport à la première afin d'identifier l'alignement temporel optimal. La méthode de partitionnement basée sur cette première mesure se révèle très rapide avec une complexité sous-quadratique $O(L\log(L))$ en la longueur des séries L. La seconde mesure est la version régularisée de la première : le calcul du minimum apparaissant dans la recherche de l'alignement temporel optimal est effectué cette fois grâce à la fonction softMin. Bien que cette seconde mesure soit plus coûteuse à évaluer, elle présente l'avantage d'être régulière, et par la même peut être adaptée à des méthodes de fouille de données nécessitant le calcul de gradients (méthodes neuronales par exemple).

Les contributions de cet article sont donc les suivantes : nous proposons deux mesures de ressemblance basées sur la forme pour la comparaison de séries temporelles. Nous montrons que ces mesures mais surtout le gradient de la seconde mesure peuvent être évalués en temps sous-quadratique ($O(L\log(L))$). Nous proposons une analyse de KSC, KS et des méthodes de partitionnement proposées dans ce travail.

2 Mesures de ressemblance et méthodes de partitionnement

2.1 Mesures de ressemblance basées sur la forme

On considère ici des séries temporelles de signes quelconques. Les séries ont toutes la même longueur, notée L. On considère l'opérateur de décalage temporel τ_o de paramètre o ($-L+1 \leq o \leq L-1$). La corrélation croisée normalisée prend la forme suivante :

$$CCN(x,y)(o) = \frac{x \cdot \tau_o(y)}{\|x\|\|y\|}$$

la notation "point" correspond au produit scalaire, et $\|x\|^2 = x.x$ (norme L2).

On note g_E la fonction $g_E(cc) = \sqrt{1-cc}$. On rappelle que la distance euclidienne entre x et y, des séries normalisées, est égale à $g_E(CCN(x,y)(0))$ à un coefficient multiplicatif près.

La première mesure d_E que nous proposons prend la forme suivante :

$$d_E(x,y) = \min_o g_E(CCN(x,y)(o))$$

Les mesures utilisées par KSC et KS sont très similaires. Seule la nature de la fonction g change : pour KS, on a $g_{KS}(cc) = 1-cc$, alors que pour KSC, on a $g_{KSC}(cc) = \sqrt{1-cc^2}$. Notons qu'à cause de l'élévation au carré intervenant sur cc, les séries corrélées et anti-corrélées sont considérées comme identiques par d_{KSC}. C'est la raison pour laquelle la méthode KSC a initialement été proposée pour des séries positives (distributions temporelles).

Le calcul de CCN pour toutes les valeurs de décalage o semble a priori quadratique en L. Mais classiquement, on peut ramener cette complexité à $O(L\log(L))$ en évaluant tous les

produits scalaires $x \cdot \tau_o(y)$ grâce à la transformée de Fourier rapide FFT. Si l'on note $\mathcal{F}$ (resp. $\mathcal{F}^{-1}$) la transformée de Fourier (resp. l'inverse de la transformée de Fourier), et z^* le conjugué du complexe z, la corrélation croisée de x et y est proportionnelle à $\mathcal{F}^{-1}\left(\mathcal{F}(x^0)\ \mathcal{F}^*({}^0y)\right)$. x^0 (resp . ${}^0y$) est le vecteur x complété à droite (resp. à gauche) par des zéros. Les auteurs de KS utilisent cette astuce mathématique pour accélérer l'évaluation de leur mesure. KSC et notre mesure d_E peuvent de même bénéficier de cette remarque.

La seconde mesure de ressemblance d_E^γ que nous proposons s'inspire des travaux portant sur le SoftDTW (Cuturi et Blondel (2017)). Ces travaux utilisent la fonction softMin pour évaluer la ressemblance entre deux séries après déformation temporelle. Les auteurs montrent que le calcul du gradient de SoftDTW peut être réalisé efficacement.

Soit g un vecteur de longueur $2L-1$, on note $\min^\gamma(g)$ la fonction SoftMin corrigée :

$$\min{}^\gamma(g_1,\ldots,g_{2L-1}) = -\gamma \log \sum_o e^{-\frac{g_o}{\gamma}} + \gamma \log(2L-1) \qquad \text{pour } \gamma > 0$$

La fonction $\min^\gamma$ corrigée est une f-moyenne généralisée. Elle vérifie la propriété $\min(g) \leq \min^\gamma(g) \leq \min(g) + \gamma \log(2L-1)$. On voit donc que $\min^\gamma$ approche la fonction $\min$ quand le paramètre de régularité γ tend vers 0. $\min^\gamma$ est d'autant plus régulière que γ est grand. Elle est dérivable pour γ strictement positif.

Grâce à la fonction SoftMin, il est aisé de définir la version régularisée de d_E, notée d_E^γ :

$$d_E^\gamma(x,y) = \min{}_o^\gamma g_E(CCN(x,y)(o))$$

d_E^γ est à valeurs positives mais ne vérifie par la propriété de séparation ($d_E^\gamma(x,x) \neq 0$). L'évaluation de d_E^γ peut s'appuyer elle aussi sur la FFT pour une complexité sous-quadratique ($O(L\log(L))$). Montrons à présent que le gradient en x de $d_E^\gamma(x,y)$ peut aussi se calculer efficacement. Après quelques calculs, on obtient le résultat suivant :

$$\frac{\partial d_E^\gamma(x,y)}{\partial x_i} = \frac{\sum_o S_o^\gamma g'_E(CCN(x,y)(o)) y_{i-o}}{\|x\|\|y\|} - \frac{x_i}{\|x\|^2} \sum_o S_o^\gamma g'_E(CCN(x,y)(o)) CCN(x,y)(o)$$

avec g'_E la dérivée de g_E, $S_o^\gamma = SM_o(-g_E(CCN(x,y)/\gamma)$ où $SM_o(z) = \frac{e^{z_o}}{\sum_{o'} e^{z_{o'}}}$ est la fonction exponentielle normalisée.

Les $CCN(x,y)(o)$ se pré-calculent en $O(L\log(L))$. Le premier terme de la dérivée partielle fait intervenir un produit de convolution, et se calcule donc pour tous les i en $O(L\log(L))$ grâce à la FFT. Le second terme est en temps linéaire, car la somme sur o peut être précalculée (indépendante de x_i). L'évaluation de la mesure d_E^γ et de son gradient ont donc des complexités identiques $O(L\log(L))$. Afin d'éviter les débordements numériques dans les calculs, on utilise pour les deux évaluations l'astuce classique de décalage des valeurs (*Log-Sum-Exp trick*).

2.2 Extraction d'un barycentre

On note KE (resp. KSE) la méthode des centres mobiles basée sur d_E (resp. d_E^γ). Comme l'extraction d'un barycentre est utilisée ici comme procédure interne aux centres mobiles, KSC, KS et KE font l'hypothèse que l'alignement optimal o_i^* de chaque série x_i avec le barycentre produit à l'itération précédente a été trouvé lors de la phase d'affectation. Les trois méthodes effectuent donc l'extraction du barycentre avec un décalage temporel fixé.

Bien que KSC et KS ne partagent pas la même mesure de ressemblance, l'extraction du barycentre μ s'effectue en résolvant le même problème d'optimisation. KSC optimise un critère d'inertie cohérent avec la phase d'affectation : $\mu = \arg\min_\mu \sum_i g^2_{KSC}(CCN(\mu, x_i)(o_i^*))$. KS optimise le critère équivalent $\mu = \arg\max_\mu \sum_i CCN^2(\mu, x_i)(o_i^*)$, qui diffère de celui optimisé lors de sa phase d'affectation. Dans le cas de séries à signes quelconques, on remarque que l'extraction du barycentre est susceptible de ne pas distinguer les séries corrélées des séries anti-corrélées (CCN au carré). Dans la pratique, cette remarque a peu de conséquences, car lors de l'affectation, d_{KS} choisit les o_i^* et les classes qui maximisent CCN.

Pour KSC comme pour KS, le calcul du barycentre μ revient à extraire le vecteur propre associé à la plus grande valeur propre d'une matrice (occupation mémoire en $O(L^2)$). Le signe du barycentre doit être déterminé car un vecteur propre peut avoir deux directions. KSC et KS effectuent une diagonalisation complète de la matrice (complexité cubique). Nous avons montré dans un précédent travail (Conan-Guez et al. (2018)), qu'en utilisant un calcul incrémental de la matrice et la méthode des puissances itérées, la complexité observée de KS et KSC était quadratique en L, d'où une réduction importante des temps de calcul.

Pour KE, la phase d'affectation fournit exactement la même partition que celle de KS, en revanche, l'extraction du barycentre est plus simple. En effet, la mesure d_E une fois o_i^* fixé correspond à la distance euclidienne entre les vecteurs unitaires μ et $\tau_{o_i^*}(x_i)$. On résout $\mu = \arg\max_\mu \frac{\mu}{\|\mu\|}.(\sum_i \frac{\tau_{o_i^*}(x_i)}{\|x_i\|})$. μ est donc le vecteur (rendu unitaire) obtenu en sommant les vecteurs $\frac{\tau_{o_i^*}(x_i)}{\|x_i\|}$. L'extraction du barycentre est donc très semblable à celle des centres mobiles classiques avec une complexité linéaire en L. KE reste pour autant en $O(L\log(L))$ du fait de la phase d'affectation. L'occupation mémoire est en $O(L)$, car la FFT est en $O(L)$.

Pour KSE, le barycentre s'obtient par $\mu = \arg\min_\mu \sum_i d_E^\gamma(\mu, x_i)^2$. Le carré est ici maintenu afin de résoudre un problème identique à celui de KE. KSE est la seule méthode qui prend en compte l'alignement temporel dans la phase d'affectation comme dans la phase de représentation. Le fait que d_E^γ soit toujours à valeurs positives permet d'avoir un critère d'inertie bien formé. Extraire le barycentre nécessite cette fois l'utilisation d'une méthode d'optimisation non linéaire (gradient conjugué, BFGS,...). Cette méthode ne nous assure pas l'obtention d'un optimum global (contrairement aux méthodes précédentes). En revanche, d'une itération à l'autre du partitionnement, le déplacement continu des barycentres doit favoriser un nombre réduit de descentes de gradient. Pour la méthode du gradient conjugué, si l'on suppose une indépendance entre la dimension L et le nombre d'évaluations du critère d'inertie, l'occupation mémoire est linéaire en L et le coût de calcul est en $O(L\log(L))$ grâce à l'évaluation efficace de d_E^γ et de son gradient.

3 Expériences

Les différentes méthodes ont été implémentées avec le langage R. Le critère d'arrêt porte sur l'évolution du critère d'inertie intra-classe. Nous utilisons les fonctions `optim("BFGS")` et `powerMethod` (package matlib) pour l'extraction des barycentres de KSE et KS. KS et KE ont été testés avec ou sans centrage des barycentres (Paparrizos et Gravano (2017)).

Afin d'illustrer l'invariance temporelle des méthodes, la figure 1 montre le résultat du partitionnement produit par KE sur un jeu de données artificielles. Ce jeu de données est constitué de 9 séries : 3 gaussiennes, 3 fonctions créneaux et 3 fonctions sinus. Les séries sont déphasées

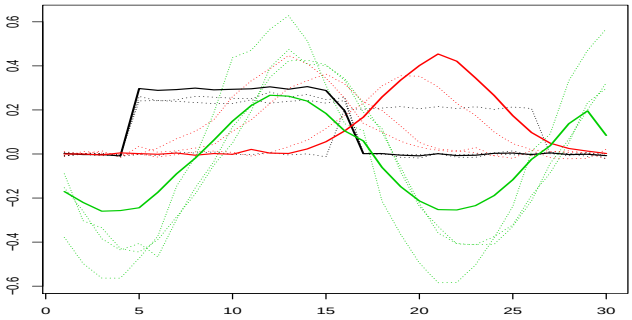


FIG. 1 – *Les séries (pointillés) et les barycentres (lignes) colorés selon la classe*

(décalage temporel) avec un changement d'échelle (amplitude) et bruitées. Les trois classes sont parfaitement retrouvées par KE.

Nous avons comparé KS, KE et KSE sur 5 jeux de données issus de l' *UCR Time Series Classification Archive*. La table 1 contient une description de chaque jeu de données. Chaque jeu est fourni avec un partitionnement réel (colonne "nb classes") qui sera comparé avec les partitionnements produits par les trois méthodes KS, KE et KSE.

	nb séries	longueur	nb classes
ECG5000	500	140	5
WordsSynonyms	267	270	25
Trace	100	275	4
Fish	175	463	7
Haptics	155	1 092	5

TAB. 1 – *Description des cinq jeux de données*

Comme nous ne pouvons par comparer les critères d'inertie intra-classe (ils sont calculés avec des mesures différentes), nous utilisons, comme les auteurs de KS, le Rand-Index pour mesurer l'adéquation entre partition réelle et partition produite. Chaque méthode a été exécutée 10 fois. Les initialisations sont identiques pour toutes les méthodes. La table 2 indique la moyenne des 10 Rand-Index et le temps cumulé des 10 exécutions en secondes.

	ECG5000	WordsSynonyms	Trace	Fish	Haptics
KS	0.750 (59)	**0.892** (176)	0.785 (12)	0.788 (55)	**0.676** (220)
KE	**0.761** (67)	0.891 (185)	0.785 (11)	**0.789** (38)	0.674 (43)
KSE $\gamma = 0.01$	0.744 (107)	0.879 (291)	**0.791** (21)	0.777 (17)	0.666 (249)

TAB. 2 – *moyenne des 10 Rand-Index (temps cumulé en secondes)*

On constate que les performances de KS et KE sont très semblables. KSE est légèrement en retrait, ceci s'explique sans doute par l'existence d'optima locaux. Les résultats pour γ valant 0.1 et 0.001 sont relativement identiques à ceux présentés pour $\gamma = 0.01$. Pour des valeurs plus grandes ($\gamma = 1$), les résultats se dégradent (Rand-Index moyen de 0.384 pour Fish par exemple). Si l'on calcule la moyenne des Rand-Index sur les cinq jeux de données, on obtient 0.769 pour les centres mobiles classiques, 0.772 pour KSE, 0.775 pour K-MeansDTW (sans et avec fenêtre 10%), 0.778 pour KS et 0.780 pour KE.

Une analyse des temps de calcul peut aussi être menée, même si le code R n'a pas été écrit dans un soucis de performance optimale. Pour des longueurs faibles ($L < 300$), KE et KS ont des temps d'exécution identiques : bien que KS ait une complexité observée quadratique alors que KE soit en $O(L\log(L))$, les temps de calcul sont dominés par la phase d'affectation qui est identique pour les deux méthodes. Pour Haptics ($L = 1092$), KE se révèle cette fois 5 fois plus rapide que KS. KSE est bien sûr plus lent dans tous les cas, mais étonnamment la différence n'est pas aussi marquée que l'on aurait pu s'y attendre. Elle s'explique d'une part par la qualité inférieure des solutions, mais aussi peut être par le fait que la phase de représentation de KSE "optimise" l'alignement temporel alors qu'il est maintenu fixe pour KS et KE. KS et KE auraient donc besoin de plus d'itérations (affectation, représentation) pour stabiliser les alignements. Pour ECG5000, KS et KE nécessitent respectivement 138 et 171 itérations (cumulées sur les 10 exécutions), alors que KSE converge en 95 itérations.

Pour conclure, KE est plus simple à implémenter que KS et produit des résultats similaires en un temps plus court. La mesure d_E^γ obtient des résultats honorables et peut être adaptée avantageusement à des méthodes nécessitant des calculs de gradient (méthodes neuronales,...).

Références

Conan-Guez, B., A. Gély, L. Boudjeloud-Assala, et A. Blansché (2018). K-spectral centroid : extension and optimizations. In *26th European Symposium on Artificial Neural Networks, ESANN 2018, Bruges, Belgium, April 25-27, 2018*.

Cuturi, M. et M. Blondel (2017). Soft-DTW : a differentiable loss function for time-series. In *In 34th International Conference on Machine Learning*, Volume 70, pp. 894–903.

Paparrizos, J. et L. Gravano (2017). Fast and accurate time-series clustering. *ACM Trans. Database Syst. 42*(2), 8 :1–8 :49.

Yang, J. et J. Leskovec (2011). Patterns of temporal variation in online media. In *Proc. of the fourth ACM international conf. on Web search and data mining*, pp. 177. ACM.

Summary

In this work, we propose two k-means-like methods, devised for time series clustering. Each method relies on a custom dissimilarity measure between time series, which is invariant to time shifting and Y-scaling. The first measure is an adaptation of the cosine dissimilarity for which the best time alignment is obtained by testing all temporal translations. The second measure is a soft version of the first measure: the min computation on the different time alignments is carried out by the soft min function.

Classification de variables : une approche dynamique en grande dimension

Christian Derquenne*

*EDF R&D - 7, boulevard Gaspard Monge - 91120 Palaiseau - France
christian.derquenne@edf.fr

Résumé. La recherche de structures dans les données représente une aide essentielle pour comprendre les phénomènes à analyser. Les méthodes de classification de variables permettent de répondre à cette problématique, mais elles peuvent être pénalisées par un trop grand nombre de variables. Nous proposons une nouvelle approche de type "Diviser pour Régner" fondée sur le principe MapReduce pour pallier ce problème. La table de données est divisée en plusieurs sous-tableaux traités en parallèle, puis réconciliés à l'aide de l'Analyse des Correspondances Multiples. Cette approche est appliquée sur des données simulées et fournit de très bons résultats.

1 Contexte - objectif

La recherche exploratoire de structures dans les données est essentielle dans de nombreuses applications (biologie, environnement, finance, management de l'énergie, ...) afin de comprendre les comportements des individus, les liens entre les variables, ... Les outils de visualisation, de réduction de dimension, de recherche de patterns permettent de répondre efficacement à ce type de problématiques. Nous nous plaçons dans cadre de classification non supervisée et plus particulièrement dans le domaine de la classification de variables numériques ayant des liens quelconques (linéaires ou non linéaires). Plusieurs approches ont été proposées pour répondre à cette problématique. Les principales reposent sur la réduction de l'espace factoriel en associant au mieux les variables initiales à de nouvelles composantes (Sarle, 1990, Vigneau et al., 2003, Chavent et al., 2011, Bühlmann et al., 2013, Chen M., 2014, Chen Y. et al., 2016). Nous avons développé une méthode nommée "double critère contrôlé dynamique" disponible pour des liens linéaires (Derquenne, 2016). Celle-ci est fondée simultanément sur un test d'indépendance linéaire simple entre les variables initiales et/ou des variables latentes (première composante principale de l'ACP) et un test d'unidimensionnalité sur les classes obtenues afin de construire une typologie de façon dynamique au moyen du contrôle du nombre de groupes et de leur qualité. Cette méthode a été étendue pour des relations quelconques (Derquenne, 2017). Cette approche est fondée sur des transformations polynomiales entre couple de variables initiales et/ou variables latentes (première composante principale issue d'une ACP non linéaire). Les résultats obtenus à l'aide de ces deux approches sont très satisfaisants comparées à d'autres méthodes qui peinent à détecter le "bon" nombre de classes et le "bon" contenu de celles-ci validés à partir de tests sur des données simulées.

En effet, les méthodes existantes sont plus ou moins performantes, en termes de qualité de la typologie obtenue (compacité, isolation, "bon" nombre de classes et "bon" contenu des groupes). Il en est de même pour le temps de calcul et la taille mémoire requise. Pour des nombres raisonnables d'observations et de variables ($n < 10000$ et $p < 100$), les algorithmes fonctionnent bien sur ces deux aspects. Par contre, si n et p deviennent très grands, alors le temps de calcul et la capacité mémoire deviennent trop élevés. Pour pallier ce problème, une stratégie du type "diviser pour mieux régner" peut alors être adéquate pour traiter cet aspect "grande dimension". Nous posons tout d'abord deux postulats et nous proposons une méthode pour résoudre ce problème. Puis, nous appliquons celle-ci sur des jeux de données simulées en grande dimension afin d'évaluer ses performances. Enfin, nous concluons sur les améliorations à apporter, les applications potentielles et les voies futures.

2 Problème, postulat et proposition

Comme indiqué dans la section précédente, une trop grande taille de la table des données (nombre de variables et nombre d'individus) peut affecter la performance des algorithmes et donc pénaliser la qualité des résultats obtenus. L'objectif de ce papier est de proposer une approche statistique afin de répondre à la question suivante : "Comment tenir compte de la grande dimension lorsque que nous appliquons une méthode statistique classique ?". Pour cela, nous posons deux postulats.

Postulat 1 : Si une méthode fournit de bons résultats sur un échantillon tiré de la table de données entière, il n'y a pas de raison que cette méthode donne de mauvais résultats sur un autre échantillon tiré de la même grande base de données.

Postulat 2 : Si nous combinons les résultats d'un nombre d'échantillons issus de la grande table de données à l'aide d'un processus adéquat, alors les résultats agrégés devraient être comparables au résultat global provenant de l'application de la méthode sur l'ensemble de la base de données.

Ce processus est fondé sur le principe : "Diviser et Conquérir" (DCP). Le principe général est le suivant : (i) la table de données entière est découpée en S échantillons ; (ii) chaque échantillon est traité en parallèle à l'aide de la méthode choisie ; (iii) les résultats sont combinés et ils sont traités selon une procédure adéquate ; (iv) le résultat final est obtenu. Ce principe est fondé sur l'approche **MapReduce** qui est une méthode générique pour traiter des bases de données massives distribuées sur de nombreux fichiers systèmes. Elle a été développée par Google^{TM} (Dean et al., 2004). Après avoir présenté brièvement la démarche de la méthode de classification de variables introduite en 2016 et 2017, nous développerons l'approche DCP associée à celle-ci. Il faut cependant préciser que ce processus fonctionne à condition que le processus de divison du tableau de données et la stratégie d'agrégation soient correctes.

2.1 Une approche dynamique pour la classification de variables

Soient $X_1, ..., X_p$, p variables numériques dont on suppose que les relations sont linéaires ou absentes, alors la première étape consiste à agréger les deux variables les plus corrélées linéairement pour constituer la première classe. Pour cela, on fixe un seuil critique du test de corrélation (par exemple, $\alpha_\rho = 0,05$), alors si la plus petite p-valeur parmi les $p(p-1)/2$ couples

de variables est inférieure à α_ρ, on regroupera ces deux variables. Puis la première composante principale est calculée sur celles-ci, soit Z_1. De nouvelles corrélations sont calculées entre Z_1 et les $p-2$ variables restantes. Trois cas peuvent se présenter : soit une classe de trois variables, soit deux classes de deux variables, soit aucune corrélation significative est trouvée, alors l'algorithme s'arrête. Dans ce dernier cas, il y aura un groupe de deux variables et $p-2$ classes singleton. Si le processus continue, dès qu'un groupe possède au moins trois variables, un test d'unidimensionnalité est pratiqué sur la deuxième valeur propre, tel que $H_0 : \lambda_2 \leq 1$ (Saporta, 1999). Si l'hypothèse nulle d'unidimensionnalité est rejetée, alors on recherche si parmi les p-valeurs restantes issues des tests de corrélations, la plus petite est inférieure au seuil fixé. Si c'est le cas, les trois possibilités indiquées précédemment se représenteront. Le processus de constitution des classes se poursuit jusqu'à ce que plus aucune p-valeur de corrélation est inférieure à α_ρ et que le test d'unidimensionnalité pour chaque classe est rejeté. A la fin de ce processus, nous obtenons M classes. Cette approche a été généralisée pour des liens quelconques entre variables (Derquenne, 2017).

2.2 Extension de la classification de variables en grande dimension

Nous nous plaçons dans le contexte suivant. Soient $X_1, ..., X_p$, p variables numériques, telles que $X_j \in \mathbb{R}^n$ où $n \gg 10000$ est le nombre d'individus contenus dans la grande base de données E.

Soient $E_1, ..., E_s, ..., E_S$, S échantillons aléatoires tirés sans remise de n individus, tel que $E = \cup_{s=1}^{S} E_s$ où card$(E_s) = n_s$ et $\sum_{s=1}^{S} n_s = n$.

Chaque E_s est découpé en L sous-échantillons aléatoires sans remise de variables tirés parmi les p variables initiales, tels que $Q_1, ..., Q_l, ..., Q_L$ où card$(Q_l) = p_l$ et $\sum_{l=1}^{L} p_l = p$.

Remarque 1 : Le découpage en L échantillons aléatoires de variables peut être différent ou non pour chaque échantillon d'individus : E_s.

Enfin, T_{sl} correspond à la sous-table de données de n_s individus et de p_l variables.

Le processus détaillé DCP adapté à la classification de variables se déroule de la façon suivante.

(i) Sur chaque sous-ensemble de données T_{sl}, l'approche dynamique de classification de variables est appliquée, alors M_{sl} classes sont obtenues, ainsi que M_{sl} premières composantes principales associées : $Y_1^{(sl)}, ...Y_{M_{sl}}^{(sl)}$. Cette étape (i) est réalisée en parallèle sur chaque T_{sl} de E_s. Il s'agit de la phase **Map**. Cela permet d'obtenir un ensemble de premières composantes principales pour l'échantillon E_s : $Y_1^{(s_1)}, ..., Y_{M_{s_1}}^{(s_1)}, ..., Y_1^{(s_l)}, ..., Y_{M_{s_l}}^{(s_l)}, ..., Y_1^{(s_L)}, ..., Y_{M_{s_L}}^{(s_L)}$.

Remarque 2 : Lorsque n est grand peut conduire à des $p$valeurs très faibles et donc des corrélations entre variables très significatives. Cependant le découpage adéquat en un nombre d'individus relativement faible (≤ 10000) permet généralement de pallier ce problème. Plus de détails sur le choix du seuil α_ρ sont fournis dans (Derquenne, 2016).

(ii) Sur ces premières composantes principales, l'approche dynamique de classification de variables est appliquée et fournit alors M_s nouveaux groupes : $(C_1^{(s)}, ..., C_k^{(s)}, ..., C_{M_s}^{(s)})$ et M_s nouvelles premières composantes principales associées : $Z_1^{(s)}, ..., Z_k^{(s)}, ..., Z_{M_s}^{(s)}$. Cette nouvelle classification permet à chaque variable initiale de départ X_j d'appartenir à une classe

$C_k^{(s)}$ parmi les M_s classes construites précédemment. Ces attributions de variables à des classes permettent de construire une nouvelle variable V_s contenant pour chaque variable X_j le numéro de sa classe parmi les M_s groupes. Cette étape (ii) est réalisée en parallèle sur chaque E_s. Il s'agit de la phase **Reduce**.

(iii) L'objectif de cette étape et la suivante est de réconcilier l'ensemble des résultats issus de (i) et (ii) afin d'obtenir une classification globale pour l'ensemble des S échantillons E_s d'individus et l'ensemble des p variables X_j. En effet, lorsque les étapes (i) et (ii) sont terminées, nous disposons de S variables catégorielles : $V_1, ..., V_s, ..., V_S$ contenant les numéros de classes de chaque variable X_j. Si, comme on le suppose, les résultats de classification issus de chaque échantillon E_s d'individus se ressemblent alors les comportements devraient être similaires. En d'autres termes, les variables $V_1, ..., V_s, ..., V_S$ devraient être dépendantes. Un moyen de mesurer cette dépendance est d'appliquer une analyse des correspondances multiples (ACM) sur ces variables pour lesquelles les individus sont simplement les variables initiales $X_1, ..., X_j, ..., X_p$. Les résultats de l'ACM fournissent R composantes principales $U_1, ..., U_r, ...U_R$. S'il y a une structure de groupes de variables X_j dans les données, alors cela devraient se retrouver dans l'espace des individus (les variables initiales) de l'ACM. Signalons que cette étape revient à réaliser un consensus de partitions de façon très simple.

(iv) Cette ultime étape permet de voir s'il y a ou pas une structure de groupes. Pour cela, nous classifions les individus (les variables X_j) à partir des composantes principales de l'ACM. Toutes celles-ci peuvent être retenues ou il est possible de sélectionner celles qui rassemblent par exemple, 80% de l'inertie expliquée. Par ailleurs, nous avons choisi d'utiliser une approche de classification hiérarchique ascendante au moyen du critère de Ward. Les résultats obtenus fournissent M classes : $G_1, ..., G_m, ..., G_M$ contenant respectivement $p_1, ..., p_m, ..., p_M$ variables initiales $X_1, ..., X_j, ..., X_p$, tel que $\sum_{m=1}^{M} p_m = p$ et contenant les n individus.

2.3 Evaluation de la classification en grande dimension

Les résultats obtenus à l'aide de l'approche proposée précédemment doivent être évalués. Pour cela, nous utilisons deux niveaux de validation.

Le premier évalue la qualité de reconstitution de la classification observée sur l'ensemble de la table des données. Nous comparons l'inertie expliquée de la classification observée (OCI) de (1) et celle de la classification estimée (ECI) de (2) à l'aide de l'approche proposée.

$$OCI = \frac{1}{p} \sum_{m=1}^{\tilde{M}} \sum_{X_j \in \tilde{G}_m} \rho^2(X_j, \tilde{Z}_m) \text{ (1)} \quad ECI = \frac{1}{p} \sum_{m=1}^{M} \sum_{X_j \in G_m} \rho^2(X_j, Z_m) \text{ (2)}$$

où $\tilde{G}_m$ et G_m sont respectivement les classes observées (simulées) et estimées, $\tilde{Z}_m$ et Z_m sont les $\tilde{M}$, respectivement les M premières composantes principales observées et estimées. Généralement ECI est inférieure à OCI. Plus ECI est proche d'OCI, plus la qualité de reconstitution de la classification observée est bonne.

Le second niveau permet d'évaluer la qualité de la classification estimée. Pour cela nous comparons les contenus des typologies observée et estimée à partir de leur tableau de contigence dont les lignes et les colonnes correspondent aux numéros des classes. Les indices de Rand, de Jaccard, γ, le T de Tchuprow, le V de Cramer et le pourcentage de bien classés permettent d'évaluer la qualité de la typologie estimée. Ces indices varient entre 0 et 1, plus la valeur obtenue est proche de l'unité, plus l'adéquation est bonne.

3 Application de l'approche de classification de variables en grande dimension et comparaisons

Afin d'évaluer la qualité de l'approche proposée, nous avons simulé un jeu de données possédant 100000 observations et 1000 variables. Celui-ci est découpé en 9 classes, tel que : $G_1 = \{X_1, X_{101}, ..., X_{800}\}$ avec $X_j = X_1 + 2\epsilon_t$ où $X_1 \rightsquigarrow \mathcal{N}(0,1)$, $G_2 = \{X_9, X_{10}, ..., X_{500}\}$ avec $X_j = 2X_9 + 2\epsilon_t$ où $X_9 \rightsquigarrow \mathcal{N}(0,1)$, $G_3 = \{X_5, X_{801}, ..., X_{1000}\}$ avec $X_j = 0,1X_5 + \epsilon_t$ où $X_5 \rightsquigarrow \mathcal{N}(0,1)$ et $\epsilon_t \rightsquigarrow \mathcal{N}(0,1)$, $G_4 = \{X_2\}, G_5 = \{X_3\}, G_6 = \{X_4\}, G_7 = \{X_6\}, G_8 = \{X_7\}, G_9 = \{X_8\}$.

La construction des sous-tables de variables et d'individus se déroule de la façon suivante. 10 échantillons issus de tirage sans remise E_s de 10000 individus sont constitués, puis chacun d'eux est découpé en 10 sous-échantillons Q_l de variables de taille 100 sont également tirés au hasard sans remise. Signalons que chaque Q_l pour $l = 1, 10$ possède les mêmes variables (cf. remarque de 2.2). Par conséquent, le processus **MapReduce** calcule 100 classifications initiales en parallèle, puis 10 classifications globales en parallèle, une ACM et une classification finale. Cela correspond à 112 tâches.

Le tableau de contingence (table 1) croise les typologies observée (en colonne) et estimée (en ligne). Les classes 1 et 2 de la classification estimée regroupent exactement les classes 1 et 2 de la typologie observée. Il en est de même pour la classe 3 calculée qui correspond à la classe 4 observée. Les classes 4 et 6 regroupent des variables séparées à l'origine. Les classes 5 et 7 sont relatives à la classe 3 observée. Par ailleurs, les résultats des indices sont satisfaisants. En effet, ECI est très légèrement inférieure à OCI : 0,3177 vs 0,3226. Les indices d'adéquation confirment ce bon résultat : $T = 0,6576$, $V = 0,7593$, $\gamma = 0,9867$, $Rand = 0,9936$ et $Jaccard = 0,9829$. Enfin, le pourcentage de bien classés vaut 79,4%.

Est/Obs->	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
1	301	0	0	0	0	0	0	0	0
2	0	492	0	0	0	0	0	0	0
3	0	0	0	1	0	0	0	0	0
4	0	0	0	0	0	1	1	0	0
5	0	0	185	0	1	0	0	0	0
6	0	0	1	0	0	0	0	0	1
7	0	0	15	0	0	0	0	1	0

TAB. 1 – *Tableau de contingence des typologies observée et estimée*

La CPU est divisée par 6 pour un ordinateur à deux coeurs entre l'application de la méthode classification dynamique de variables sur l'ensemble du jeu de données et l'application de la méthode à grande dimension proposée (200% de gain).

4 Apports, applications et voies futures

La méthode proposée est fondée sur l'approche dynamique qui offrait déjà une bonne qualité de résultats (2016, 2017). L'extension de celle-ci à la grande dimension préserve l'inertie globale et le contenu des classes (études de simulation), et fournit de bonnes performances en termes de temps de calcul, même dans le cas de processus séquentiel. Des fonctions R ont été développées. D'autres applications sur données simulées et réelles ont fourni des résultats satisfaisants. Les voies futures sont : intégration d'autres méthodes de classification (CLV, ClustOfVar, ...) en DCP, comparaison avec des méthodes de classification en grande dimension, utilisation d'autres stratégies de classification dans l'étape (iv) que Ward et étude de leur impact sur le temps calcul, plus de simulations afin de valider l'approche dans différents cas de figure (taille et structure des classes, force de corrélation entre variables, ...), traitements sur des données plus complexes, utilisation d'autres critères d'évaluation de la classification estimée et développement d'une méthode de co-clustering pour des données en grande dimension.

Bibliographie

Bühlmann P., Rütimann P., van de Geer S., and Zhang C-H, (2013). Correlated in regression : Clustering and sparse estimation. *Journal of Stat. Planning and Inference*, **143**(11).

Chavent M., Kuentz V., Liquet B. et Saracco J., (2011). Classication de variables : le package ClustOfVar, *43ièmes Journées de Statistique*, Tunis, Tunisie.

Chen M., (2014). *Classification de variables autour de variables latentes avec filtrage de l'information : application à des données en grande dimension*, Thèse, Univ. Nantes.

Chen Y. and Yang U., (2016). A Novel Information-Theoretic Approach for Variable Clustering and Predictive Modeling Using Dirichlet Process Mixtures, www.nature.com/scientificreports.

Dean, J. and Ghemawat, S., (2004). MapReduce : simplified data processing on large clusters. In Proceedings of Sixth Symposium on Operating System Design and Implementation.

Derquenne Ch., (2016). Classification de variables : une approche à double critères contrôlés dynamiques, *48ièmes Journées de Statistique*, Montpellier, France.

Derquenne Ch., (2017). Classification de variables avec des relations non linéaires, *49ièmes Journées de Statistique*, Avignon, France.

Saporta G., (1999). Some Simple Rules for interpreting Outputs of Principal Components and Correspondence Analysis, *IXth International Symposium on ASMDA*, Lisbon, Portugal.

Sarle W., (1990). *The VARCLUS Procedure. SAS/STAT 9.2 User's Guide*. Cary, NC : SAS.

Vigneau E. and Qannari E.M., (2003). Clustering of Variables Around Latent Components. *Communications in Statistics - Simulation and Computation*, **32**(4), 1131-1150.

Summary

The methods of clustering of numerical variables represents an essential help for the search for structures in the data, but they can be penalized by too many variables. We propose a new "Divide and Conquer" approach based on the MapReduce principle to overcome this problem. The data table is divided into several sub-tables processed in parallel, then reconciled using the MCA. This approach is applied to simulated data and provides very good results.

Formal Concept Analysis for Identifying Biclusters with Coherent Sign Changes

Nyoman Juniarta*, Miguel Couceiro*, Amedeo Napoli*

*Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
nyoman.juniarta@loria.fr

Abstract. In this paper we are studying the task of finding coherent-sign-changes biclusters in a binary matrix. This task can be applied to the interpretation of gene expression data, where such a bicluster represents a set of experiments that affect a set of genes in a consistent way. Our approach is purely symbolic. We start with a binary table and study biclustering methods based on FCA and partition pattern structures. Pattern concepts provide biclusters and their hierarchical relation, which can be used to analyze the profile of genes in the given expression data.

1 Introduction

Gene expression data can be represented as a matrix, where rows and columns represent genes and experiments respectively. Each cell contains the numeric expression level of a given gene under a given experiment. In such data, we can say that an experiment affect a gene by either lowering or raising its expression, according to the gene's normal level. One may be interested in finding a subset of genes and a subset of experiments, such that the experiments affect the genes in a consistent way. In other words, any two experiments in the subset have always either the same effect or the opposite effect on every gene in the subset. This task corresponds to the mining of coherent-sign-changes (`CSC`) biclusters.

Biclustering is an important technique aimed at discovering patterns in a matrix representing a dataset. It is related to standard clustering whose main objective is to group the rows based on their similarity. On the other hand, biclustering refers to the problem of discovering submatrices whose cells exhibit similar behavior. This problem is also called co-clustering Govaert and Nadif (2013), where rows and columns are clustered simultaneously.

In this paper, we present a method based on FCA and pattern structures for discovering a specific type of bicluster: coherent-sign-changes bicluster. An existing approach in Tanay et al. (2002) can mine this bicluster type, but it is statistical, since its discovery of `CSC` biclusters is based on the magnitude of the expression changes. Our approach is more symbolic, by taking into account only the direction of the changes, with expectation of detecting larger biclusters.

TAB. 1 – *Examples of two bicluster types: (a) constant-values and (b) coherent-sign-changes (CSC).*

2	2	2	2	+	+	−
2	2	2	2	+	+	−
2	2	2	2	−	−	+
2	2	2	2	−	−	+
(a)				(b)		

2 Related Work

In the domain of gene expression data, the algorithm called SAMBA was proposed in Tanay et al. (2002) to discover a submatrix where the expressions of a subset of genes significantly changes across a subset of conditions. The first model of SAMBA searches a submatrix where there is a *joint* change across all genes, without looking whether it is an increase or a decrease. The second model takes into account the direction of the change, such that any two conditions in the submatrix have either always the same effect or always the opposite effect. We call this type of submatrix a coherent-sign-changes bicluster, as denoted in Madeira and Oliveira (2004).

Regarding bicluster discovery based on FCA, several methods were proposed. In a binary matrix, dense approximate bicluster discovery was studied in Gnatyshak et al. (2012); Ignatov et al. (2012) based on standard FCA. This is similar to mining formal concept, but instead of "exact" concepts, the authors relax the problem such that the "approximate" concepts (having a certain amount of empty cells) can also be detected. For biclustering with similar values in a numerical matrix, Kaytoue et al. (2011) proposed standard FCA with scaling and interval pattern structures.

3 Biclustering

We consider that a dataset is composed of a set of objects G, each of which has values over a set of attributes M. This dataset can be represented as a numerical context (G, M, I) where G is a set of objects, M is a set of attributes, and I corresponds to $m(g)$, which is the value of $m \in M$ for object $g \in G$.

One may be interested in finding which subset of objects possesses the same values w.r.t. a subset of attributes. Regarding the matrix representation, this is equivalent to the problem of finding a submatrix that has a constant value over all of its elements (example in Table 1a). This task is called biclustering with constant values, which is a simultaneous clustering of the rows and columns of a matrix.

In coherent-sign-changes (CSC) bicluster, the matrix is binary. In this bicluster, each row is correlated (either entirely identical or entirely opposite) to all other rows. In the example in Table 1b, the first row is identical to the second and opposite to the third and fourth. We can also see this bicluster by comparing the columns. In the example, the first column is identical to the second and opposite to the third.

In a binary dataset (G, M, I), given a set of objects $A \subseteq G$ and an attribute $m \in M$, $m(A)$ is the *column submatrix* formed by the attribute m over A. The submatrix $m_j(A)$ is equal

TAB. 2 – *Running example of five objects and four attributes.*

	m_1	m_2	m_3	m_4
g_1	$+$	$+$	$-$	$-$
g_2	$+$	$+$	$-$	$-$
g_3	$-$	$-$	$+$	$-$
g_4	$+$	$+$	$+$	$+$
g_5	$-$	$-$	$-$	$-$

to $m_k(A)$, denoted as $m_j(A) \simeq m_k(A)$, if all rows in $m_j(A)$ are either entirely identical or entirely opposite to the corresponding rows in $m_k(A)$. With the previous notation, given a binary dataset (G, M, I), a pair (A, B) (where $A \subseteq G$, $B \subseteq M$) is a *coherent-sign-changes bicluster* if $\forall m_j, m_k \in B : m_j(A) \simeq m_k(A)$.

4 The Pattern Structures of Signed Partition

4.1 Formalization

In the task of CSC bicluster discovery in a formal context (G, M, I), we propose an approach based on partition pattern structures. Instead of partition of objects in G as described in Baixeries et al. (2014); Codocedo and Napoli (2014), here we use *partition of attributes* in M. It is still similar to an object partition since an attribute partition covers every attribute in M and there is no overlapping between any two partition components.

To formally define our signed partition, first we define the notion of signed attribute and signed partition component as follows.

Definition 1 (Signed attribute)**.** Let M be a set of attributes, $m \in M$ be an attribute, and $* \in \{-, +\}$ be a sign. A *signed attribute* m^* is an attribute m having a sign $*$.

Definition 2 (Signed partition component)**.** A *signed partition component* (or *sp-component*) c is a subset of M, where each attribute in c is associated to their corresponding sign $*$. Therefore, $c = \{m_1^*, \cdots, m_n^*\}$.

For example, m_1^+ is a signed attribute where the sign $+$ is given to m_1, and $\{m_1^+, m_2^-, m_4^+\}$ is a signed partition component. Since an sp-component contains not only attributes but also their associated sign, we define the equality of two sp-components according to these two aspects as follows.

Definition 3 (SP-component equality)**.** Any two sp-components are equal iff both contain the same set of attributes, and they have either entirely same sign or entirely opposite sign.

Therefore, if we have $c_1 = \{m_1^+, m_2^-, m_4^+\}$, $c_2 = \{m_1^+, m_2^-, m_4^+\}$, and $c_3 = \{m_1^-, m_2^+, m_4^-\}$, then $c_1 = c_2 = c_3$.

Definition 4 (Signed partition)**.** A *signed partition* (or *s-partition*) d is a collection of sp-components, written as $d = \{c_1, \cdots, c_n\}$, such that every attribute in M is present in exactly one sp-component.

For example, given $M = \{m_1, \cdots, m_4\}$, then $\{\{m_1^+, m_2^-, m_4^+\}, \{m_3^+\}\}$ is a valid signed partition of M. The set of all possible s-partitions is denoted as D. This allows us to create an s-partition mapping $\delta : G \to D$ which assigns an object to an s-partition over M. For an object m, $\delta(m)$ is an s-partition containing only one sp-component. This sp-component contains all attributes in M with the corresponding sign according to the object g. Example from Table 2:

$$\begin{aligned}
\delta(g_1) = \delta(g_2) &= \{\{m_1^+, m_2^+, m_3^-, m_4^-\}\} \\
\delta(g_3) &= \{\{m_1^-, m_2^-, m_3^+, m_4^-\}\} \\
\delta(g_4) &= \{\{m_1^+, m_2^+, m_3^+, m_4^+\}\} \\
\delta(g_5) &= \{\{m_1^-, m_2^-, m_3^-, m_4^-\}\}.
\end{aligned}$$

Notice that since the sp-components in $\delta(g_4)$ and $\delta(g_5)$ contain the same attributes with entirely opposite sign, according to Def. 3 we have $\delta(g_4) = \delta(g_5)$. This mapping is formulated as follows:

$$\begin{aligned}
\delta(g) &= \{\{m_j^{*_j} | m_j \in M\}\} \\
&\text{where } *_j = m_j(g).
\end{aligned} \tag{1}$$

4.2 Signed Partition Space

For the task of CSC bicluster discovery, here we define relations between any two s-partitions. The set of all possible s-partitions D is a meet-semilattice where we can define the meet of any two s-partitions.

First, we define the notation $m(c)$ as the sign of an attribute m in an sp-component c. For example, if $c = \{m_1^+, m_2^-, m_3^-\}$, then $m_1(c) = +$. With this notation, we define the similarity ($\cap^\pm$) between any two sp-components as:

$$\begin{aligned}
c_1 \cap^\pm c_2 = \{&\{m_j^* \in c_1 | m_j(c_1) = m_j(c_2)\}, \\
&\{m_j^* \in c_1 | m_j(c_1) = \neg m_j(c_2)\}\},
\end{aligned} \tag{2}$$

where $*$ corresponds to the sign of m_j in c_1, i.e. $m_j(c_1)$.

In other words, the operator $\cap^\pm$ between c_1 and c_2 gives $\{c_{12}, c_{1|2}\}$. The c_{12} represents all attributes who are present in c_1 and c_2 with the same sign, while $c_{1|2}$ represents all attributes who are present in c_1 and c_2, but with opposite sign. The signs in the resulting sp-component are the same as those in the first sp-component. Example:

$$\begin{aligned}
\text{if } c_x &= \{m_1^+, m_2^-, m_3^-, m_4^-\} \\
\text{and } c_y &= \{m_1^+, m_2^-, m_3^+, m_4^+, m_5^-\}, \\
\text{then } c_x \cap^\pm c_y &= \{\{m_1^+, m_2^-\}, \{m_3^-, m_4^-\}\}.
\end{aligned}$$

Since the signs in $c_{1|2}$ follow the first sp-component, the result of $c_1 \cap^\pm c_2$ could be different to $c_2 \cap^\pm c_1$. This can be resolved by Def. 3 that ensures the commutativity of $\cap^\pm$. For example:

$$\begin{aligned}
c_x \cap^\pm c_y &= \{\{m_1^+, m_2^-\}, \{m_3^-, m_4^-\}\}, \\
c_y \cap^\pm c_x &= \{\{m_1^+, m_2^-\}, \{m_3^+, m_4^+\}\}, \\
c_x \cap^\pm c_y &= c_y \cap^\pm c_x.
\end{aligned}$$

TAB. 3 – *Some sign partition pattern concepts from Table 2 and their corresponding* CSC *biclusters.*

Concept		CSC bicluster	
Extent	Intent	Objects	Attributes
$\{g_4, g_5\}$	$\{\{m_1^+, m_2^+, m_3^+, m_4^+\}\}$	$\{g_4, g_5\}$	$\{m_1, m_2, m_3, m_4\}$
$\{g_1, g_2, g_3\}$	$\{\{m_1^+, m_2^+, m_3^-\}, \{m_4^-\}\}$	$\{g_1, g_2, g_3\}$	$\{m_1, m_2, m_3\}$
		$\{g_1, g_2, g_3\}$	$\{m_4\}$
$\{g_1, g_2, g_4, g_5\}$	$\{\{m_1^+, m_2^+\}, \{m_3^+, m_4^+\}\}$	$\{g_1, g_2, g_4, g_5\}$	$\{m_1, m_2\}$
		$\{g_1, g_2, g_4, g_5\}$	$\{m_3, m_4\}$

Having defined the similarity of any two sp-components, we can now define the similarity of any two s-partitions. The similarity (or the meet) of two s-partitions $d_1 = \{c_1 \cdots c_k\}$ and $d_2 = \{c_1 \cdots c_n\}$, with $k = |d_1|$ and $n = |d_2|$, is defined as:

$$d_1 \sqcap d_2 = \{c_i \cap^{\pm} c_j | \forall c_i \in d_1, c_j \in d_2\}, \tag{3}$$

and the order between two s-partitions is given by:

$$d_1 \sqsubseteq d_2 \iff d_1 \sqcap d_2 = d_1. \tag{4}$$

Let C the set of all sp-components in M, and D is the set of all s-partitions in M. We have $\cap^{\pm} : C^2 \to D$ and $\sqcap : D^2 \to D$. Example from Table 2:

$$\begin{aligned} \delta(g_1) \sqcap \delta(g_3) &= \{\{m_1^+, m_2^+, m_3^-, m_4^-\}\} \sqcap \{\{m_1^-, m_2^-, m_3^+, m_4^-\}\} \\ &= \{\{m_4^-\}, \{m_1^+, m_2^+, m_3^-\}\}. \end{aligned}$$

Suppose that $d_1 = \{\{m_4^-\}, \{m_1^+, m_2^+, m_3^-\}\}$. Then $d_1 \sqsubseteq \delta(g_1)$, $d_1 \sqsubseteq \delta(g_2)$, and $d_1 \sqsubseteq \delta(g_3)$.

4.3 Signed Partition Pattern Structures

A *signed partition pattern structure* is determined by the triple $(G, (D, \sqcap), \delta)$, where the derivation operators for $A \subseteq G$ and $d \in D$ are defined as:

$$A^{\square} = \prod_{g \in A} \delta(g), \tag{5}$$

$$d^{\square} = \{g \in G | d \sqsubseteq \delta(g)\}. \tag{6}$$

(A, d) is a *signed partition pattern concept* (or *spp-concept*) when $A^{\square} = d$ and $d^{\square} = A$. From an spp-concept (A, d), a CSC bicluster is any pair (A, c) where $c \in d$ (we can ignore the attribute signs in c). Some spp-concepts from Table 2 are listed in Table 3. In the concept $(\{g_1, g_2, g_3\}, \{\{m_1^+, m_2^+, m_3^-\}, \{m_4^-\}\})$ for example, we can find the CSC bicluster $(\{g_1, g_2, g_3\}, \{m_1, m_2, m_3\})$. Looking back to the original table, this CSC bicluster means that in $A = \{g_1, g_2, g_3\}$, we have $m_1(A) \simeq m_2(A) \simeq m_3(A)$ (recall the definition of $\simeq$ in Section 3).

5 Conclusion

In this paper we have presented an approach to mine biclusters with coherent sign changes in a binary matrix. We formulated our method based on partition pattern structures. A research perspective is the possibility of a matrix that has another sign in addition to $+$ and $-$. This new sign can represent a missing value, or in the case of threshold-based transformation, a value that is equal to the threshold.

References

Baixeries, J., M. Kaytoue, and A. Napoli (2014). Characterizing functional dependencies in formal concept analysis with pattern structures. *Annals of Mathematics and Artificial Intelligence 72*, 129–149.

Cheng, Y. and G. M. Church (2000). Biclustering of expression data. In *ISMB*, Volume 8, pp. 93–103.

Codocedo, V. and A. Napoli (2014). Lattice-based biclustering using partition pattern structures. In *Proceedings of the Twenty-first European Conference on Artificial Intelligence*, pp. 213–218. IOS Press.

Gnatyshak, D., D. I. Ignatov, A. Semenov, and J. Poelmans (2012). Analysing online social network data with biclustering and triclustering. In *Proceedings of the "Concept Discovery in Unstructured Data" conference*, Volume 871, pp. 30–39. Citeseer.

Govaert, G. and M. Nadif (2013). *Co-clustering*. Wiley-IEEE Press.

Hartigan, J. A. (1972). Direct clustering of a data matrix. *Journal of the american statistical association 67*(337), 123–129.

Ignatov, D. I., S. O. Kuznetsov, and J. Poelmans (2012). Concept-based biclustering for internet advertisement. In *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on*, pp. 123–130. IEEE.

Kaytoue, M. (2011). *Traitement de données numériques pas analyse formelle de concepts et structures de patrons*. Ph. D. thesis, Université Henri Poincare – Nancy 1.

Kaytoue, M., S. O. Kuznetsov, J. Macko, and A. Napoli (2014). Biclustering meets triadic concept analysis. *Annals of Mathematics and Artificial Intelligence 70*(1-2), 55–79.

Kaytoue, M., S. O. Kuznetsov, and A. Napoli (2011). Biclustering numerical data in formal concept analysis. In *International Conference on Formal Concept Analysis*, pp. 135–150. Springer.

Kaytoue, M., S. O. Kuznetsov, A. Napoli, and S. Duplessis (2011). Mining Gene Expression Data with Pattern Structures in Formal Concept Analysis. *Information Science 181*(10), 1989–2001.

Madeira, S. C. and A. L. Oliveira (2004). Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) 1*(1), 24–45.

Tanay, A., R. Sharan, and R. Shamir (2002). Discovering statistically significant biclusters in gene expression data. *Bioinformatics 18*(suppl_1), S136–S144.

Analyse de Concepts Formels, distributivité et modèles de graphes médians pour la phylogénie

Alain Gély*, Miguel Couceiro*, Amedeo Napoli*

*LORIA (CNRS - Inria Nancy Grand Est - Université de Lorraine),
BP 239, 54506 Vandoeuvre-les-Nancy, France
alain.gely, miguel.couceiro, amedeo.napoli@loria.fr

Résumé. La phylogénie est l'étude des relations de parentés entre les êtres vivants. La classification phylogénétique consiste à classer les êtres vivants à partir de données de phylogénie. Traditionnellement, les modèles utilisés pour ce faire sont les arbres phylogénétiques. Ces arbres ne permettent cependant pas de capturer toute la complexité des phénomènes évolutifs. Du fait de cette complexité, plusieurs arbres peuvent convenir. Pour ne pas privilégier de solution particulière, l'utilisation de graphes médians permet d'encoder l'ensemble des arbres dans un graphe particulier, le graphe médian. Les graphes médians ont des liens étroits avec certains types de treillis, une autre structure souvent utilisée en classification. L'Analyse de Concepts Formels (FCA) a fait des treillis de concepts l'objet central d'étude pour des problèmes d'analyse de données. Dans cet article, nous montrons comment utiliser la FCA pour produire des graphes médians, et nous mettons en avant les verrous techniques à franchir.

1 Introduction

La phylogénie est l'étude des relations de parentés entre les êtres vivants. La classification phylogénétique consiste à classer les êtres vivants à partir de données de phylogénie (variations génétiques par exemple). Traditionnellement, les modèles utilisés pour de telles classifications sont les arbres phylogénétiques. Ces derniers ne permettent cependant pas de capturer toute la complexité des phénomènes évolutifs (mutations inverses, transfert horizontal de gènes). Du fait de cette complexité, plusieurs arbres phylogénétiques peuvent convenir pour les mêmes données initiales.

L'utilisation de graphes médians, introduits par Bandelt (Bandelt et Hedlíková (1983); Bandelt et al. (1999)), permet d'encoder la famille de tous les arbres parcimonieux (minimisant le nombre de changements nécessaires pour passer d'une espèce à l'autre) dans un graphe particulier, le graphe médian. Les graphes médians ont des liens étroits avec certains types de treillis, en particulier les treillis distributifs : tout treillis distributif est un graphe médian, et tout graphe médian peut être considéré comme un semi-treillis vérifiant la propriété de médiane (voir ci-après) sur chaque triplet d'éléments.

Les treillis distributifs sont étudiés dans de nombreux domaines, tant pour leur intérêt théorique que pratique, entre autre pour le lien fort entre ordres partiels et treillis distributifs (Birkhoff (1937)). Birkhoff leur consacre un chapitre dans son ouvrage de référence sur les treillis

(Birkhoff (1967)) et son résultat de représentation des treillis distributifs par des ordres partiels sera central dans les travaux présentés ici.

Les treillis distributifs ne sont qu'une classe particulière de treillis. Pour un treillis quelconque, la distributivité des deux opérations $\vee$ et $\wedge$ n'est pas vérifiée. Les treillis en général sont centraux en Analyse de Concept Formels (FCA) (Ganter et Wille (1999); Barbut et Monjardet) pour l'analyse de données. Uta Priss, dans une série de deux publications (Priss (2012, 2013)) étudie les liens entre FCA et graphes médians pour la phylogénie. Ces deux articles restent au niveau conceptuel et abordent peu les détails algorithmiques sous-jacents.

Dans nos travaux (Gély et al. (2018b,a)), nous nous sommes intéressés a cet aspect algorithmique et avons formalisé une approche. Nous faisons ici un point sur la manière d'obtenir un graphe médian en utilisant les outils de l'Analyse de Concepts Formels. La section 2 détaille les différents modèles possibles, la section 3 montre l'algorithme utilisé. Nous concluerons en section 4 en évoquant les travaux en cours et en donnant quelques perspectives.

2 Modèles

2.1 Graphes médians

Un graphe médian est un graphe $G = (V, E)$ tel que pour tout triplet de sommets $x, y, z \in V$, il existe un unique sommet t à l'intersection de tous les plus courts chemins entre chaque paire de sommets. Les arbres sont un cas particulier de graphe médian.

En phylogénie, le graphe de Buneman (Buneman (1971)), qui est le graphe représentant l'ensemble des arbres phylogénétiques parcimonieux (minimisant le nombre de mutations nécessaires pour passer d'un individu à l'autre) est un graphe médian. Les sommets du graphe de Buneman représentent d'une part les espèces à considérer pour la phylogénie, et d'autre part un certain nombre de sommets latents, ajoutés de façon à vérifier la propriété de médiane. Lorsque les espèces sont décrites par un ensemble de caractères (de type booléen "présent/absent", ou bien "muté/non muté"), il y a une arête entre deux espèces lorsqu'elles ne diffèrent que par un caractère.

Notons que si la phylogénie est parfaite (il n'y a pas eu de mutation inverse ou de transfert latéral), le graphe obtenu est alors naturellement un arbre. Si elle n'est pas parfaite, alors plusieurs arbres peuvent convenir. Chacun de ces arbres est un arbre couvrant du graphe médian obtenu.

2.2 Analyse de Concepts Formels

La classification phylogénétique peut souvent se ramener à utiliser des données binaires entre objets (les espèces) et variables (présence/absence d'une mutation). Ainsi, on peut définir un contexte formel $C = (O, A, I)$, avec O l'ensemble des objets (espèces), A l'ensemble des attributs (mutations) et I une relation binaire entre O et A, telle que pour $o \in O$, $a \in A$ $I(o, a)$ (noté oIa) se lit comme "l'objet o possède l'attribut a" (l'espèce o possède la mutation a).

Un treillis $\mathbf{T} = (T, \leq, \vee, \wedge)$ est un ensemble ordonné muni de deux opérateurs $\vee$ (resp. $\wedge$) correspondant à la borne supérieure (resp. inférieure) de deux éléments de T. Par définition, dans un treillis (contrairement à un ordre quelconque), les bornes supérieures et inférieures

existent toujours. On parlera de semi-treillis si l'on se restreint à l'existence d'une seule de ces deux bornes.

A partir du contexte $C = (O, A, I)$ et des connections de Galois rappelées ci-dessous (Def. 1), on peut définir un treillis $\mathcal{B}(C)$, treillis des concepts du contexte C. Les éléments de ce treillis sont les concepts, c'est à dire les ensembles (X, Y), $X \subseteq O$, $Y \subseteq A$ tels que $X' = Y$ et $Y' = X$. On appelle *extent* l'ensemble X et *intent* l'ensemble Y. En particulier, X et Y vérifient $X = X''$ et $Y = Y''$ et sont des ensembles fermés. La relation d'ordre entre concept est une relation d'inclusion entre les extensions des concepts. Pour plus de détails sur l'Analyse de Concepts Formels, le lecteur pourra se reporter à l'ouvrage de base Ganter et Wille (1999)

Définition 1 *Soit $(O, A, \leq)$ un contexte, on peut définir une connections de Galois entre O et A comme suit :*

— $' : 2^O \to 2^A$, $X' = \{a \mid \forall o \in O, oIa\}$
— $' : 2^A \to 2^O$, $Y' = \{o \mid \forall a \in A, oIa\}$

Un concept représente l'ensemble maximal des individus partageant un ensemble maximal d'attributs. L'ajout d'un nouvel attribut à l'intent (resp. d'un nouvel objet à l'extent) va séparer les objets (resp. attributs) en deux parties strictement non vides : les objets (resp. attributs) en relation avec ce nouvel attribut (resp. objets), et les autres.

Un graphe médian est isomorphe à un semi-treillis distributif particulier. Un treillis des concepts $\mathcal{B}(C)$ n'a pas de raison *a priori* d'être distributif. Il faut donc pouvoir transformer un treillis quelconque en un treillis distributif. Aussi, utiliser le formalisme FCA pour la production de graphe médian va nous amener à décrire plus en détail les treillis distributifs, ce qui est fait dans la section suivante.

2.3 Treillis distributifs

Par définition, un treillis distributif est un treillis pour lequel la loi de distributivité s'applique entre $\vee$ et $\wedge$, c'est à dire : $\forall x, y, z \in T : x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$. Birkhoff s'est énormément intéressé aux treillis distributifs dès les années 30 avec un article dont est issu un des résultats utilisé ici (Birkhoff (1933)). On retrouve aussi la plupart des résultats détaillés dans l'ouvrage Caspard et al. (2012)

Il découle de cette définition plusieurs caractérisations, dont l'une établit le lien entre treillis distributif et graphe médian : un treillis $(T, \leq, \vee, \wedge)$ est distributif ssi $\forall x, y, z$, $(x \wedge y) \vee (y \wedge z) \vee (z \wedge x) = (x \vee y) \wedge (y \vee z) \wedge (z \vee x)$

Or, on peut définir une opération de médiane sur un ensemble M comme une fonction :

$$m : M^3 \to M$$

vérifiant

$$m(a, a, b) = a \text{ et } m(m(a, b, c), d, c) = m(a, m(b, c, d), c)$$

Ainsi, $m(a, b, c) = (a \wedge b) \vee (b \wedge c) \vee (c \wedge a)$ définit une opération de médiane sur un treillis distributif et ce résultat est utilisé par Bandelt (Bandelt et al. (1999)) pour rapprocher les treillis distributifs des graphes médians.

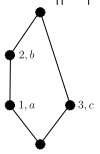
Une autre caractérisation des treillis distributifs est qu'ils ne contiennent ni M_3 ni N_5 comme sous-treillis (un sous-treillis T_1 est un sous-ordre stable pour les opérations $\vee$ et $\wedge$, c'est à dire que si $x, y \in T_1$ alors $x \vee y \in T_1$ et $x \wedge y \in T_1$). Sur la figure 1, on trouve le treillis non distributif N_5 et un treillis distributif. Notons que si N_5 est un sous-ordre du treillis de droite, il n'en est pas un sous-treillis. Pour des raisons de place, M_3, composé d'une antichaîne de 3 éléments, d'un plus petit élément $\perp$ et un plus grand élément $\top$, n'est pas représenté.

Théorème de représentation de Birkhoff. Notre algorithme s'appuie sur un des résultats principaux de Birkhoff pour les treillis distributifs, le théorème de représentation. Ce théorème établit que les treillis distributifs sont en bijection avec les treillis des idéaux d'un ensemble ordonné. De plus, étant donné un treillis distributif, on peut facilement retrouver l'ensemble ordonné dont il est le treillis des idéaux. Il s'agit de l'ensemble ordonné induit par les éléments $\vee$-irréductibles du treillis.

Un *idéal* X est un ensemble ordonné tel que si $x \in X$ et $y < x$, alors $y \in X$. Sur la figure 1, l'ensemble ordonné à droite de la figure correspond par exemple aux idéaux suivants $\{\emptyset, \{1\}, \{3\}, \{1, 2\}, \{1, 3\}, \{1, 2, 3\}\}$. Les idéaux de cet ordre, ordonnés par inclusion, forment un treillis isomorphe au treillis représenté à droite. Ce treillis est un treillis distributif.

Un élément $\vee$-irréductible d'un treillis est un élément qui n'est pas borne supérieure de deux éléments autre que lui même. Par exemple, l'élément d'étiquette d (Fig. 1 Droite) n'est pas $\vee$-irréductible puisque $d = 1 \vee 3$. La famille des éléments $\vee$-irréductible de ce treillis est $\{1, (2, b), (3, c)\}$. Ordonné par inclusion, l'ensemble ordonné obtenu est isomorphe à l'ensemble ordonné de droite.

(O, A, I)	a	b	c
1	×	×	
2		×	
3			×



$(O, O, \not\geq)$	a	b	c
1		×	×
2			×
3	×	×	

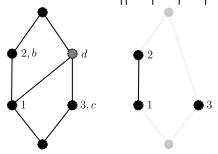


FIG. 1 – *Gauche. Le treillis non distributif N_5 et son contexte. Droite. Un treillis distributif, son contexte et l'ordre induit par les éléments $\vee$-irréductibles. Les deux treillis présentés partagent le même ordre induit par les éléments $\vee$-irréductibles. Le treillis de gauche (N_5) peut se plonger (plongement d'ordre) dans le treillis de droite. Remarquons que les sommets du treillis ont parfois deux étiquettes, selon que l'on considère l'ensemble d'objets ou d'attributs*

3 FCA et construction de graphe médian

Il est possible d'utiliser le théorème de représentation de Birkhoff pour produire un treillis distributif T_d à partir d'un treillis quelconque T tel que T puisse se plonger (plongement

d'ordre) dans T_d. Un treillis distributif étant un graphe médian, c'est donc un moyen de construire un graphe médian en utilisant le formalisme FCA.

On notera que l'entrée des algorithmes est rarement un treillis des concepts, mais plutôt un contexte. Celui-ci contient forcément les éléments $\vee$-irréductibles et il est connu (Ganter et Wille (1999)) qu'à partir du contexte d'un treillis, on peut calculer le contexte du treillis distributif ayant le même ordre induit par les éléments $\vee$-irréductibles. Ce contexte est $C = (O, O, \not\geq)$.

Construction d'un semi-treillis distributif à partir d'un contexte. Si tout treillis distributif est un graphe médian, les graphes médians ne sont pas tous des treillis. Dans ce cas, ils sont isomorphes à un semi-treillis distributif particulier (on étend la notion de distributivité aux semi-treillis en considérant qu'un $\vee$-semi-treillis est distributif si, pour chaque élément minimal o, les éléments qui lui sont supérieurs – le filtre de o – forment un treillis).

Depuis un contexte, on peut construire un semi-treillis en considérant le treillis des concepts privé de son élément minimum. Une méthode pour obtenir un semi-treillis distributif est alors d'appliquer le théorème de représentation de Birkhoff sur chacun des filtres des éléments minimaux. C'est la méthode qui est présentée dans l'algorithme 1. Cette méthode nécessite une boucle externe pour vérifier que les modifications d'un filtre n'ont pas remis en question la distributivité d'un autre filtre. Il est en effet possible que des éléments soient communs à plusieurs filtres.

Algorithme 1 : Construction du contexte du $\vee$-semi-treillis distributif.

Données : Un contexte $C = (O, A, I)$
Résultat : Le contexte $C_d = (O, A_d, I_d)$ d'un semi-treillis distributif
pour chaque $o \in O$, *minimal* **faire**
  $(P_o, \leq) \leftarrow \emptyset$
répéter
  stabilité $\leftarrow$ vrai ;
  pour chaque $o \in O$, *minimal* **faire**
    calculer P_o l'ensemble ordonné des éléments $\vee$-irreductibles supérieurs à o
    Produire le contexte $C_o = (P_o, P_o, \not\geq)$
    si P_o *modifié depuis la dernière itération* **alors**
      stabilité $\leftarrow$ faux ;
  Fusionner les différents contextes $C_o = (P_o, P_o, \not\geq)$
jusqu'à *stabilité*

4 Conclusion

Il est possible d'utiliser les outils de l'analyse de concepts formels pour produire un graphe médian. Dans les grandes lignes, cela revient à plonger un treillis dans un treillis distributif en utilisant le théorème de représentation de Birkhoff. Cependant, dans le cas où la sortie recherchée est un semi-treillis, il faut considérer les filtres de chaque élément minimal et nous avons

proposé une méthode en ce sens. Parce que les filtres ne sont pas forcément disjoints, l'approche présentée peut produire une solution non minimale (voir Gély et al. (2018a)). D'autre part, des travaux en cours montrent que plusieurs solutions minimales non isomorphes peuvent exister. Il reste maintenant à caractériser une solution minimale canonique et à obtenir un algorithme produisant cette solution.

Références

Bandelt, H.-J., P. Forster, et A. Röhl (1999). Median-joining networks for inferring intraspecific phylogenies. *Molecular biology and evolution 16*(1), 37–48.

Bandelt, H.-J. et J. Hedlíková (1983). Median algebras. *Discrete mathematics 45*(1), 1–30.

Barbut, M. et B. Monjardet. Ordre et classification, paris, hachette, 1970. *Zbl0267 6001*.

Birkhoff, G. (1933). On the combination of subalgebras. In *Mathematical Proceedings of the Cambridge Philosophical Society*, Volume 29, pp. 441–464. Cambridge University Press.

Birkhoff, G. (1937). Rings of sets. *Duke Math. J. 3*(3), 443–454.

Birkhoff, G. (1967). *Lattice Theory* (3rd ed.). Providence : American Mathematical Society.

Buneman, P. (1971). The recovery of trees from measures of dissimilarity. *Mathematics in the archaeological and historical sciences*.

Caspard, N., B. Leclerc, et B. Monjardet (2012). *Finite ordered sets : concepts, results and uses*. Number 144. Cambridge University Press.

Ganter, B. et R. Wille (1999). *Formal Concept Analysis : Mathematical Foundations*. Springer.

Gély, A., M. Couceiro, Y. Namir, et A. Napoli (2018b). Contribution à l'étude de la distributivité d'un treillis de concepts. In *Extraction et Gestion des Connaissances, EGC 2018, Paris, France, January 23-26, 2018*, pp. 107–118.

Gély, A., M. Couceiro, et A. Napoli (2018a). Steps towards achieving distributivity in formal concept analysis. In *Proceedings of the Fourteenth International Conference on Concept Lattices and Their Applications, CLA 2018, Olomouc, Czech Republic, June 12-14, 2018.*, pp. 105–116.

Priss, U. (2012). Concept lattices and median networks. In *CLA*, pp. 351–354.

Priss, U. (2013). Representing median networks with concept lattices. In *ICCS*, pp. 311–321. Springer.

Summary

Philogenetic classification uses phylogeny data to classify species. The more traditionnal models are phylogenetic trees. Nevertheless, trees miss some complexity of evolution, and so, several trees should be used. Median graphs permit to encode all these trees in a unique structure. Median graphs have links with some kind of lattices, another structure used in data analysis. Concept lattices are the central object of Formal Concept Analysis (FCA), a framework for data analysis. In this article, we show how to use FCA to produce median graphs and we enlight some technical difficulties to be tackled.