Actes des 30es rencontres de la Société Francophone de Classification

Pascal Préa

AfIA

Association française
pour l'Intelligence Artificielle

SFC

30es rencontres de la Société Francophone de Classification

PFIA 2025

Table des matières

Remerciements

30es rencontres de la Société Francophone de Classification

Les rencontres de la SFC ont été soutenues par le Laboratoire d'Informatique et des Systèmes (LIS, UMR 7020).

Pascal Préa

Comité de programme

Présidence

— Pascal Préa, École Centrale Méditerranée.

Membres

— Rafik Abdesselam, Université Lumière Lyon ;
— Séverine Affeldt, Université de Paris ;
— Alexandre Bazin, LIRMM, Montpellier ;
— Patrice Bertrand, Université Paris Dauphine ;
— Paula Brito, Université de Porto, Portugal ;
— François Brucker, École Centrale Méditerranée ;
— Véronique Cariou, ONIRIS Nantes ;
— Christian Derquenne, EDF R & D ;
— Dominique Desbois, INRAE-Paris-Saclay ;
— Jean Diatta, Université de La Réunion ;
— Nadia Ghazalli, Université du Québec à Trois-Rivières, Canada ;
— Pascale Kuntz, Université de Nantes ;
— Lazhar Labiod, Université Paris Descartes ;
— Mustapha Lebbah, Université Paris 13 ;
— Ahmed Moussa, ENSA Tanger, Maroc ;
— Mohamed Nadif, Université Paris Descartes ;
— Amedeo Napoli, LORIA, Nancy ;
— Ndèye Niang, CNAM Paris ;
— Allou Samé, Université Gustave Eiffel ;
— Rosanna Verde, Université della Campania, Caserta, Italie.

Conférence Invitée

On decision making in cluster analysis

Christian Martin Hennig[1]

[1] Universita di Bologna

christian.hennig@unibo.it

Résumé

There are many different approaches to cluster analysis, and when applied to the same data, different methods will pften produce quite different clusterings. Data analysts do not only have to choose a clustering method, also pre- and post-processing decisions need to be made, such as selection and transformation of features and the number of clusters.
Making all the required decisions is very difficult. As there is no unique definition of the clustering problem, neither is there a unique or "optimal" way to measure the quality of a clustering, and the data alone do not hold all the information required to make these decisions. This is a big challenge for automatising cluster analysis for machine learning in particular.
I will discuss some of the required decisions and quality criteria, illustrating problems with automated decision making, and how background knowledge and techniques such as data visualisation can help.

Prix Simon Régnier

Vers un Clustering plus équilibré : Augmentation de Données par GMMs et LLMs

Noor Khalal[1], Abdallah Alaa-Eddine Djamai[2], Imed Keraghel[1,2], Mohamed Nadif[1]
[1] Centre Borelli UMR 9010, Université Paris Cité
[2] Kernix Software

Résumé

En NLP, la gestion des thématiques sous-représentées est un défi, notamment en apprentissage non supervisé où le clustering peine à capturer les sujets minoritaires. Pour y remédier, nous proposons une méthode d'augmentation des données combinant les modèles de mélange gaussien (GMMs) et les grands modèles de langage (LLMs). Les GMMs identifient les clusters sous-représentés, tandis que les LLMs génèrent des documents synthétiques pour les enrichir. Nos expériences sur divers ensembles de données déséquilibrés montrent que cette approche préserve les performances du clustering et améliore l'interprétabilité des clusters, offrant une solution robuste et évolutive en NLP non supervisé.

Mots-clés

Augmentation des données, Grands Modèles de Langage, Modèles de Mélange Gaussien, Apprentissage non supervisé, Clustering.

Abstract

In NLP, handling underrepresented topics is challenging, particularly in unsupervised tasks where clustering may fail to capture minority topics effectively. To address this, we propose an unsupervised data augmentation method that combines Gaussian Mixture Models (GMMs) and Large Language Models (LLMs). GMMs identify underrepresented clusters, while LLMs generate synthetic documents to enrich them. Experiments on various imbalanced text datasets show that our approach maintains clustering performance and often improves interpretability, providing a robust and scalable solution for enhancing data representation in unsupervised NLP.

Keywords

Data Augmentation, Large Language Models, Gaussian Mixture Models, Unsupervised Learning, Clustering.

1 Introduction

Dans les tâches de NLP non supervisées, la qualité de la représentation des données est cruciale pour un clustering efficace. Cependant, les ensembles de données réels contiennent souvent des concepts sous-représentés, ce qui entraîne des groupes fortement déséquilibrés, difficiles à traiter pour la plupart des algorithmes de clustering. Ces clusters sous-représentés peuvent contenir des informations significatives qui restent mal capturées par les algorithmes traditionnels, conduisant ainsi à des clusterings peu pertinents ou difficiles à interpréter.

Les techniques classiques d'augmentation de données, telles que le remplacement de synonymes et l'insertion aléatoire [23], augmentent la diversité des échantillons, mais elles s'appliquent de manière uniforme sans corriger les déséquilibres des données. De plus, de nombreuses approches existantes reposent sur des données annotées pour réaliser une augmentation ciblée [2, 10, 8], ce qui les rend inadaptées aux scénarios d'apprentissage non supervisé où les étiquettes de classe ne sont pas disponibles.

Les modèles génératifs, en particulier les grands modèles de langage (LLMs), ont récemment suscité un intérêt croissant en raison de leur capacité à produire du texte synthétique de haute qualité en capturant des relations sémantiques complexes au sein des données. Cependant, la plupart des travaux existants utilisent les LLMs pour une augmentation uniforme des ensembles de données [25], ce qui ne permet pas de résoudre le problème du déséquilibre.

Pour surmonter ces limitations, nous proposons une méthode combinant les modèles de mélange gaussien (GMMs) [1] et les LLMs pour une augmentation ciblée des données. Les GMMs analysent la distribution des embeddings et identifient les zones sous-représentées, où les LLMs génèrent ensuite des documents synthétiques pour améliorer leur représentation. La Figure 1 illustre ce processus sur le jeu de données Tweet Emotion : trois tweets sur l'optimisme sont fournis à un LLM, qui en génère un nouveau reflétant la même idée. Ce procédé enrichit la diversité du corpus et améliore sa représentation.

2 Contexte et travaux connexes

L'augmentation des données est une technique courante en NLP pour diversifier et accroître la taille des ensembles d'entraînement en générant de nouveaux échantillons [15, 2]. En classification de texte, les méthodes traditionnelles comme le remplacement de synonymes ou l'insertion aléatoire—regroupées sous Easy Data Augmentation (EDA) [23]—accroissent la variabilité des données mais n'adressent pas le déséquilibre des classes, étant appliquées uniformément. Pour corriger ce déséquilibre, des approches comme le rééchantillonnage et l'apprentissage sensible aux coûts ont été proposées [3, 10]. Cependant,

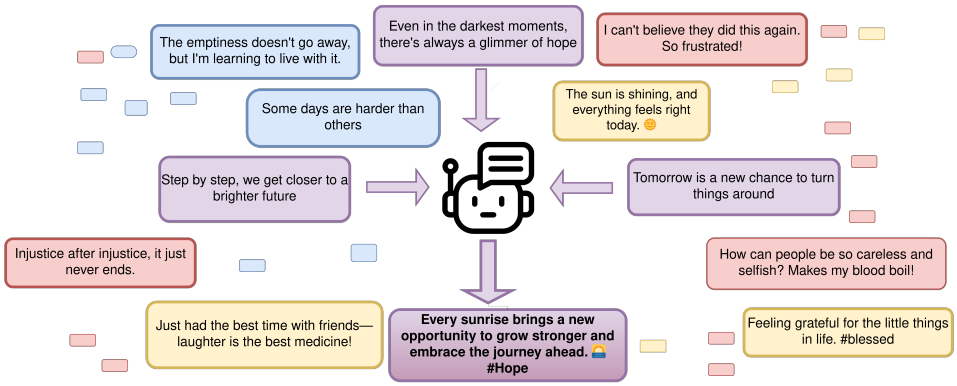

FIGURE 1 – Exemple issu du jeu de données Tweet Emotion : quatre clusters d'émotions (joie en jaune, tristesse en bleu, colère en rouge, optimisme en violet) où un LLM génère un nouveau tweet sur l'optimisme à partir de trois exemples représentatifs.

elles nécessitent des étiquettes de classe, limitant leur applicabilité en apprentissage non supervisé. Les modèles génératifs, tels que les GANs [6, 16] et les LLMs [12, 22], ont également été explorés pour l'augmentation des données, mais leur utilisation reste souvent uniforme et non ciblée [4]. Malgré ces avancées, peu de travaux combinent le clustering et les modèles génératifs pour traiter les classes sous-représentées en apprentissage non supervisé. Nos travaux comblent cette lacune en s'appuyant sur les GMMs pour détecter les clusters sous-représentés et sur les LLMs pour générer des documents synthétiques, sans dépendre de données annotées.

3 Contribution

Dans cette section, nous présentons notre approche d'augmentation des données textuelles en combinant des techniques d'encodage et des modèles gaussiens paramétriques [1]. Notre méthodologie repose sur plusieurs étapes clés : 1) représentation des documents, 2) clustering avec l'algorithme d'Expectation-Maximization (EM) [5], 3) génération de points de données synthétiques, et 4) utilisation d'un LLM pour l'augmentation des données. Cette approche est illustrée dans la Figure 2 et L'Algorithme 1.

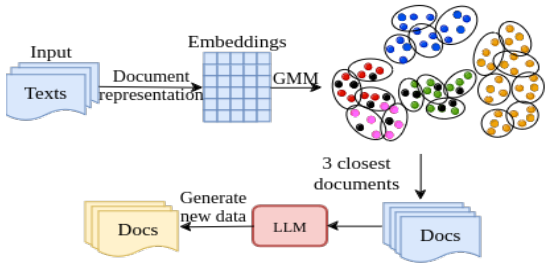

FIGURE 2 – Flux de travail pour générer des documents synthétiques. 1) Création d'embeddings, 2) clustering des embeddings via l'algorithme EM, 3) Génération de points synthétiques dans les clusters avec les meilleurs ratios volume/proportion, 4) Utilisation de ces points pour générer de nouvelles données via un LLM.

3.1 Représentation des documents

Pour le clustering, nous représentons les documents à l'aide d'un modèle d'encodage basé sur les Transformers, générant des embeddings qui capturent une information sémantique approfondie. Ces embeddings facilitent l'identification de connexions subtiles entre documents non étiquetés, améliorant ainsi la qualité du clustering [11]. Nous appliquons également UMAP [13] pour réduire la dimension des embeddings. Cette technique non linéaire préserve la structure des données tout en diminuant la complexité computationnelle, optimisant ainsi l'efficacité des algorithmes de clustering.

3.2 Clustering ciblé avec les GMMs

clustering avec les modèles de mélange gaussien (GMM). Dans un GMM fini, les données $\mathbf{x}_1, \ldots, \mathbf{x}_n$ sont supposées être un échantillon de n instances indépendantes d'une variable aléatoire $\mathbf{X}$ dans $\mathbb{R}^d$, où d est la dimension de l'espace. La densité des données est exprimée comme suit :

$$f(\mathbf{x}_i; \Theta) = \sum_{k=1}^{g} \pi_k \varphi_k(\mathbf{x}_i | \mu_k, \Sigma_k), \quad \forall i \in \{1, \ldots, n\} \quad (1)$$

où $\Theta = (\pi_1, \ldots, \pi_g, \mu_1, \ldots, \mu_g, \Sigma_1, \ldots, \Sigma_g)$, $\varphi_k(\mathbf{x}_i | \mu_k, \Sigma_k)$ est la densité de la k-ième composante pour l'observation $\mathbf{x}_i$ avec les paramètres (μ_k, Σ_k). Les π_k sont les poids de mélange (avec $\pi_k > 0, \sum_k \pi_k = 1$), et g est le nombre de composantes du mélange. Chaque cluster est ainsi représenté par une distribution gaussienne, dont les propriétés géométriques (volume, forme, orientation) sont définies par la matrice de covariance Σ_k [1]. L'estimation des paramètres Θ se fait en maximisant la log-vraisemblance :

$$L(\mathbf{X}; \Theta) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{g} \pi_k \varphi_k(\mathbf{x}_i | \mu_k, \Sigma_k) \right).$$

L'algorithme EM est utilisé pour maximiser cette fonction de manière itérative.

3.3 Augmentation ciblée des données

Génération de points de données synthétiques. Dans chaque cluster sous-représenté, de nouveaux points de données sont générés selon la distribution gaussienne associée. Le nombre de points générés suit une distribution multinomiale basée sur les poids π_k :

$$P(n_1, n_2, \ldots, n_g) = \frac{n_{\text{samp}}!}{n_1! n_2! \cdots n_g!} \pi_1^{n_1} \pi_2^{n_2} \cdots \pi_g^{n_g}.$$

Ensuite, chaque composante k génère des échantillons selon une loi normale multivariée de paramètres μ_k et Σ_k. Cela garantit que les points synthétiques suivent la structure des clusters d'origine.

Génération de documents textuels. Pour chaque point généré, nous identifions ses trois plus proches voisins dans les données originales, puis un LLM génère un nouveau document en combinant leurs contenus, assurant ainsi cohérence et pertinence sémantique.

Algorithm 1 clustering et génération de documents

Entrée : $D = \{d_1, \ldots, d_n\}$, g (nombre de composantes), k^* (nombre de clusters à augmenter), $\mathcal{M}$ (modèle d'encodage), $\mathcal{M}^*$ (modèle instruction-tuné)

$X_{\text{emb}} \leftarrow \mathcal{M}(D)$
$X_R \leftarrow \text{UMAP}(X_{\text{emb}})$
$\{C_1, \ldots, C_g\} \leftarrow \text{EM}(X_R)$
for $k \leftarrow 1, g$ **do**
    $S_k \leftarrow \frac{V_k}{\pi_k}$
end for
Trier C_k par S_k et retenir k^* clusters
Générer les nouveaux documents avec le LLM

4 Expériences

Nous évaluons l'efficacité de notre augmentation ciblée via le clustering, en comparant les performances avant et après augmentation avec des algorithmes comme KMeans.

4.1 Ensembles de Données

Les jeux de données utilisés dans nos expériences sont décrits dans la Table 1.

Données	Nb. Classes	Nb. Docs	Balance	#Tokens
Arxiv	12	7000	6.8×10^{-3}	10
Biorxiv	26	53787	4.1×10^{-4}	13
Medrxiv	51	17647	4.9×10^{-4}	16
Reddit	15	5114	1.2×10^{-3}	11
Tweet Emotion	4	3257	2.1×10^{-1}	16

TABLE 1 – Caractéristiques des jeux de données utilisés. **Balance** représente le ratio entre les classes minoritaires et majoritaires et **#Tokens** indique le nombre moyen de tokens.

4.2 Configuration Expérimentale

Nous utilisons le modèle *NoInstruct-Small-v0* [1], qui génère des embeddings de dimension 384, sélectionné pour ses bonnes performances sur MTEB [2]. Pour réduire la complexité computationnelle, nous appliquons UMAP pour réduire la dimension des embeddings à 10.

Nous ajustons un GMM aux embeddings réduits X_R, avec un nombre de composantes g fixé à $\lceil nDocs^{1/3} \rceil$, suivant [24]. L'utilisation de matrices de covariance complètes permet à chaque cluster d'avoir sa propre variance.

Processus d'Augmentation Après obtention des clusters GMM, nous calculons le ratio volume/proportion $S_k = \frac{V_k}{\pi_k}$ pour chaque cluster, où V_k est le volume et π_k son poids. Les clusters sont triés par ordre décroissant de S_k afin d'identifier ceux couvrant un grand volume mais contenant peu de points de données. Nous sélectionnons les k^* clusters ayant les plus hauts S_k pour l'augmentation. Pour chaque cluster sélectionné, nous générons des embeddings synthétiques $(DA)_k$. Chaque embedding est associé aux trois documents les plus proches dans X_R, qui sont fournis au modèle *Mistral-7B-Instruct-v0.2* [9] afin de générer de nouveaux documents textuels. Ces documents $(DA)^*$ sont ensuite encodés et ajoutés à X_{emb}, formant ainsi l'ensemble de données augmenté.

5 Résultats et discussion

Nous évaluons l'impact de notre augmentation des données selon trois aspects. D'abord, nous analysons la distribution des clusters pour observer son influence sur l'équilibre et la répartition des documents dans l'espace latent. Ensuite, nous comparons les mots-clés afin d'évaluer l'amélioration de la couverture thématique et de l'interprétabilité des clusters. Enfin, nous mesurons la performance du clustering à l'aide des métriques NMI [21] et ARI [20].

5.0.1 Comparaison des mots-clés

Pour évaluer l'interprétabilité des classes, nous analysons la distribution des mots-clés dans les clusters KMeans avant et après augmentation des données. Les mots-clés sont extraits à l'aide de KeyBERT. Le tableau 2 présente les 12 mots-clés les plus fréquents pour chaque cluster dans les données augmentées et non augmentées sur plusieurs jeux de données : *Tweet_Emotions*, *Reddit* et les corpus scientifiques (*Arxiv*, *Biorxiv* et *Medrxiv*).

L'analyse révèle l'impact de l'augmentation sur les clusters sous-représentés et sur-représentés selon les domaines. Dans *Tweet_Emotions*, le cluster *Optimism* s'est enrichi de termes positifs comme *happy* et *smile*, tandis que des mots négatifs tels que *nervous* et *panic* ont disparu, affinant ainsi son orientation. Pour le cluster *Anger*, de nouveaux termes comme *game* et *racism* reflètent une expression plus large et intense de la colère, bien que la structure globale du cluster reste stable, comme attendu pour une catégorie sur-représentée.

Dans *Reddit*, le cluster sous-représenté *Skincare* est devenu plus précis avec des termes comme *cream* et *toner*, tandis que le cluster sur-représenté *Dogecoin* a peu évolué, conservant son focus sur *doge* avec une légère augmentation de termes comme *bought* et *currency*. L'impact est plus variable dans les corpus scientifiques. Dans *Arxiv*, le cluster *Economics* s'est élargi avec des termes comme *regression* et *sparsity*, reflétant un champ méthodologique plus large,

1. https://huggingface.co/instructor/NoInstruct-Small-v0
2. https://huggingface.co/spaces/mteb/leaderboard

TABLE 2 – Comparaison des mots-clés et de leurs fréquences dans les clusters sur-représentés (+) et sous-représentés (-), avant et après l'augmentation des données. Le nombre en bas à droite indique la fréquence du mot. Les mots en gras sont nouveaux, tandis que les autres sont communs aux deux ensembles et soulignés s'ils ont la fréquence la plus élevée.

Cluster	Non Augmenté	Augmenté	Observations
Optimism (-)	depression$_{58}$, life$_{38}$, day$_{26}$, feel$_{24}$, nervous$_{22}$, lost$_{21}$, panic$_{16}$, optimism$_{16}$, despair$_{15}$, gloomy$_{15}$, love$_{14}$, shy$_{13}$	depression$_{58}$, **music**$_{40}$, life$_{40}$, day$_{36}$, feel$_{25}$, lost$_{22}$, **happy**$_{22}$, love$_{21}$, optimism$_{17}$, **sober**$_{17}$, **smile**$_{16}$, **birthday**$_{15}$	L'augmentation a renforcé les thèmes optimistes et positifs, apportant de la diversité tout en réduisant l'accent sur les termes négatifs ou neutres.
Anger (+)	angry$_{41}$, bully$_{35}$, outrage$_{30}$, terror$_{28}$, people$_{26}$, rage$_{25}$, dont$_{24}$, offended$_{23}$, revenge$_{17}$, irritate$_{17}$, insult$_{17}$, hate$_{13}$	angry$_{41}$, bully$_{35}$, terror$_{29}$, people$_{28}$, outrage$_{27}$, rage$_{25}$, insult$_{17}$, hate$_{13}$, **game**$_{10}$, **play**$_{10}$, revenge$_{9}$, **racism**$_{9}$	Les termes sont restés globalement stables, mais quelques nouveaux mots ont été ajoutés, exprimant une colère plus intense (ex. : *game*, *racism*).
Skincare (-)	sellus$_{6}$, farmacy$_{2}$, moisturizer$_{1}$, brand$_{1}$, moisture$_{1}$, birthday$_{1}$, glossier$_{1}$, balm$_{1}$, rituals$_{1}$, babor$_{1}$, small$_{1}$, look$_{1}$,	sellus$_{8}$ **skincare**$_{6}$, sale$_{5}$, glossier$_{4}$, **cream**$_{4}$, **toner**$_{4}$, balm$_{3}$, **treatments**$_{3}$, **dark**$_{3}$, **spot**$_{3}$, **sun**$_{3}$, farmacy$_{2}$, moisturizer$_{2}$,	L'augmentation a introduit des termes spécifiques aux soins de la peau (ex. : *skincare*, *cream*, *toner*) et a légèrement orienté le focus vers les ventes (*sale*).
DogeCoin (+)	doge$_{250}$, bought$_{17}$, currency$_{11}$, market$_{7}$, coin$_{7}$, community$_{7}$, dollar$_{6}$	doge$_{254}$, bought$_{27}$, currency$_{17}$, dollar$_{11}$, coin$_{11}$, community$_{11}$, price$_{10}$, market$_{8}$	Les termes sont restés cohérents, avec de légères augmentations de fréquence. L'augmentation a légèrement mis l'accent sur les aspects financiers, sans modifier le focus principal sur *doge*.
Economics (-)	sustainable$_{5}$, inequality$_{4}$, frequency$_{4}$, likelihood$_{3}$, dimensional$_{2}$, regression$_{2}$ pandemic$_{1}$, governments$_{1}$ evolutionary$_{1}$, strategy$_{1}$, approach$_{1}$, macroecon$_{1}$	**model**$_{7}$, regression$_{5}$, **series**$_{4}$, dimensional$_{4}$, **learning**$_{3}$, pandemic$_{3}$, **sparsity**$_{2}$, sustainable$_{2}$, approach$_{2}$, likelihood$_{1}$	L'augmentation a ajouté des termes liés aux méthodes économiques (ex. : *series*, *sparsity*) et renforcé l'accent sur les applications statistiques.
CS (+)	learning$_{47}$, neural$_{21}$, networks$_{20}$, classification$_{17}$, adversarial$_{15}$, deep$_{15}$, models$_{14}$, detection$_{12}$, recognition$_{11}$	learning$_{74}$, neural$_{43}$, detection$_{35}$, networks$_{31}$, deep$_{31}$, classification$_{30}$, recognition$_{27}$, adversarial$_{22}$, models$_{21}$,	Les termes restent stables, avec un accent renforcé sur des notions clés en informatique comme *learning*, *neural* et *detection*.
A B&C (-)	drosophila$_{14}$, olfactory$_{10}$, neurons$_{6}$, circadian$_{4}$, cortex$_{4}$, pathway$_{3}$, learning$_{3}$, dopaminergic$_{3}$, model$_{3}$, light$_{2}$, endocrine$_{2}$	drosophila$_{257}$, olfactory$_{176}$, circadian$_{166}$, **sleep**$_{66}$, **clock**$_{62}$, neurons$_{54}$, **taste**$_{47}$, light$_{30}$, **sensory**$_{26}$, **rhythms**$_{24}$, **odorant**$_{24}$	Les nouveaux termes comme *sleep*, *clock* et *taste* élargissent le contexte comportemental.
Neuroscience(+)	neural$_{69}$, cortex$_{58}$, learning$_{39}$, visual$_{38}$, memory$_{30}$, auditory$_{25}$, temporal$_{19}$, cortical$_{19}$, speech$_{17}$, spatial$_{16}$, dynamics$_{15}$, prefrontal$_{15}$	neural$_{546}$, cortex$_{407}$, visual$_{318}$, learning$_{280}$, **brain**$_{251}$, memory$_{244}$, cortical$_{195}$, auditory$_{168}$, **attention**$_{132}$, **perception**$_{121}$, dynamics$_{119}$, speech$_{117}$	L'accent sur *neural* et *cortex* s'intensifie, tandis que *brain* et *perception* élargissent le focus aux processus cognitifs et sensoriels.
Neurology (-)	alzheimers$_{35}$, cognitive$_{14}$, disease$_{12}$, dementia$_{10}$, brain$_{10}$, genetic$_{5}$, impairment$_{5}$, biomarkers$_{4}$, diagnosis$_{4}$, amyotrophic$_{4}$, trials$_{4}$, risk$_{4}$	alzheimers$_{156}$, **parkinsons**$_{115}$, cognitive$_{115}$, disease$_{107}$, brain$_{97}$, dementia$_{85}$, impairment$_{45}$, **epilepsy**$_{44}$, genetic$_{30}$, **eeg**$_{29}$, **stroke**$_{28}$, **cognition**$_{25}$	Les termes *parkinsons* et *stroke* élargissent le champ aux maladies neurologiques, tandis que *alzheimers* et *dementia* gagnent en importance, renforçant le focus sur les maladies dégénératives et la fonction cérébrale.
Epidemiology(+)	sarscov2$_{89}$, covid19$_{28}$, antigen$_{22}$, testing$_{19}$, diagnostic$_{16}$, rtpcr$_{15}$, saliva$_{14}$, rna$_{13}$, detection$_{10}$, test$_{9}$, nasopharyngeal$_{9}$, viral$_{8}$	sarscov2$_{329}$, antigen$_{125}$, covid19$_{104}$, detection$_{70}$, testing$_{70}$, saliva$_{67}$, diagnostic$_{54}$, rtpcr$_{44}$, test$_{39}$, rna$_{32}$, **tests**$_{31}$, **screening**$_{30}$	L'accent sur le diagnostic et les tests se renforce avec l'augmentation de termes comme *sarscov2*, *antigen* et *covid19*. Les ajouts de *tests* et *screening* élargissent les approches de détection épidémiologique.

alors que le cluster *Computer Science* est resté stable avec quelques ajouts mineurs. Dans *Biorxiv*, le cluster *Animal Behavior and Cognition* (*A B&C*) a introduit des termes sensoriels comme *sleep* et *taste*, suggérant un intérêt accru pour les mécanismes sensoriels. Le cluster *Neuroscience* a vu apparaître de nouveaux termes comme *brain*, tandis que des mots-clés centraux tels que *cortex* ont connu une augmentation significative (de 69 à 546 occurrences), reflétant l'augmentation proportionnelle des données générées. Dans *Medrxiv*, le cluster *Neurology* a mis en avant des termes

TABLE 3 – Résultats des performances de clustering (moyenne ± écart-type). Les résultats sont reportés pour les ensembles de données non augmentés (N.A) et augmentés (A).

Algo.	Type	Métrique	Tweet_Emo.	Reddit	Arxiv	Medrxiv	Biorxiv
KMeans	N.A	NMI	20.32 ± 2.28	54.74 ± 1.73	44.22 ± 0.91	30.07 ± 0.21	**34.03 ± 0.17**
		ARI	22.79 ± 6.23	25.69 ± 2.52	32.09 ± 2.26	6.92 ± 0.35	20.09 ± 1.79
	A	NMI	**22.10 ± 3.49**	**55.08 ± 0.58**	**44.28 ± 0.57**	**34.62 ± 0.21**	30.39 ± 0.21
		ARI	**22.81 ± 5.83**	**29.41 ± 1.78**	**34.10 ± 1.64**	**7.22 ± 0.24**	**20.59 ± 1.49**
SKmeans	N.A	NMI	20.85 ± 2.80	**55.91 ± 0.83**	43.59 ± 0.48	30.03 ± 0.30	**34.12 ± 0.16**
		ARI	19.39 ± 6.72	29.88 ± 1.83	32.75 ± 1.49	7.17 ± 0.44	22.24 ± 2.25
	A	NMI	**22.42 ± 5.04**	55.46 ± 1.21	**44.42 ± 0.72**	**30.44 ± 0.29**	33.86 ± 0.29
		ARI	**23.67 ± 9.74**	**31.53 ± 2.78**	**33.58 ± 1.60**	**7.65 ± 0.34**	**23.36 ± 3.01**

comme *parkinsons* et *stroke*, renforçant la thématique des maladies neurologiques, tandis que le cluster *Epidemiology* a introduit *sarscov2* et *testing*, soulignant un accent mis sur le diagnostic et la détection épidémiologique.
Globalement, ces résultats montrent que l'augmentation a efficacement enrichi les clusters sous-représentés avec des termes spécifiques et variés, améliorant leur interprétabilité. Les clusters sur-représentés, comme attendu, ont montré peu de changements structurels.

5.0.2 Distribution des Clusters

La Figure 3 illustre la distribution des documents dans les clusters pour deux ensembles de données : Reddit et Arxiv. Ces ensembles, avec un nombre modéré de clusters (15 et 12), permettent une visualisation claire des schémas de répartition. Les graphiques en barres comparent trois scénarios : les étiquettes de classe originales (zigzags bleus), le regroupement KMeans sur les données non augmentées (briques rouges) et après augmentation (points roses).

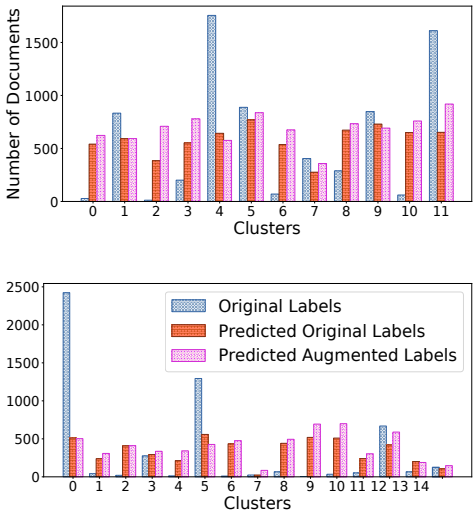

FIGURE 3 – Répartition des documents dans les clusters pour les ensembles de données Arxiv (en haut) et Reddit (en bas).

Dans les étiquettes initiales (zigzags bleus), le déséquilibre des ensembles de données est évident, avec certaines classes surreprésentées et d'autres sous-représentées. Le KMeans appliqué aux données d'origine (briques rouges) répartit uniformément les documents, réduisant la dominance des classes surreprésentées sans pour autant refléter fidèlement le déséquilibre initial.
Après l'augmentation des données (points roses), KMeans continue d'égaliser la taille des clusters. Cependant, une tendance importante émerge : les pics des données augmentées (points roses) s'alignent souvent avec les clusters des classes minoritaires dans la distribution initiale (zigzags bleus).
En résumé, l'augmentation des données améliore la représentation des classes minoritaires, confirmant son efficacité pour corriger le déséquilibre des données.

5.0.3 Performance du clustering

Dans cette section, nous présentons l'analyse des performances du clustering, résumée dans le Tableau 3. Les résultats comparent les métriques de clustering (NMI et ARI) sur nos cinq ensembles de données, pour deux algorithmes : KMeans et Spherical KMeans, en utilisant les données avant et après augmentation. Les métriques, moyennées sur cinq exécutions, montrent que l'augmentation améliore souvent les performances, en particulier pour l'ARI, et maintient des résultats comparables même dans les pires scénarios, garantissant ainsi que la qualité du clustering n'est pas compromise.

6 Conclusion

Dans ce travail, nous avons proposé un nouveau cadre d'augmentation des données intégrant les GMMs et les LLMs afin de répondre aux défis liés au déséquilibre des classes dans les tâches de traitement automatique du langage naturel non supervisées. Notre approche cible spécifiquement les régions sous-représentées des ensembles de données, en utilisant les GMMs pour identifier avec précision ces clusters et les LLMs pour générer des documents synthétiques contextuellement pertinents. À travers des expériences approfondies sur plusieurs ensembles de données textuelles déséquilibrés, nous avons démontré que notre méthode maintient les performances du clustering, améliore la représentation des classes minoritaires et enrichit l'interprétabilité des clusters. Les résultats montrent que l'aug-

mentation ciblée des données est une stratégie efficace pour pallier les déséquilibres tout en préservant la qualité des algorithmes de clustering.
Dans cette contribution, nous avons utilisé des GMMs, mais il serait pertinent d'explorer d'autres modèles de mélanges, comme les modèles von-Mises Fisher [17, 18] ou les modèles de blocs latents [7, 14]. Une limitation de l'utilisation des données générées par des LLMs est le risque de renforcer les biais des ensembles d'entraînement [19]. Bien que notre approche aligne le texte généré avec les thèmes des clusters, les travaux futurs devraient évaluer et atténuer ces biais dans les données synthétiques.

Références

[1] Jeffrey D. Banfield and Adrian E. Raftery. Model-based gaussian and non-gaussian clustering. *Biometrics*, 49(3) :803–821, 1993.

[2] Markus Bayer, Marc-André Kaufhold, and Christian Reuter. A survey on data augmentation for text classification. *ACM Computing Surveys*, 55(7) :1–39, 2022.

[3] Xunxin Cai, Meng Xiao, Zhiyuan Ning, and Yuanchun Zhou. Resolving the imbalance issue in hierarchical disciplinary topic inference via llm-based data augmentation. In *ICDMW*, pages 1424–1429, 2023.

[4] Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Wei Liu, Ninghao Liu, et al. Auggpt : Leveraging chatgpt for text data augmentation. 2023.

[5] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society : Series B*, 39(1) :1–22.

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[7] Gérard Govaert and Mohamed Nadif. Clustering with block mixture models. *Pattern Recognition*, 36(2) :463–473, 2003.

[8] Hongyu Guo, Yongyi Mao, and Richong Zhang. Augmenting data with mixup for sentence classification : An empirical study, 2019.

[9] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. 2023.

[10] Akbar Karimi, Leonardo Rossi, and Andrea Prati. Aeda : An easier data augmentation technique for text classification, 2021.

[11] Imed Keraghel, Stanislas Morbieu, and Mohamed Nadif. Beyond words : a comparative analysis of LLM embeddings for effective clustering. In *IDA*, pages 205–216, 2024.

[12] Varun Kumar, Ashutosh Choudhary, and Eunah Cho. Data augmentation using pre-trained transformer models. 2020.

[13] Leland McInnes, John Healy, and James Melville. Umap : Uniform manifold approximation and projection for dimension reduction. 2018.

[14] Mohamed Nadif and Gérard Govaert. Block clustering of contingency table and mixture model. In *International Symposium on Intelligent Data Analysis*, pages 249–259. Springer, 2005.

[15] Siyuan Qiu, Binxia Xu, Jie Zhang, Yafang Wang, Xiaoyu Shen, Gerard De Melo, Chong Long, and Xiaolong Li. Easyaug : An automatic textual data augmentation platform for classification tasks. In *Companion proceedings of the web conference 2020*, pages 249–252, 2020.

[16] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. 2015.

[17] Aghiles Salah and Mohamed Nadif. Model-based von mises-fisher co-clustering with a conscience. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 246–254. SIAM, 2017.

[18] Aghiles Salah and Mohamed Nadif. Directional co-clustering. *Advances in Data Analysis and Classification*, 13 :591–620, 2019.

[19] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022) :755–759, Jul 2024.

[20] Douglas Steinley. Properties of the hubert-arable adjusted rand index. *Psychological methods*, 9(3) :386, 2004.

[21] Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec) :583–617, 2002.

[22] Jens Van Nooten and Walter Daelemans. Improving dutch vaccine hesitancy monitoring via multi-label data augmentation with gpt-3.5. In *the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, volume 1, pages 251–270, 2023.

[23] Jason Wei and Kai Zou. Eda : Easy data augmentation techniques for boosting performance on text classification tasks, 2019.

[24] M Anthony Wong. A hybrid clustering method for identifying high-density clusters. *Journal of the American Statistical Association*, 77(380) :841–847, 1982.

[25] Junjie Ye, Nuo Xu, Yikun Wang, Jie Zhou, Qi Zhang, Tao Gui, and Xuanjing Huang. Llm-da : Data augmentation via large language models for few-shot named entity recognition. *arXiv preprint arXiv :2402.14568*, 2024.

Communications

Une Approche Topologique de l'Analyse Discriminante

R. Abdesselam

Université Lumière Lyon 2, Laboratoires ERIC & COACTIS

rafik.abdesselam@univ-lyon2.fr

Résumé

L'objectif de ce travail est de proposer une analyse discriminante topologique selon les différents types de données avec des variables explicatives quantitatives, qualitatives ou mixtes. Cette analyse topologique décisionnelle est une méthode de classification supervisée qui tente de découvrir les structures intrinsèques et les informations discriminantes intégrées dans les données. Il existe de nombreuses techniques prédictives, elles sont le plus souvent et le plus utilement appliquées à divers problèmes dans de nombreux domaines. L'approche topologique de discrimination proposée est basée sur la notion de graphes de voisinage dans un contexte décisionnel. Les variables explicatives sont plus ou moins corrélées ou liées selon que les variables sont de type quantitatives, qualitatives ou un mélange des deux. Ce modèle de discrimination topologique analyse la structure des corrélations ou dépendances observées dans chaque classe en fonction des variables explicatives. Pour valider l'efficacité de notre approche topologique, une série d'expériences est réalisée sur plusieurs jeux de données réelles de référence UCI, avec des variables explicatives quantitatives, qualitatives et mixtes. Les résultats sont comparés à ceux des modèles prédictifs existants de l'apprentissage automatique (Machine Learning).

Mots-clés

Mesure de proximité, graphe de voisinage, matrice d'adjacence, analyse discriminante, modèles prédictifs.

Abstract

The objective of this work is to propose a topological discriminant analysis according to the type of data, quantitative, qualitative or mixed explanatory variables. This decisional topological classification is a supervised clustering method attempts to discover the intrinsic structures and discriminant information embedded in the data. There are many predictive techniques, they are most often and most usefully applied to various problems in many fields. The proposed topological approach of discrimination is based on the notion of neighborhood graphs in a decisional context. The explanatory variables are more or less correlated or linked depending on whether the variables type, quantitative, qualitative or a mixture of both. This topological model of discrimination analyzes the structure of the correlations or dependencies observed in each class according to the explanatory variables. To validate the effectiveness of our topological approach, a series of experiments are performed on several UCI benchmark datasets, with quantitative, qualitative and mixed explanatory variables. The results are compared to those of existing machine learning predictive models.

Keywords

Proximity measure, neighborhood graph, adjacency matrix, discriminant analysis, predictive models.

1 Introduction

L'analyse discriminante topologique (ADT) proposée est un modèle prédictif comparable aux nombreuses techniques d'apprentissage automatique (machine learning) existantes, qui est une forme d'intelligence artificielle utilisée pour créer des modèles prédictifs. L'ADT est notamment comparée à l'analyse discriminante de Fisher (AD), méthode de classification utilisée depuis longtemps dans de nombreux contextes, à sa rivale, la régression logistique (RL) ainsi qu'aux autres modèles machine learning existants [15], tels que les algorithmes de regroupement les plus populaires : K-plus proches voisins (KNN), Support Vecteur Machine (SVM), Forêt aléatoire (RF), Réseau bayésien (BN), arbre de décision (DT), Réseau de neurones (NN).
Les techniques du machine learning sont à la fois explicatives et prédictives. Elles permettent de vérifier l'appartenance de groupes d'individus distincts, d'identifier leurs caractéristiques à partir de variables explicatives et de prédire le groupe d'appartenance d'un nouvel individu.
La modélisation prédictive est largement utilisée dans divers secteurs et domaines. En marketing, elle permet de prédire les préférences des consommateurs, de personnaliser les offres promotionnelles et de cibler la publicité en ligne. En finance et assurance (scoring de crédit), la modélisation prédictive joue un rôle clé dans l'analyse des risques, la prévision des tendances du marché et la gestion des investissements. En santé et médecine, ce type de modèle peut être utilisé pour prédire les diagnostics médicaux. Cela permet d'identifier les tendances en matière de santé, de personnaliser le traitement des patients et d'améliorer la gestion

des dossiers médicaux. La modélisation prédictive est également de plus en plus utilisée dans différents autres secteurs, la technologie et l'Internet, des ressources humaines et des sciences environnementales pour améliorer la prise de décisions.
L'AD suppose que les variables explicatives sont normalement distribuées et que les matrices de covariance intra-groupes sont égales. Cependant, elle est étonnamment résistante aux violations de ces hypothèses et constitue généralement un bon modèle de classification supervisée et de prise de décisions. Il existe des approches spécifiques à l'AD, mais à notre connaissance aucune de ces approches n'a été proposée dans un contexte topologique.
L'objectif de cet article est de proposer une approche topologique d'analyse discriminante sur variables explicatives quantitatives, qualitatives ou mixtes.
Le choix de la mesure de proximité parmi les nombreuses mesures existantes, joue un rôle important en analyse de données multidimensionnelles [4,15]. Il a un fort impact sur les résultats de toute opération de structuration, de regroupement ou de classification d'objets. La structure de corrélation ou de dépendance des variables quantitatives ou qualitatives dépend des données considérées. Les résultats peuvent changer en fonction de la mesure de proximité choisie.

2 Contexte topologique de discrimination

L'analyse discriminante topologique consiste à analyser simultanément chaque table de données associée à chacune des modalités-classes de la variable cible à discriminer. On utilisera les notations suivantes :
- $Y_{(n,q)}$ est la matrice des données associée aux q variables binaires $\{y^k; k = 1, q\}$ de la variable qualitative cible à expliquer y à q modalités-groupes à discriminer,
- $X_{(n,p)}$ est la matrice des données associée aux p variables continues explicatives, ensemble de p variables discriminantes $\{x^j; j = 1, p\}$, à $n = \sum_{k=1}^{q} n_k$ lignes-individus et p colonnes-variables,
- $X_{k(n_k,p)}$ est la matrice des données associée aux p variables explicatives à n_k individus ayant la k^{ime} modalité de y,
- $Z_{(n,r)}$ est la matrice des données associée aux r variables qualitatives explicatives, ensemble de r variables discriminants $\{z^j; j = 1, r\}$, à $n = \sum_{k=1}^{r} n_k$ lignes-individus et r colonnes–variables qualitatives,
- $Z_{r(n_k,r)}$ est la matrice des données associée aux r variables qualitatives à n_k individus ayant la k^{ime} modalité de y.
Etant donnée une mesure de proximité u_k, on peut définir une relation de voisinage, V_{u_k}, comme étant une relation binaire basée sur $E_k \times E_k$, $E_k = \{x^1, \cdots, x^j, \cdots, x^p\}$ étant l'ensemble des p variables explicatives. Plusieurs définitions sont possibles pour construire cette relation de voisinage binaire, on peut choisir l'Arbre de Longueur Minimal(ALM)[7], le Graphe de Gabriel (GG)[10] ou, comme c'est le cas ici, le Graphe de Voisinage Relatif (GVR)[12]. Étant donné un ensemble E_k de p variables du tableau de données X_k et une mesure de proximité u_k, pour des données continues ou binaires, on construit la matrice d'adjacence associée V_{u_k} d'ordre p, où toutes les paires de variables voisines dans E_k satisfont la propriété GVR suivante :

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

Cela signifie que si deux variables x^l et x^r qui vérifient la propriété GVR sont connectées par une arête, les sommets x^l et x^r sont voisins.

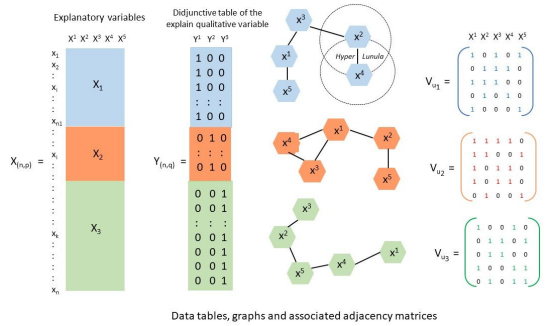

FIGURE 1 – GVR - Matrices d'adjacence pour la discrimination topologique

La Figure 1 présente un exemple illustratif avec cinq variables explicatives quantitatives $\{x^j; j = 1, 5\}$ et $y = \{y^k; k = 1, 3\}$ une variable qualitative cible à expliquer à trois modalités-groupes. A partir des tableaux de données de chaque groupe $X_1 = (X/Y = 1), X_2 = (X/Y = 2)$ et $X_3 = (X/Y = 3)$, on établit les matrices d'adjacence binaires associées V_{u_1}, V_{u_2} et V_{u_3}, selon la structure de voisinage et les mesures de proximité choisies u_1, u_2 et u_3.
Par exemple, pour le tableau de données X_1, on voit que pour la première et la quatrième variable, $V_{u_1}(x^1, x^4) = 1$, cela signifie que sur le plan géométrique, l'hyper-Lunule (intersection entre les deux hypersphères centrées sur les deux variables x^1 et x^4) est vide.

2.1 Matrices d'adjacence de référence

L'objectif est d'analyser de manière topologique les structures de corrélation ou de dépendance des variables explicatives des données considérées. Les expressions des matrices de référence d'adjacence appropriées sont décrites dans [2,3] en fonction du type de variables explicatives considérées, quantitatives ou qualitatives ou un mélange des deux.

2.1.1 Variables explicatives quantitatives

Nous construisons la matrice d'adjacence notée $V_{u^\star}$, qui correspond le mieux à la matrice de corrélation. Ainsi, pour

examiner la structure de corrélation entre les variables, nous regardons la significativité de leur coefficient de corrélation linéaire. Cette matrice d'adjacence peut s'écrire comme suit en utilisant le test t ou le test t de Student du coefficient de corrélation linéaire de Bravais-Pearson :
Pour chaque tableau de données $X_k = (X/Y = k)$, nous construisons la matrice d'adjacence de référence notée $V_{u_k^\star}$ à partir de la matrice de corrélation du tableau de données X_k.
Pour chaque tableau de données X_k, la matrice d'adjacence de référence $V_{u_k^\star}$ associée à la mesure de référence $u_k^\star$ est définie ainsi :

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

Où, T_{n-2} désigne la loi de Student à $\nu = n - 2$ degrés de liberté et la p-value, le seuil de signification du test du coefficient de corrélation linéaire.
La matrice d'adjacence de référence $V_{u_k^\star}$ ainsi construite est associée à une mesure de proximité de référence inconnue, notée $u_k^\star$.

2.1.2 Variables qualitatives explicatives

L'objectif est d'analyser la structure de dépendance à savoir s'il existe une association topologique entre toutes ces variables qualitatives. Pour chaque tableau de données Z_k, on construit la matrice d'adjacence de référence $V_{u_k^\star}$, selon l'association significative entre les variables du tableau Z_k, obtenue à partir du tableau de Burt $\mathcal{B}_k$.
Pour chaque table de données Z_k, la matrice d'adjacence de référence $V_{u_k^\star}$ associée à la mesure de référence $u_k^\star$ est ainsi définie :

$$V_{u_k^\star}(z^{ht}, z^{ls}) = \begin{cases} 1 & \text{if } \frac{\mathcal{B}_{htls}}{\mathcal{B}_{ht\,\cdot\cdot}} \geq \frac{\mathcal{B}_{ht\,\cdot\cdot}}{nq^2}; \ \forall h, l = 1, r\, t = 1, m_h \\ & \qquad\qquad and\, s = 1, m_l \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Ainsi, pour examiner les similitudes entre les modalités des variables, nous examinons l'écart entre chaque profil-modalité et son profil moyen, c'est-à-dire l'écart vers l'indépendance.

2.1.3 Variables explicatives mixtes

Le traitement simultané de données mixtes (quantitatives et qualitatives) ne peut être réalisé directement par les méthodes conventionnelles d'analyse de données. Ainsi, nous transformons tout d'abord les données qualitatives en données quantitatives [1]. Cette transformation est basée sur l'analyse de variance multivariée (MANOVA) et sur la maximisation du critère mixte, proposé en termes de carrés de corrélation par Tenenhaus [11] et géométriquement en termes de cosinus carrés des angles par Escofier [5].
Dans un deuxième temps, on construit la matrice d'adjacence $V_{u^\star}$, associée à la mesure de proximité de référence $u^\star$, à partir de la matrice de corrélation de toutes les variables, quantitatives et qualitatives transformées, selon l'expression (1).

2.2 Analyse Discriminante Topologique

On utilisera les notations matricielles suivantes :

$$X_{(n,p)} = \begin{pmatrix} X_{1(n_1,p)} \\ \vdots \\ X_{k(n_k,p)} \\ \vdots \\ X_{q(n_q,p)} \end{pmatrix} \quad Z_{(n,r)} = \begin{pmatrix} Z_{1(n_1,r)} \\ \vdots \\ Z_{k(n_k,r)} \\ \vdots \\ Z_{q(n_q,r)} \end{pmatrix} \quad Y_{(n,q)} = \begin{pmatrix} Y_{1(n_1,q)} \\ \vdots \\ Y_{k(n_k,q)} \\ \vdots \\ Y_{q(n_q,q)} \end{pmatrix}$$

$$V_{u^\star\,(qp,p)} = \begin{pmatrix} V_{u_1^\star(p)} \\ \vdots \\ V_{u_k^\star(p)} \\ \vdots \\ V_{u_q^\star(p)} \end{pmatrix} \quad \widehat{X}_{(n,p)} = \begin{pmatrix} X_1 V_{u_1^\star} \\ \vdots \\ X_k V_{u_k^\star} \\ \vdots \\ X_q V_{u_q^\star} \end{pmatrix}$$

- $V_{u_k^\star}$ est la matrice symétrique d'adjacence d'ordre p, associée à la mesure de proximité de référence $u_k^\star$, qui résume le mieux la structure des corrélations du tableau de données X_k ou la structure de dépendance du tableau de données Z_k,
- $\widehat{X}_{(n,p)} = Diag[X]V_{u^\star} = XV_{u^\star}$ est la matrice des données projetées à n individus et p variables,
- M_p est la matrice des distances d'ordre p dans l'espace des individus,
- $D_n = \frac{1}{n} I_n$ est la matrice diagonale des poids d'ordre n dans l'espace des variables.

On analyse d'abord de manière topologique, la structure de corrélation des variables à l'aide d'une ACP topologique, qui consiste à réaliser l'ACP [8] du triplet $(\widehat{X}, M_p, D_n)$ de la matrice de données projetée $\widehat{X} = XV_{u^\star}$, puis nous procédons à une analyse discriminante sur les composantes principales significatives de l'ACP topologique précédente.

L'approche ADT proposée, consiste à effectuer une analyse discriminante sur les facteurs significatifs de l'analyse en composantes principales topologiques du triplet $(\widehat{X}, M_p, D_n)$.

3 Exemple illustratif

Pour illustrer l'approche ADT, on a établi un Benchmark de plusieurs jeux de données extraits du référentiel UCI Machine Learning sur différents thèmes et avec différentes dimensions des données [13,6]. L'approche topologique proposée a été testée sur sept bases de données réelles, les résultats obtenus, présentés dans le Tableau 1, ont été comparés à ceux de l'analyse discriminante et de la régression logistique. L'exemple illustratif traite les caractéristiques bancaires des clients d'une agence bancaire (jeu de données n°2).

TABLE 1 – Benchmark - Bases de données

Explanatory Variables	Data	n	p	q	Well classified (%) TDA	DA	LR
1 - Continuous	Iris	150	4	3	88.00	98.00	**98.67**
2 - Mixed	Credit bank	468	8	2	**97.01**	88.46	76.28
3 - Continuous	Brands of bottled water	38	8	2	89.47	89.47	**94.74**
4 - Continuous	Wine	178	13	3	**100.00**	99.44	**100.00**
5 - Categorical	Bank customer	3808	6	2	**90.21**	62.61	89.15
6 - Continuous	Raisin	900	6	2	**100.00**	86.33	**100.00**
7 - Continuous	Wine Quality	6497	12	2	**99.65**	99.48	99.54

Les statistiques descriptives des variables explicatives mixtes sont présentées dans le tableau 2 ainsi que celles de la variable cible à discriminer dans le tableau 3.

TABLE 2 – Statistiques sommaires des variables bancaires explicatives mixtes

Credit bank Data Continuous variables	Mean	Standard Deviation	Coefficient of variation	Min	Max
Savings Amount	1040.79	2884.77	2.77	0.00	21000.00
Years Seniority	6.17	5.35	0.87	0.50	28.00
Average outstanding	759.18	381.24	0.50	145.00	2315.00
Average Banking Transactions	5277.01	3950.83	0.75	450.00	17500.00
Average cumulative debits	70.02	43.50	0.62	8.00	187.00

Modalities of Categorical variables	Frequency	Percentage	Cummuled
Domiciled salary	316	67.52	67.52
Non-domiciled salary	152	32,48	32.48
Total	468	100.00	100.00
Authorized-Overdraft	202	43.16	43.16
Prohibited-Overdraft	266	56.84	56.84
Total	468	100.00	100.00
Authorized-Checkbook	415	88.68	88.68
Prohibited-Checkbook	53	11.32	11.32
Total	468	100.00	100.00

TABLE 3 – Statistiques de la variable cible

Modality	Frequency	Percentage	Cummuled
Good customers	237	50.64	50.64
Bad customers	231	49.36	49.36
Total	468	100.00	100.00

La matrice d'adjacence de référence globale $V_{u^\star}$ présentée dans le Tableau 4 est associée à la mesure de proximité $u^\star$ la plus adaptée aux données considérées, elle est construite à partir des matrices de corrélation des tableaux de données X_1 et X_2 selon l'expression (1).

TABLE 4 – Matrices globales de correlations et d'adjacence de référence

$$R = \begin{pmatrix} R_1 & 0 \\ 0 & R_2 \end{pmatrix} \qquad V_{u^*} = \begin{pmatrix} V_{u_1^*} & 0 \\ 0 & V_{u_2^*} \end{pmatrix}$$

Dans ce cas de variables explicatives mixtes, les données qualitatives ont d'abord été transformées en données quantitatives, puis on a considéré toutes les données quantitatives, c'est-à-dire toutes les variables quantitatives et qualitatives transformées.

Notons que deux variables quantitatives corrélées positivement sont liées et deux variables corrélées négativement sont liées, mais distantes, nous prendrons donc en compte le signe de la corrélation entre les variables dans la matrice d'adjacence.

On a effectué une ACP Topologique non normée pour identifier la structure de corrélation de toutes les variables, une analyse discriminante est ensuite appliquée sur les composantes principales significatives de cette ACP Topologique des données projetées.

La figure 2 présente sur le premier plan factoriel de l'ACP Topologique, les corrélations entre les composantes

principales-facteurs et les variables initiales. Les deux premiers facteurs de l'ACP topologique expliquent respectivement $79,84\%$ et $19,68\%$, soit $99,53\%$ de la variation totale de l'ensemble des données, ils fournissent une synthèse adéquate des données mixtes, c'est-à-dire des caractéristiques bancaires des clients de l'agence.

TABLE 5 – ADT - Topological Discriminant Analysis on mixed variables

Fisher linear function Variable	Coefficient function Discriminant	Standard Deviation	Ratio t-Student	Proba p-value
Good customers				
Domiciled salary	0.0000	0.0000	3.40**	0.00034
Authorized-Overdraft	0.0000	0.0000	3.37**	0.00038
Authorized-Checkbook	0.0001	0.0000	15.44**	0.00001
Bad customers				
Savings Amount	-0.0000	0.0000	-0.94	0.23860
Years Seniority	-0.0000	0.0000	-0.94	0.23860
Average outstanding	-0.0000	0.0000	-0.94	0.23860
Average Banking Transactions	-0.0000	0.0000	-0.94	0.23860
Average cumulative debits	-0.0001	0.0000	-19.76**	0.00001
Non-domiciled salary	-0.0000	0.0000	-3.40**	0.00034
Prohibited-Overdraft	-0.0000	0.0000	-3.37**	0.00038
Prohibited-Checkbook	-0.0001	0.0000	-15.44**	0.00001
Constant	-0.668013	-0.147380		
R2 = 0.5829	F = 324.8962	PROBA = 0.001		
D2 = 5.5666	T2 = 651.1898	PROBA = 0.001		

Significance level α : $^{**}\alpha \leq 1\%$; $^{*}\alpha \in]1\%; 5\%]$

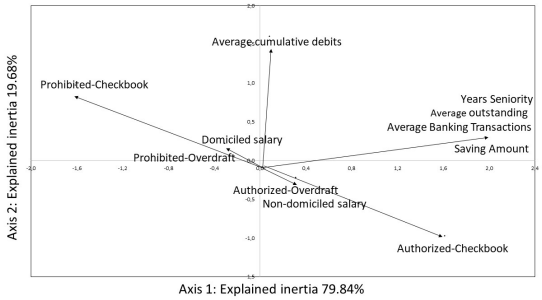

FIGURE 2 – Représentation des variables explicatives mixtes

Le tableau 5 résume les profils significatifs des deux groupes de clients ; avec un risque d'erreur inférieur ou égal à 5%. Pour établir les profils des groupes de clients de l'agence bancaire, une Analyse Discriminante de Fischer a été appliquée sur les variables explicatives mixtes.
Les coefficients de la fonction discriminante qui sépare au mieux les deux groupes de clients, ont été classés dans le Tableau 5, selon la valeur du test t-student.

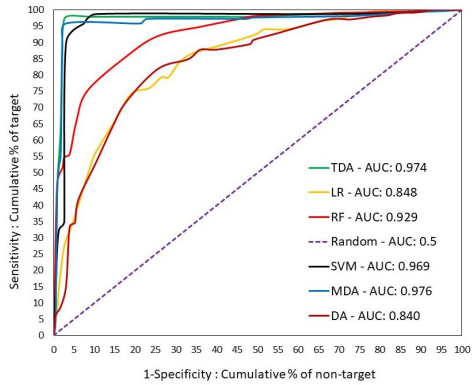

FIGURE 3 – Courbes ROC

Le premier groupe composé de 237 "bons clients" est caractérisé par une domiciliation en agence du salaire du client et des autorisations de découvert et de chéquier. Le second groupe constitué de 231 "mauvais clients" se caractérise par une combinaison de débits moyens élevés, de salaire non domicilié dans l'agence et d'interdictions de découvert et de chéquier.

À titre de comparaison, nous avons considéré 8 approches d'apprentissage supervisé les plus populaires, à savoir, l'Analyse Discriminante classique (AD), l'Analyse Discriminante Mixte (ADM), La Régression logistique (RL), les k plus proches voisins (k-NN), Support Vecteur Machine (SVM), Forêt aléatoire (RF), arbre de décision (DT), Réseau de neurones (NN). Le Tableau 6 et la Figure 3 résument les performances des modèles prédictifs appliqués aux données mixtes considérées. Ils présentent les matrices de confusion, les courbes ROC (Receiver Operating Characteristic) et les résultats AUC (Area Under the Curve) du modèle ADT proposé et des principaux modèles d'apprentissage automatique utilisés. Le modèle prédictif topologique ADT, suivi des modèles MDA [Abdesselam (2008)] et SVM [Marjanović et al.(2011)], donnent d'excellents résultats de discrimination.
Par ailleurs, un tableau résume les performances des techniques d'apprentissage automatique appliquées sur sept ensembles de données pour la classification binaire, avec différents types de variables explicatives et différentes dimensionnalités des données. Les expériences sur ce benchmark confirment les bonnes performances du modèle prédictif ADT topologique.

TABLE 6 – Résultats expérimentaux

ADT

Actual classification	Predicted classification: Good customers	Bad customer	Total
Good customers	228	9	237
Bad customers	5	226	231
Total	233	235	468

% of well classified : 97.01% AUC-ROC : 0.977

DA

Actual classification	Predicted classification: Good customers	Bad customers	Total
Good customers	183	54	237
Bad customers	50	181	231
Total	233	235	468

% of well classified : 77.78% AUC-ROC : 0.840

MDA

Actual classification	Predicted classification: Good customers	Bad customer	Total
Good customers	223	14	237
Bad customers	5	226	231
Total	228	240	468

% of well classified : 95.94% AUC-ROC : 0.976

LR

Actual classification	Predicted classification: Good customers	Bad customers	Total
Good customers	187	50	237
Bad customers	61	170	231
Total	248	220	468

% of well classified : 76.28% AUC-ROC : 0.848

SVM

Actual classification	Predicted classification: Good customers	Bad customers	Total
Good customers	226	11	237
Bad customers	14	217	231
Total	240	228	468

% of well classified : 94.66% AUC-ROC : 0.969

DT

Actual classification	Predicted classification: Good customers	Bad customers	Total
Good customers	192	45	237
Bad customers	45	186	231
Total	237	231	468

% of well classified : 80.77%

RF

Actual classification	Predicted classification: Good customers	Bad customers	Total
Good customers	191	46	237
Bad customers	60	171	231
Total	251	217	468

% of well classified : 83.76% AUC-ROC : 0.929

NN

Actual classification	Predicted classification: Good customers	Bad customers	Total
Good customers	207	30	237
Bad customers	29	202	231
Total	236	232	468

% of well classified : 87.39%

4 Conclusion

Cet article propose une approche topologique de l'analyse discriminante (ADT) qui peut enrichir les méthodes classiques d'analyse de données dans le cadre des modèles prédictifs.
Les performances du modèle ADT, basé sur la notion de graphe de voisinage, obtenues à partir d'un benchmark de base de données, sont aussi bonnes, voire meilleures, selon les critères du pourcentage de bien classés et de l'aire sous la courbe ROC, que celles des autres modèles d'apprentissage automatique existants.
La méthode de classement ADT peut être mise en œuvre à partir des procédures d'analyse en composantes principales et de discrimination des logiciels SAS, SPAD ou R.
Il serait intéressant d'étendre cette approche topologique à d'autres modèles prédictifs d'analyse de données, notamment dans le cadre de la régression multiple.

Références

[1] R. Abdesselam, Analyse en composantes principales mixte. *Revue des Nouvelles Technologies de l'Information, Classification : points de vue croisés*. Cépaduès Ed., pp. 31–41, 2008.

[2] R. Abdesselam, A topological multiple correspondence analysis. *Journal of Mathematics and Statistical Science* 5, 8, pp. 175–192, 2019.

[3] R. Abdesselam, A topological clustering of variables. *Journal of Mathematics and System Science* 11, 2, 1–17, 2021.

[4] Batagelj, V., Bren, M. (1995) *Comparing resemblance measures*. In Journal of classification, 12, 73–90, 1995.

[5] Escofier, B. et Pagès, J. (1985) Mise en oeuvre de l'AFM pour des tableaux numériques, qualitatifs, ou mixtes. Publication interne de l'IRISA, 429, 1985.

[6] Govaert, G. : Analyse des données (2003). Hermes Science, Lavoisier.

[7] Kim, J.H. and Lee, S. (2003) Tail bound for the minimal spanning tree of a complete graph. *In Statistics & Probability Letters*, 4, 64, 425–430.

[8] Lebart, L. (1989) *Stratégies du traitement des données d'enquêtes*. La Revue de MODULAD, 3, 21–29, 1989.

[9] Marjanović, M., Kovačević, M., Bajat, B., Voženílek, V. : Landslide susceptibility assessment using SVM machine learning algorithm. *Engineering Geology, Elsevier,* Volume 123, Issue 3, 225–234, 2011.

[10] Panagopoulos, D. (2022) *Topological data analysis and clustering*. Chapter for a book, Algebraic Topology (math.AT) arXiv :2201.09054, Machine Learning.

[11] Park, J. C., Shin, H. and Choi, B. K. (2006) *Elliptic Gabriel graph for finding neighbors in a point set and its application to normal vector estimation*. In Computer-Aided Design Elsevier, 38, 6, 619–626.

[12] Tenenhaus, M. (1977) Analyse en composantes principales d'un ensemble de variables nominales ou numériques. *Revue de statistique appliquée*, tome 25, no 2, 39–56, 1977.

[13] G.T. Toussaint, The relative neighbourhood graph of a finite planar set. *In Pattern recognition*, 12, 4, pp. 261-268, 1980.

[14] UCI Machine Learning Repository (1988), https ://archive.ics.uci.edu/datasets.

[15] Vafeiadisa,T., Diamantarasb, K-I., Sarigiannidisa, G., Chatzisavvasa, K-Ch. (2015) A comparison of machine learning techniques for customer churn prediction *Simulation Modelling Practice and Theory*.

[16] Ward, J.R. (1963) *Hierarchical grouping to optimize an objective function*. In Journal of the American statistical association JSTOR, 58, 301, 236–244, 1963.

[17] D. Zighed, R. Abdesselam, A. Hadgu, Topological comparisons of proximity measures. 16th PAKDD Conference (Part I, LNAI 7301), pp. 379–391, 2012.

Étude de variabilité par bootstrap résiduel pour une méthode de subspace clustering

Yasmine Agliz [1], Vincent Audigier[1], Mohamed Nadif[2], Ndèye Niang[1]

[1] CNAM, CEDRIC
[2] Université de Paris Cité, Centre Borelli

yasmine.agliz@lecnam.net, vincent.audigier@cnam.fr, n-deye.niang_keita@cnam.fr, mohamed.nadif@u-paris.fr

Résumé

Une stratégie classique pour la classification d'observations non supervisée en grande dimension consiste à identifier simultanément une partition des observations et un sous-espace de représentation des données qui met en évidence cette partition. Cette stratégie est efficace car elle relie les tâches de classification et de recherche du sous-espace. Dans ce contexte, ce travail vise à évaluer la sensibilité de la partition et du sous-espace aux fluctuations d'échantillonnage. Pour cela, nous proposons d'abord une méthode pour simuler ces fluctuations à l'aide de différentes approches de bootstrap. Cette première étape permet, par exemple, de visualiser la sensibilité des résultats en construisant des ellipses de confiance autour des centres de gravité. Ensuite, nous proposons une manière de quantifier cette sensibilité. La procédure est évaluée par simulation sur différentes structures de données. Les résultats montrent que l'approche bootstrap permet d'estimer correctement la sensibilité de la méthode, à condition que la stratégie de bootstrap choisie soit adaptée à la structure des données.

Mots-clés

Classification non supervisée, grande dimension, bootstrap résiduel.

Abstract

Combining dimensionality reduction and clustering techniques is effective for clustering high-dimensional data by simultaneously identifying low-dimensional subspaces and their corresponding partitions. This work aims to assess the sensitivity of the partition and the subspace to sampling fluctuations. To achieve this, we first propose a way to simulate these fluctuations using different bootstrap approaches. This initial step, for instance, provides a means of visualizing the sensitivity of the results through the construction of confidence ellipses around the centroids. We then propose a method to quantify this sensitivity. The proposed procedure is evaluated through simulations on different data structures. The results show that, with the proposed bootstrap approach, it is possible to correctly assess the sensitivity of the method, provided that the chosen bootstrap strategy matches the structure of the data.

Keywords

Clustering, High dimensionality, Residual Bootstrap.

1 Introduction

En classification non supervisée en grande dimension, les variables redondantes ou non pertinentes sont fréquentes et peuvent compliquer l'identification des groupes d'individus [9]. Pour y remédier, certaines méthodes visent à identifier des sous-espaces où les groupes sont bien séparés. Parmi ces méthodes, on trouve le Reduced K-means (RKM) [2], le Factorial K-means (FKM) [11], et la méthode du semiNMF-PCA [1]. Dans ce travail, nous nous concentrons sur le RKM. Le Reduced K-means est une méthode qui étend l'algorithme classique K-means en intégrant une réduction de dimension, permettant ainsi de regrouper les observations dans un sous-espace commun. Le RKM repose conjointement sur l'estimation des vecteurs engendrant le sous-espace et la classification des observations dans l'espace réduit. Du point de vue modélisétion, cette méthode suppose que les observations suivent une structure fixe des groupes d'individus dans un sous-espace, perturbée par du bruit, c'est-à-dire un modèle à effets fixes (*fixed-effect model*).

Une des limites de ces approches géométriques réside dans le fait qu'elles ne prennent pas en compte la sensibilité des résultats aux fluctuations d'échantillonnage. Une approche naturelle pour refléter cette sensibilité consiste à effectuer du bootstrap. Cette approche a fait l'objet de différents travaux, par exemple, Dudoit et al. [4] proposent une méthode inspirée du *bagging* pour améliorer la stabilité des partitions en agrégeant les résultats obtenus à partir d'échantillons bootstrap. Ils appliquent un algorithme de classification à des jeux de données rééchantillonnés et combinent les partitions résultantes par vote ou en construisant une nouvelle matrice de dissimilarité. De même, Dolnicar et al. [3] utilisent une approche similaire, mais agrègent les centroïdes des classes par classification hiérarchique avant d'assigner les observations originales aux classes finales. D'autres travaux, comme ceux de Hofmans et al. [5], se concentrent sur l'évaluation de la fiabilité des résultats de classification. Ils soulignent que le K-means, en tant que méthode déterministe, ne fournit pas d'estimations d'incertitude pour les centres des classes et les assignations. Pour remédier à cela,

ils proposent une procédure de bootstrap pour construire des régions de confiance autour des centroïdes et estimer les probabilités d'appartenance des observations aux classes. Cependant, on ne retrouve aucune application de ces approches par bootstrap pour le RKM, ce qui pose des difficultés pour évaluer la sensibilité des résultats de la méthode. Parallèlement, dans le domaine de la réduction de dimension, de nombreux auteurs se sont intéressés à l'inférence en ACP. Par exemple, Josse et al. [7] étudient la variabilité de l'ACP dans le cadre du modèle à effets fixes. Ils proposent plusieurs approches pour évaluer la variabilité des paramètres, notamment le bootstrap paramétrique, qui s'appuie sur la matrice de résidus obtenue à partir de la différence entre la structure fixe de projection et les données observées. D'autres travaux, comme ceux de Josse et al. [6], se concentrent sur des représentations visuelles pour analyser cette variabilité. Ces auteurs utilisent l'analyse Procrustes pour aligner plusieurs plans factoriels sur une configuration de référence, permettant ainsi de calculer des zones de confiance pour les individus et de visualiser la dispersion des estimations.

Pour quantifier la variabilité des paramètres estimés, nous proposons une approche basée sur des distances matricielles, définies à l'aide de différentes normes, telles que la norme de Frobenius. Une étape consiste à aligner les matrices correspondant aux différents paramètres avant de les comparer. Ensuite, en calculant une mesure de dissimilarité moyenne entre les différentes estimations, cette approche permet d'offrir une quantification robuste et interprétable de la variabilité des paramètres

Nous nous concentrons sur le modèle de bruit à effets fixes, où une matrice de données $\mathbf{X}$ a une structure fixe déterministe avec du bruit aléatoire. Une question clé que nous abordons est la suivante : si l'échantillon change, comment les paramètres estimés par RKM varient-ils ? L'apport de ce travail est de fournir un cadre pour évaluer statistiquement la stabilité des estimations du sous-espace et des centroïdes de RKM. Pour cela, nous approchons la distribution empirique des estimateurs par bootstrap, quantifions numériquement leur variabilité des estimateurs à l'aide de mesures adaptées. La Section 2 présente la méthode de *subspace clustering* utilisée pour l'estimation des partitions et du sous-espace, ainsi que l'approche de bootstrap employée pour estimer la variabilité des paramètres. La Section 3 propose une étude par simulation des différentes approches de bootstrap envisagées, en fonction de la structure des données.

2 Approche proposée

La méthode Reduced K-means est une adaptation de la méthode des K-means au contexte du *Subspace clustering*. Elle consiste en la minimisation du critère suivant :

$$\mathcal{C}_{RKM} = ||\mathbf{X} - \mathbf{UFA}^\top||_F^2 \tag{1}$$

où $\mathbf{X}$ de taille $(I \times J)$ correspond à la matrice de données, $\mathbf{U}$ $(I \times c)$ est la matrice d'appartenance des observations aux c classes, composée de 0 et 1, $\mathbf{F}$ $(c \times q)$ est la matrice des centroïdes dans l'espace réduit de dimension q, $\mathbf{A}$ $(J \times q)$ est une matrice de loadings déterminant la contribution de chaque variable à la structure en classe des observations et $||.||_F$ la norme de Frobenius. L'optimisation du critère (1) s'effectue en alternant entre la recherche de la partition $\mathbf{U}$, obtenue par K-means et la mise à jour du sous-espace $\mathbf{A}$, obtenues par décomposition en valeurs singulières. La matrice des centroïdes $\mathbf{F}$ est calculée à partir des matrices $\mathbf{U}$ et $\mathbf{A}$ [10]. Sur la base du critère (1), $\mathbf{X}$ peut être exprimée à travers le modèle à effets fixes suivant :

$$\mathbf{X} = \mathbf{UFA}^\top + \mathbf{E}, \tag{2}$$

où $\mathbf{E}$ est une matrice de bruit. Les lignes et les colonnes de $\mathbf{X}$ sont considérées comme fixes, et l'aléatoire provient uniquement du bruit $\mathbf{E}$.

2.1 Estimation empirique de la distribution des estimateurs

Dans le but d'évaluer la variabilité des paramètres estimés par la méthode RKM, nous utilisons une méthode de bootstrap résiduel basée sur la matrice des résidus estimés $\hat{\mathbf{E}}$ est définie comme :

$$\hat{\mathbf{E}} = \mathbf{X} - \hat{\mathbf{U}}\hat{\mathbf{F}}\hat{\mathbf{A}}^\top,$$

où $\mathbf{X}$ est la matrice de données observées, et $\hat{\mathbf{U}}$, $\hat{\mathbf{F}}$, $\hat{\mathbf{A}}$ sont les paramètres estimés par la méthode RKM. Cette méthode repose sur les étapes suivantes :

1. Tirage avec remise les éléments de la matrice des résidus estimés $\hat{\mathbf{E}}$ pour générer une nouvelle matrice de résidus $\mathbf{E}_b$.
2. Cette nouvelle matrice de résidus $\mathbf{E}_b$ est ajoutée aux paramètres estimés pour construire une nouvelle matrice de données $\mathbf{X}_b$:

$$\mathbf{X}_b = \hat{\mathbf{U}}\hat{\mathbf{F}}\hat{\mathbf{A}}^\top + \mathbf{E}_b.$$

Cette procédure est répétée B fois, produisant ainsi B nouvelles matrices de données notées $(\mathbf{X}_b)_{1\leq b\leq B}$. Ensuite, la méthode RKM est appliquée à chacune de ces matrices pour obtenir leurs partitions respectives et les sous-espaces associés, notés $(\hat{\mathbf{U}}_\mathbf{b}, \hat{\mathbf{F}}_\mathbf{b}, \hat{\mathbf{A}}_\mathbf{b})_{1\leq b\leq B}$. Étant donné que la structure de la matrice de résidus $\hat{\mathbf{E}}$ est inconnue (c'est-à-dire que les bruits peuvent être i.i.d, corrélés, ou présenter des structures de corrélation par classes), il est nécessaire d'envisager différentes méthodologies pour le tirage des résidus. Quatre stratégies sont proposées :

— Tirage avec remise d'un élément e_{ij} de la matrice des résidus $\hat{\mathbf{E}}$, où e_{ij} représente le résidu associé à la i-ème observation et à la j-ème variable :
  a) Sans considération de la structure en classes : Cette méthode consiste à tirer aléatoirement un élément de la matrice des résidus, indépendamment de toute structure de classe ou de corrélation entre les colonnes. **(Bootstrap 1)**
  b) Avec considération de la structure en classes : Dans ce cas, le tirage d'un élément e_{ij} est effectué en tenant compte de la structure en classes définie par $\hat{\mathbf{U}}$. **(Bootstrap 2)**

— Tirage avec remise d'une ligne $e_{i.}$ de la matrice des résidus $\hat{\mathbf{E}}$, où $e_{i.}$ représente les résidus associés à la i-ème observation :
  c) Sans considération de la structure en classes : Cette approche consiste à tirer aléatoirement une ligne de la matrice des résidus, indépendamment de toute structure de classe. **(Bootstrap 3)**
  d) Avec considération de la structure en classes : Ici, le tirage d'une ligne $e_{i.}$ est effectué en respectant de la structure en classes définie par $\hat{\mathbf{U}}$, préservant ainsi les corrélations potentielles entre les variables appartenant à la même classe. **(Bootstrap 4)**

Ces quatre méthodologies permettent d'explorer différentes hypothèses sur la structure des résidus et d'évaluer leur impact sur l'estimation de la variabilité des paramètres $(\hat{\mathbf{U}}_\mathbf{b}, \hat{\mathbf{F}}_\mathbf{b}, \hat{\mathbf{A}}_\mathbf{b})_{1\leq b\leq B}$.

2.2 Quantification de la variabilité des estimateurs

La variabilité des paramètres $(\hat{\mathbf{U}}_b, \hat{\mathbf{F}}_b, \hat{\mathbf{A}}_b)_{1\leq b\leq B}$ est évaluée numériquement à partir des B partitions, centroïdes et sous-espaces générés par la méthode RKM appliquée aux matrices bootstrap $(\mathbf{X}_b)_{1\leq b\leq B}$.

Variabilité de la matrice de partition U : La matrice $\mathbf{U}$ est sujette au phénomène de *label switching*. Pour résoudre ce problème, on aligne les labels de classes estimés par le bootstrap $\hat{\mathbf{U}}_b$ avec ceux d'une partition de référence $\mathbf{R}$ (ici, $\mathbf{R} = \hat{\mathbf{U}}$), en maximisant leur chevauchement [4]. Plus précisément, soit $\mathbf{U}$ une matrice de partition binaire que l'on cherche à aligner avec la matrice de référence $\mathbf{R}$. Soit S_c l'ensemble de toutes les permutations des entiers $1, \ldots, c$. On trouve la permutation $\tau \in S_c$ qui maximise :

$$\tau_b = \arg\max_{\tau\in S_c} \sum_{i=1}^{n}\sum_{k=1}^{c} u_{i\tau(k)} \cdot r_{ik},$$

où :

— $u_{i\tau(k)}$ est l'indicateur d'appartenance de l'observation i à la classe $\tau(k)$ dans la partition après permutation.
— r_{ik} est l'indicateur d'appartenance de l'observation i à la classe k dans la partition de référence.
— $\tau(k)$ est le label de la classe k après application de la permutation τ.

Cette permutation τ_b est ensuite appliquée aux labels de $\hat{\mathbf{U}}_b$ pour aligner les classes avec la référence $\hat{\mathbf{U}}$. La variabilité de $\mathbf{U}$ est calculée par :

$$\text{Var}(\mathbf{U}) = \frac{1}{B-1}\sum_{b=1}^{B} \|\hat{\mathbf{U}}_b - \overline{\mathbf{U}}\|_F^2,$$

où $\overline{\mathbf{U}} = \frac{1}{B}\sum_{b=1}^{B} \hat{\mathbf{U}}_b$ est la moyenne des matrices $\hat{\mathbf{U}}_b$.

Variabilité de la matrice des coefficients A : La matrice $\mathbf{A}$ est estimée à une rotation près par le RKM, une rotation de Procrustes est donc appliquée pour aligner $\hat{\mathbf{A}}_b$ avec une matrice de référence (ici $\hat{\mathbf{A}}$). La variabilité de $\mathbf{A}$ est calculée par :

$$\text{Var}(\mathbf{A}) = \frac{1}{B-1}\sum_{b=1}^{B} \|\hat{\mathbf{A}}_b - \overline{\mathbf{A}}\|_F^2,$$

où $\overline{\mathbf{A}} = \frac{1}{B}\sum_{b=1}^{B} \hat{\mathbf{A}}_b$ est la moyenne des matrices $\hat{\mathbf{A}}_b$.

Variabilité de la matrice des centroïdes F : Étant donnée que la matrice $\mathbf{F}$ est déduite de la matrice $\mathbf{U}$ et $\mathbf{A}$. Les matrices $\hat{\mathbf{F}}_b$ sont calculées à partir des matrices $\hat{\mathbf{A}}_b$ auxquelles la rotation de Procrustes a été appliquée et $\hat{\mathbf{U}}_b$ auxquelles la permutation τ_b a été appliquée.

$$\hat{\mathbf{F}}_b = (\hat{\mathbf{U}}_b^\top \hat{\mathbf{U}}_b)^{-1}\hat{\mathbf{U}}_b^\top \mathbf{X}_b \hat{\mathbf{A}}_b$$

La variabilité de $\mathbf{F}$ est calculée par :

$$\text{Var}(\mathbf{F}) = \frac{1}{B-1}\sum_{b=1}^{B} \|\hat{\mathbf{F}}_b - \overline{\mathbf{F}}\|_F^2,$$

où $\overline{\mathbf{F}} = \frac{1}{B}\sum_{b=1}^{B} \hat{\mathbf{F}}_b$ est la moyenne des matrices $\hat{\mathbf{F}}_b$.

Variabilité de la globalité des paramètres UFA : La variabilité globale est calculée de la sorte :

$$\text{Var}(\mathbf{UFA}) = \frac{1}{B-1}\sum_{b=1}^{B} \|\hat{\mathbf{U}}_b\hat{\mathbf{F}}_b\hat{\mathbf{A}}_b^\top - \overline{\mathbf{UFA}}\|_F^2,$$

où $\overline{\mathbf{UFA}} = \frac{1}{B}\sum_{b=1}^{B} \hat{\mathbf{U}}_b\hat{\mathbf{F}}_b\hat{\mathbf{A}}_b^\top$ est la moyenne des reconstructions.

3 Évaluation

3.1 Plan de simulation

Le but de cette simulation est d'identifier la stratégie de bootstrap la plus adaptée à chaque modèle de résidu. Pour cela, nous nous basons sur le modèle du RKM (2). Dans un premier temps, la matrice $\mathbf{UFA}$ est générée de manière à assurer une bonne séparation et un équilibre entre les classes, avec $c = 3$ et $q = 2$. Les scénarios sont définis selon la structure fixe établie précédemment, en variant la matrice de résidus $\mathbf{E}$. Quatre types de matrices de résidus ont été considérés :

1. **Bruit homoscédastique non corrélé** : Génère un bruit indépendant pour chaque variable, où chaque élément de la matrice $\mathbf{E}$ suit une distribution normale centrée réduite $\mathcal{N}(0, 1)$.
2. **Bruit homoscédastique corrélé** : Génère un bruit corrélé entre les variables, où chaque ligne de $\mathbf{E}$ est un échantillon tiré d'une distribution normale multivariée avec une moyenne nulle et une matrice de covariance Σ_p

$$\Sigma_p = \begin{pmatrix} 1 & 0.7 & 0.7 & 0.7 \\ 0.7 & 1 & 0.7 & 0.7 \\ 0.7 & 0.7 & 1 & 0.7 \\ 0.7 & 0.7 & 0.7 & 1 \end{pmatrix}$$

3. **Bruit hétéroscédastique non corrélé** : Génère un bruit spécifique à chaque classe, où chacun des individus d'une classe est tiré d'une distribution normale centrée et de variances différentes, définies pour chaque classe par $(\sigma_1^2 = 1, \sigma_2^2 = 2, \sigma_3^2 = 3)$.
4. **Bruit hétéroscédastique corrélé** : Génère un bruit avec des variables corrélées, spécifique à chaque classe. Chaque individu d'une classe est tiré d'une distribution normale multivariée avec une moyenne nulle et une matrice de covariance Σ_p, définie comme suit :

$$\Sigma_p = \begin{cases} \Sigma_{p1} = \sigma_1^2 \mathbf{I}_4 + \rho_1(\mathbf{J}_4 - \mathbf{I}_4) & \text{pour la classe 1,} \\ \Sigma_{p2} = \sigma_2^2 \mathbf{I}_4 + \rho_2(\mathbf{J}_4 - \mathbf{I}_4) & \text{pour la classe 2,} \\ \Sigma_{p3} = \sigma_3^2 \mathbf{I}_4 + \rho_3(\mathbf{J}_4 - \mathbf{I}_4) & \text{pour la classe 3,} \end{cases}$$

où :
— $\sigma_1^2 = 1$, $\sigma_2^2 = 2$, $\sigma_3^2 = 3$ sont les variances,
— $\rho_1 = 0.7$, $\rho_2 = 0.8$, $\rho_3 = 0.9$ sont les coefficients de corrélation,
— $\mathbf{I}_4$ est la matrice identité de taille 4×4,
— $\mathbf{J}_4$ est la matrice de uns de taille 4×4.

L'article de Morris et al.[8] propose des mesures pour estimer la variabilité des paramètres dans des études de simulation. Dans notre étude, pour chaque scénario, 100 jeux de données ont été générés ($n = 100$), et pour chaque jeu de données, 50 matrices de résidus ont été tirées par bootstrap ($B = 50$). Pour évaluer la stabilité de l'estimation d'un paramètre $\hat{\theta}_{\mathbf{b}}$, nous nous basons sur les deux indicateurs suivants, où $\hat{\theta}$ peut représenter $\mathbf{U}$, $\mathbf{F}$, $\mathbf{A}$, ou $\mathbf{UFA}$:

— **Écart-type estimé par le bootstrap (BootSE)** :

$$\text{BootSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{I}\left(\frac{1}{B-1}\sum_{b=1}^{B}\|\hat{\theta}_{\mathbf{b}} - \overline{\theta_b}\|_F^2\right)}$$

où $\overline{\theta_b} = \frac{1}{B}\sum_{b=1}^{B}\hat{\theta}_{\mathbf{b}}$ représente la moyenne des $\hat{\theta}_{\mathbf{b}}$.

— **Erreur standard de Monte Carlo pour BootSE (MCSE)** :

$$\text{Var}(\theta)_i = \frac{1}{B-1}\sum_{b=1}^{B}\|\hat{\theta}_b - \overline{\theta_b}\|_F^2$$

$$\text{MCSE}_{\text{BootSE}} = \sqrt{\frac{\frac{1}{n-1}\sum_{i=1}^{n}(\text{Var}(\theta)_i - \overline{\text{Var}(\theta)})^2}{4n \times \text{BootSE}^2}}$$

Dans un contexte de simulation, il est attendu que la variabilité estimée par la méthode de bootstrap soit comparable à la variabilité observée lors de différentes répétitions de la méthode RKM. Pour ce fait, nous introduisons deux mesures supplémentaires : l'écart-type empirique (EmpSE) et son erreur standard de Monte Carlo associée (MCSE).

— **Écart-type empirique :**

$$\text{EmpSE} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\|\hat{\theta} - \overline{\theta}\|_F^2}$$

— **Monte Carlo Standard Error pour EmpSE :**

$$\text{MCSE}_{\text{EmpSE}} = \frac{\text{EmpSE}}{\sqrt{2(n-1)}}$$

3.2 Résultats

Pour les expérimentations, nous avons fait varier le nombre d'individus $I \in \{300, 1500\}$ afin d'analyser l'impact du nombre d'individus sur l'estimation de la variabilité des paramètres. De même, la configuration du bruit présentée précédemment a été réduite afin d'évaluer l'influence du bruits sur l'estimation de la variabilité des paramètres. L'intégralité des résultats est présentée sous forme de tableaux en annexe (Table 3,4,1,2).

Pour $I = 300$: On constate que pour le scénario où le bruit est homoscédastique et non corrélé, toutes les stratégies de bootstrap permettent d'estimer correctement l'écart type empirique des paramètres. Pour les trois autres scénarios, où le bruit est soit homoscédastique et corrélé, soit hétéroscédastique (corrélé ou non), l'écart type est sous-estimé lorsque le bruit est fort et surestimé lorsqu'il est faible. Toutefois, la stratégie de bootstrap qui fonctionne le mieux est généralement celle qui correspond à la nature du bruit, sauf dans le cas du scénario avec un bruit hétéroscédastique fort.

Pour $I = 1500$: Lorsque l'on augmente le nombre d'individus, pour le scénario où le bruit est homoscédastique et non corrélé, on observe qu'avec un bruit fort, l'écart type des paramètres est correctement estimé avec les différentes stratégies de bootstrap. Cependant, lorsque le bruit diminue, on constate une sous-estimation de cet écart type avec toutes les stratégies, ce qui est dû à une sous-estimation de la variabilité du sous-espace et de la matrice des centroïdes qui en dépend.

Pour le scénario avec un bruit homoscédastique et corrélé, on observe que toutes les stratégies de bootstrap permettent d'estimer correctement la variabilité des paramètres, avec une meilleure performance pour la quatrième stratégie, qui prend en compte la corrélation entre les variables par classe, ce qui est attendu.

Pour les deux autres scénarios, où le bruit est hétéroscédastique (corrélé ou non), on retrouve des résultats similaires à celles observées avec 300 individus : l'écart type est sous-estimé lorsque le bruit est fort et surestimé lorsqu'il est faible. Les stratégies de bootstrap les plus adaptées à la nature du bruit fournissent les meilleurs résultats, à l'exception du cas où le bruit est hétéroscédastique et corrélé avec un niveau de bruit élevé, où des écarts persistent.

4 Conclusion et perspectives

Ce travail porte sur l'évaluation numérique de la sensibilité de la partition et du sous espace via-à-vis des fluctuations d'échantillonage. Pour cela, nous avons proposé une façon de simuler ces fluctuations via différentes approches de bootstrap. Ensuite, nous avons proposé une façon de quantifier cette sensibilité.

Les résultats montrent que, pour un bruit homoscédastique non corrélé, toutes les stratégies de bootstrap permettent

de bien estimer l'écart type des paramètres. En revanche, pour les autres scénarios (bruit homoscédastique corrélé ou hétéroscédastique), l'écart type est sous-estimé lorsque le bruit est fort et surestimé lorsqu'il est faible. La stratégie de bootstrap la plus performante est généralement celle qui correspond à la nature du bruit, sauf dans le cas d'un bruit hétéroscédastique de forte intensité.
Cette étude pourrait être directement étendue à d'autres méthodes reposant sur des modèles à effets fixes, telles que le semi-NMF PCA [1] ou le FKM [11]. Une autre perspective intéressante serait d'adapter cette approche à des modèles de réduction de dimension non linéaires. Enfin, l'application du bootstrap résiduel en présence de données manquantes permettrait d'évaluer sa stabilité dans des contextes où l'information est partiellement observée.

Annexe

Structure fixe du plan de simulation : En se plaçant dans le cadre d'un modèle RKM (2). Ceci implique de définir les trois matrices $\mathbf{U}$, $\mathbf{F}$, $\mathbf{A}$. Pour cela, nous nous sommes inspirés du plan de simulation de Terada [10]. Pour une matrice $\mathbf{X}$ de dimensions $(I \times J)$, avec p_1 le nombre des variables informatives, p_2 le nombre des variables de bruit $(J = p1 + p2)$. Dans ce plan, la matrice $\mathbf{U}$ est générée grâce à une loi multinomiale avec des probabilités égales. La matrice de centroïdes $\mathbf{F}$ est générée à partir d'une distribution uniforme q-dimensionnelle sur $[-20, 20]^q$. $\mathbf{A}$ est ensuite construite de la sorte :

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}^{*T} & 0^T_{q\times(p2)} \end{bmatrix}$$

avec $\mathbf{A}^*$ une matrice orthogonale de dimension $p1 \times q$ générée de manière aléatoire.

Références

[1] Kais Allab, Lazhar Labiod, and Mohamed Nadif. A semi-nmf-pca unified framework for data clustering. *IEEE Transactions on Knowledge and Data Engineering*, 29(1) :2–16, 2016.

[2] Geert De Soete and J Douglas Carroll. K-means clustering in a low-dimensional euclidean space. In *New approaches in classification and data analysis*, pages 212–219. Springer, 1994.

[3] Sara Dolnicar and Friedrich Leisch. Segmenting markets by bagged clustering. *Australasian Marketing Journal (AMJ)*, 12(1) :51–65, 2004.

[4] Sandrine Dudoit and Jane Fridlyand. Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, 19(9) :1090–1099, 2003.

[5] Joeri Hofmans, Eva Ceulemans, Douglas Steinley, and Iven Van Mechelen. On the added value of bootstrap analysis for k-means clustering. *Journal of classification*, 32(2) :268–284, 2015.

[6] Julie Josse and François Husson. Handling missing values in exploratory multivariate data analysis methods. *Journal de la société française de statistique*, 153(2) :79–99, 2012.

[7] Julie Josse, Stefan Wager, and François Husson. Confidence areas for fixed-effects pca. *Journal of Computational and Graphical Statistics*, 25(1) :28–48, 2016.

[8] Tim P Morris, Ian R White, and Michael J Crowther. Using simulation studies to evaluate statistical methods. *Statistics in medicine*, 38(11) :2074–2102, 2019.

[9] Lance Parsons, Ehtesham Haque, and Huan Liu. Subspace clustering for high dimensional data : a review. *Acm sigkdd explorations newsletter*, 6(1) :90–105, 2004.

[10] Yoshikazu Terada. Strong consistency of reduced k-means clustering. *Scandinavian Journal of Statistics*, 41(4) :913–931, 2014.

[11] Maurizio Vichi and Henk AL Kiers. Factorial k-means analysis for two-way data. *Computational Statistics & Data Analysis*, 37(1) :49–64, 2001.

	Bootstrap 1 (MCSE)		Bootstrap 2 (MCSE)		Bootstrap 3 (MCSE)		Bootstrap 4 (MCSE)		Écart type empirique (MCSE)	
Scenario	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**
Bruit homoscédastique non corrélé	UFA : 3.15 (0.16) U : 0.04 (0.07) F : 0.22 (0.06) A : 0.02 (0.07)	UFA : 0.79 (0.02) U : 0.00 (0.00) F : 0.06 (0.07) A : 0.01 (0.07)	UFA : 3.13 (0.15) U : 0.00 (0.00) F : 0.22 (0.06) A : 0.02 (0.07)	UFA : 0.79 (0.02) U : 0.00 (0.00) F : 0.06 (0.07) A : 0.01 (0.07)	UFA : 3.12 (0.15) U : 0.00 (0.00) F : 0.22 (0.06) A : 0.02 (0.07)	UFA : 0.80 (0.02) U : 0.00 (0.00) F : 0.06 (0.07) A : 0.01 (0.07)	UFA : 3.12 (0.15) U : 0.00 (0.00) F : 0.22 (0.02) A : 0.02 (0.07)	UFA : 0.80 (0.02) U : 0.00 (0.00) F : 0.06 (0.07) A : 0.01 (0.07)	UFA : 3.05 (0.22) U : 0.00 (0.00) F : 0.22 (0.02) A : 0.02 (0.00)	UFA : 0.80 (0.08) U : 0.00 (0.00) F : 0.06 (0.01) A : 0.01 (0.00)
Bruit homoscédastique corrélé	UFA : 3.14 (0.015) U : 0.00 (0.00) F : 0.22 (0.06) A : 0.02 (0.07)	UFA : 1.89 (0.07) U : 0.00 (0.00) F : 0.12 (0.06) A : 0.01 (0.07)	UFA : 3.22 (0.18) U : 0.11 (0.08) F : 0.23 (0.05) A : 0.02 (0.07)	UFA : 1.83 (0.07) U : 0.00 (0.00) F : 0.12 (0.06) A : 0.01 (0.07)	UFA : 3.15 (0.16) U : 0.05 (0.07) F : 0.22 (0.06) A : 0.02 (0.07)	UFA : 1.90 (0.07) U : 0.00 (0.00) F : 0.12 (0.06) A : 0.01 (0.07)	UFA : 3.17 (0.16) U : 0.07 (0.07) F : 0.23 (0.05) A : 0.02 (0.07)	UFA : 1.84 (0.07) U : 0.00 (0.00) F : 0.12 (0.06) A : 0.01 (0.07)	UFA : 3.49 (0.35) U : 0.20 (0.02) F : 0.23 (0.02) A : 0.02 (0.00)	UFA : 1.61 (0.16) U : 0.00 (0.00) F : 0.11 (0.01) A : 0.01 (0.00)
Bruit hétéroscédastique non corrélé	UFA : 6.11 (0.38) U : 0.62 (0.04) F : 0.31 (0.05) A : 0.03 (0.07)	UFA : 2.59 (0.12) U : 0.00 (0.00) F : 0.17 (0.06) A : 0.02 (0.07)	UFA : 5.55 (0.42) U : 0.46 (0.07) F : 0.32 (0.05) A : 0.03 (0.07)	UFA : 2.56 (0.11) U : 0.00 (0.00) F : 0.16 (0.06) A : 0.02 (0.07)	UFA : 6.55 (0.44) U : 0.73 (0.04) F : 0.31 (0.05) A : 0.03 (0.07)	UFA : 2.51 (0.11) U : 0.00 (0.00) F : 0.16 (0.06) A : 0.02 (0.07)	UFA : 4.85 (0.31) U : 0.33 (0.06) F : 0.31 (0.05) A : 0.03 (0.07)	UFA : 2.52 (0.11) U : 0.00 (0.00) F : 0.16 (0.06) A : 0.02 (0.07)	UFA : 7.92 (0.80) U : 0.98 (0.10) F : 0.31 (0.03) A : 0.03 (0.00)	UFA : 2.17 (0.22) U : 0.00 (0.00) F : 0.16 (0.02) A : 0.01 (0.00)
Bruit hétéroscédastique corrélé	UFA : 5.96 (0.37) U : 0.58 (0.04) F : 0.32 (0.05) A : 0.03 (0.07)	UFA : 2.58 (0.11) U : 0.00 (0.00) F : 0.16 (0.06) A : 0.02 (0.07)	UFA : 7.17 (0.53) U : 0.83 (0.06) F : 0.33 (0.05) A : 0.03 (0.07)	UFA : 2.56 (0.11) U : 0.00 (0.00) F : 0.17 (0.06) A : 0.02 (0.07)	UFA : 6.30 (0.42) U : 0.69 (0.05) F : 0.31 (0.05) A : 0.03 (0.07)	UFA : 2.51 (0.11) U : 0.00 (0.00) F : 0.16 (0.06) A : 0.02 (0.07)	UFA : 5.02 (0.33) U : 0.36 (0.07) F : 0.33 (0.05) A : 0.03 (0.07)	UFA : 2.47 (0.11) U : 0.00 (0.00) F : 0.17 (0.06) A : 0.02 (0.07)	UFA : 9.44 (0.95) U : 1.25 (0.13) F : 0.30 (0.03) A : 0.03 (0.00)	UFA : 2.12 (0.21) U : 0.00 (0.00) F : 0.15 (0.02) A : 0.02 (0.00)

TABLE 1 – Écarts-types estimés par bootstrap ($B = 50$) et $n = 100$, ainsi que l'erreur de Monte Carlo associée, les écarts-types empiriques et leur erreur de Monte Carlo correspondante. $I = 300$

	Bootstrap 1 (MCSE)		Bootstrap 2 (MCSE)		Bootstrap 3 (MCSE)		Bootstrap 4 (MCSE)		Écart type empirique (MCSE)	
Scenario	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**
Bruit homoscédastique non corrélé	UFA : 3.28 (0.17) U : 0.04 (0.07) F : 0.10 (0.06) A : 0.01 (0.07)	UFA : 0.70 (0.03) U : 0.00 (0.00) F : 0.02 (0.07) A : 0.00 (0.07)	UFA : 3.30 (0.17) U : 0.00 (0.00) F : 0.10 (0.06) A : 0.01 (0.07)	UFA : 0.71 (0.03) U : 0.00 (0.00) F : 0.02 (0.07) A : 0.00 (0.07)	UFA : 3.33 (0.17) U : 0.05 (0.07) F : 0.10 (0.06) A : 0.01 (0.07)	UFA : 0.71 (0.03) U : 0.00 (0.00) F : 0.02 (0.07) A : 0.00 (0.07)	UFA : 3.47 (0.18) U : 0.00 (0.00) F : 0.10 (0.06) A : 0.01 (0.07)	UFA : 0.71 (0.03) U : 0.00 (0.00) F : 0.03 (0.07) A : 0.00 (0.07)	UFA : 3.15 (0.32) U : 0.00 (0.00) F : 0.10 (0.01) A : 0.01 (0.00)	UFA : 0.82 (0.08) U : 0.00 (0.00) F : 0.02 (0.00) A : 0.00 (0.00)
Bruit homoscédastique corrélé	UFA : 3.32 (0.17) U : 0.05 (0.07) F : 0.10 (0.06) A : 0.01 (0.07)	UFA : 1.55 (0.04) U : 0.00 (0.00) F : 0.05 (0.07) A : 0.01 (0.07)	UFA : 3.79 (0.26) U : 0.27 (0.07) F : 0.11 (0.06) A : 0.01 (0.07)	UFA : 1.55 (0.04) U : 0.00 (0.00) F : 0.05 (0.07) A : 0.01 (0.07)	UFA : 3.40 (0.18) U : 0.08 (0.08) F : 0.10 (0.06) A : 0.01 (0.07)	UFA : 1.56 (0.04) U : 0.00 (0.00) F : 0.05 (0.07) A : 0.01 (0.07)	UFA : 3.64 (0.21) U : 0.16 (0.07) F : 0.11 (0.06) A : 0.01 (0.07)	UFA : 1.56 (0.04) U : 0.00 (0.00) F : 0.05 (0.07) A : 0.01 (0.07)	UFA : 3.63 (0.37) U : 0.28 (0.03) F : 0.10 (0.01) A : 0.01 (0.00)	UFA : 1.59 (0.16) U : 0.00 (0.00) F : 0.05 (0.01) A : 0.01 (0.00)
Bruit hétéroscédastique non corrélé	UFA : 9.93 (0.66) U : 1.29 (0.03) F : 0.14 (0.06) A : 0.01 (0.07)	UFA : 2.42 (0.12) U : 0.00 (0.00) F : 0.07 (0.07) A : 0.01 (0.07)	UFA : 8.99 (0.70) U : 1.12 (0.07) F : 0.14 (0.06) A : 0.01 (0.07)	UFA : 2.44 (0.12) U : 0.00 (0.00) F : 0.07 (0.07) A : 0.01 (0.07)	UFA : 11.61 (0.79) U : 1.57 (0.05) F : 0.14 (0.06) A : 0.01 (0.07)	UFA : 2.36 (0.11) U : 0.00 (0.00) F : 0.07 (0.07) A : 0.01 (0.07)	UFA : 5.63 (0.40) U : 0.46 (0.07) F : 0.14 (0.06) A : 0.01 (0.07)	UFA : 2.41 (0.12) U : 0.00 (0.00) F : 0.07 (0.07) A : 0.01 (0.07)	UFA : 12.03 (1.22) U : 1.64 (0.17) F : 0.14 (0.01) A : 0.01 (0.00)	UFA : 2.14 (0.22) U : 0.00 (0.00) F : 0.07 (0.01) A : 0.01 (0.00)
Bruit hétéroscédastique corrélé	UFA : 9.84 (0.65) U : 1.27 (0.03) F : 0.14 (0.06) A : 0.01 (0.07)	UFA : 2.41 (0.11) U : 0.00 (0.00) F : 0.07 (0.07) A : 0.01 (0.07)	UFA : 13.08 (0.93) U : 1.79 (0.08) F : 0.15 (0.06) A : 0.01 (0.07)	UFA : 2.37 (0.11) U : 0.00 (0.00) F : 0.07 (0.07) A : 0.01 (0.07)	UFA : 11.42 (0.78) U : 1.53 (0.05) F : 0.14 (0.06) A : 0.01 (0.07)	UFA : 2.32 (0.11) U : 0.00 (0.00) F : 0.07 (0.07) A : 0.01 (0.07)	UFA : 6.58 (0.48) U : 0.68 (0.06) F : 0.15 (0.06) A : 0.01 (0.07)	UFA : 2.27 (0.10) U : 0.00 (0.00) F : 0.07 (0.07) A : 0.01 (0.07)	UFA : 18.26 (1.30) U : 2.59 (0.18) F : 0.14 (0.01) A : 0.01 (0.00)	UFA : 2.09 (0.21) U : 0.00 (0.00) F : 0.07 (0.01) A : 0.01 (0.00)

TABLE 2 – Écarts-types estimés par bootstrap ($B = 50$) et $n = 100$, ainsi que l'erreur de Monte Carlo associée, les écarts-types empiriques et leur erreur de Monte Carlo correspondante. $I = 1500$

	Biais bootstrap 1		Biais bootstrap 2		Biais bootstrap 3		Biais bootstrap 4		Écart type empirique (MCSE)	
Scenario	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**
Bruit homoscédastique non corrélé	UFA : -3.28%	UFA : 1.25%	UFA : -2.62%	UFA : 1.25%	UFA : -2.30%	UFA : 0.00%	UFA : -2.30%	UFA : 0.00%	UFA : 3.05 (0.22)	UFA : 0.80 (0.08)
Bruit homoscédastique corrélé	UFA : 10.03%	UFA : -17.39%	UFA : 7.74%	UFA : -13.66%	UFA : 9.74%	UFA : -18.01%	UFA : 9.17%	UFA : -14.29%	UFA : 3.49 (0.35)	UFA : 1.61 (0.16)
Bruit hétéroscédastique non corrélé	UFA : 22.85%	UFA : -19.35%	UFA : 29.92%	UFA : -17.97%	UFA : 17.30%	UFA : -15.67%	UFA : 38.76%	UFA : -16.13%	UFA : 7.92 (0.80)	UFA : 2.17 (0.22)
Bruit hétéroscédastique corrélé	UFA : 36.86%	UFA : -21.70%	UFA : 24.05%	UFA : -20.75%	UFA : 33.26%	UFA : -18.40%	UFA : 46.82%	UFA : -16.51%	UFA : 9.44 (0.95)	UFA : 2.12 (0.21)

TABLE 3 – Biais relatif des estimations des écarts-type du paramètre **UFA** avec ($I = 300$), la dernière colonne correspond à l'écart-type empirique du paramètre **UFA** et son erreur de Monte Carlo associée.

	Biais bootstrap 1		Biais bootstrap 2		Biais bootstrap 3		Biais bootstrap 4		Écart type empirique (MCSE)	
Scenario	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**	**Bruit fort**	**Bruit faible**
Bruit homoscédastique non corrélé	UFA : -4.13%	UFA : 14.63%	UFA : -4.76%	UFA : 13.41%	UFA : -5.71%	UFA : 13.41%	UFA : -1.16%	UFA : 13.41%	UFA : 3.15 (0.32)	UFA : 0.82 (0.08)
Bruit homoscédastique corrélé	UFA : 8.54%	UFA : 2.52%	UFA : -4.41%	UFA :2.52%	UFA : -6.34%	UFA : 1.89%	UFA : -0.28%	UFA : 0.89%	UFA : 3.63 (0.37)	UFA : 1.59 (0.16)
Bruit hétéroscédastique non corrélé	UFA : 17.46%	UFA : -13.08%	UFA : 25.27%	UFA : -14.02%	UFA : 3.49%	UFA : -10.28%	UFA : 53.20%	UFA : -12.62%	UFA : 12.03 (1.22)	UFA : 2.14 (0.22)
Bruit hétéroscédastique corrélé	UFA : 46.11%	UFA : -15.31%	UFA : 28.37%	UFA : -13.40%	UFA : 37.46%	UFA : -11.00%	UFA : 63.96%	UFA : -8.61%	UFA : 18.26 (1.30)	UFA : 2.09 (0.21)

TABLE 4 – Biais relatif des estimations des écarts-type du paramètre **UFA** avec ($I = 1500$), la dernière colonne correspond à l'écart-type empirique du paramètre **UFA** et son erreur de Monte Carlo associée.

Le critère d'Apresjan en classification hiérarchique ascendante

Patrice Bertrand[1], Jean Diatta[2]

[1] Université Dauphine-PSL, Ceremade
[2] Université de la Réunion, LIM-EA2525

patrice.bertrand@ceremade.dauphine.fr, jean.diatta@univ-reunion.fr

Résumé

Les classes d'Apresjan sont constituées d'objets qui sont plus proches entre eux qu'avec les objets extérieurs à la classe. Nous déterminons les liens d'agrégation pour lesquels chaque hiérarchie générée par l'algorithme de la CAH contient toutes les classes d'Apresjan. Dans la plupart des cas, les classes d'Apresjan sont peu nombreuses et souvent de petite taille. Aussi, nous proposons une relaxation du critère les définissant, ce qui permet de construire une suite croissante (par inclusion) de hiérarchies allant de la hiérarchie d'Apresjan à toute hiérarchie la contenant.

Mots-clés

Hiérarchie d'Apresjan, algorithme de la CAH, Convexité d'intervalle.

Abstract

Apresjan clusters are made up of objects that are more similar with each other than they are with objects outside the clusters. We first determine the usual aggregation links for which each hierarchy generated by the AHC algorithm contains all Apresjan clusters. In most cases, Apresjan clusters are not very numerous and are often small in size. We then propose a relaxation of the criterion defining them, which makes it possible to construct an increasing sequence (by inclusion) of hierarchies from the Apresjan hierarchy to any hierarchy containing it.

Keywords

Apresjan hierarchy, AHC algorithm, Interval convexity.

1 Introduction

La classification hiérarchique révèle la structure cachée en classes d'un ensemble selon différents niveaux de granularité. Si S désigne l'ensemble des objets à classifier, une méthode de classification ascendante hiérarchique génère une hiérarchie $\mathcal{H}$ définie sur S, c'est-à-dire une collection $\mathcal{H}$ de parties de S, qui forme une suite de partitions de moins en moins fines, allant de l'ensemble des singletons de S au singleton $\{S\}$. Les différents niveaux de granularité de la hiérarchie $\mathcal{H}$ sont valués par une fonction d'indiçage $f : \mathcal{H} \mapsto \mathbb{R}^+$ strictement monotone au sens suivant : pour tout $A, B \in \mathcal{H}$, $A \subset B$ implique $f(A) < f(B)$. Le couple $(\mathcal{H}, f)$ est alors appelé hiérarchie indicée. La classification ascendante hiérarchique (CAH) est largement utilisée par les statisticiens et praticiens de divers domaines d'application. Rappelons que la CAH requiert un lien d'agrégation, i.e. une application δ qui évalue le degré de proximité, noté $\delta(A, B)$, entre deux parties disjointes A et B de S. L'algorithme de la CAH procède récursivement comme suit : chaque itération génère une classe de la forme $A \cup B$ où $\delta(A, B)$ est minimum dans l'ensemble des valeurs $\delta(X, Y)$ où X et Y désignent des classes maximales au sens de l'inclusion parmi les classes déjà créées.

Dans ce texte, nous nous intéressons à un certain type de classes introduites indépendamment par A.F. Parker Rhodes and R.M. Needham [13] et par J.D. Apresjan [3] afin de définir des classes pertinentes au sens d'une dissimilarité donnée. Plus précisément, ces classes appelées *classes d'Apresjan*, sont les parties A de S telles que la dissimilarité entre deux éléments quelconques x et y de A, est nécessairement inférieure à la dissimilarité entre x et z, où z désigne un élément arbitraire de $S \setminus A$. L'ensemble de ces classes, définies au sens d'une dissimilarité d, forme une hiérarchie appelée *hiérarchie d'Apresjan* de d. L'intérêt pour ces caractéristiques intéressantes des classes d'Apresjan, est toutefois tempéré par le fait que le critère définissant les classes d'Apresjan s'avère strict pour la plupart des dissimilarités, et qu'en conséquence, la hiérarchie d'Apresjan contient généralement peu de classes non triviales (i.e. non réduites à un singleton et différentes de S) et que la taille de ces classes est souvent petite. Plusieurs auteurs ont ainsi considéré des assouplissements du critère définissant les classes d'Apresjan (e.g. [8, 11]). Dans ce qui suit, nous établissons les résultats suivants :

- une caractérisation des sous-ensembles appartenant à toute hiérarchie générée par la CAH pour un lien d'agrégation donné ;
- une condition suffisante vérifiée par un lien d'agrégation pour que la CAH génère une hiérarchie contenant la hiérarchie d'Apresjan, pour toute dissimilarité utilisée ;
- la liste des liens parmi les liens d'agrégation standard pour lesquels chaque hiérarchie générée par la CAH inclut la hiérarchie d'Apresjan ;
- une relaxation croissante du critère définissant les classes d'Apresjan. Il en résulte une suite de hiérarchies emboîtées allant de la hiérarchie d'Apresjan à toute hiérarchie la contenant.

1.1 Travaux antérieurs

Comme souligné par Carlsson et Mémoli [10], les fondations théoriques des méthodes de clustering restent, malgré plusieurs avancées, insuffisantes. Pour la méthode de la CAH, des résultats théoriques récents ont notamment concerné le passage à l'échelle, e.g. [12, 2, 14]. Mentionnons aussi une meilleure compréhension de l'utilisation des liens d'agrégation, qui a été proposée par Ackerman *et al.* [1].
En ce qui concerne les classes d'Apresjan, elles ont été souvent re-découvertes indépendamment dans divers contextes théoriques. Dans la plupart des cas, les auteurs soulignent leur intérêt en tant que classes idéales. Les classes d'Apresjan ont été appelées K-*groups* par Bock [7], et *classes fortes* par Bandelt et Dress [4] et par Bryant et Berry [8]. La hiérarchie d'Apresjan a été appelée *nice clustering* par Ackerman *et al.* [1].

2 Liens pour lesquels la CAH inclut la hiérarchie d'Apresjan

Dans ce qui suit, d désigne une dissimilarité définie sur un ensemble S fini, i.e. d est une application de $S \times S$ dans $\mathbb{R}^+$ telle que :

$$d(x, y) = d(y, x) \geq d(x, x) = 0,$$

pour tout $x, y \in S$. L'ensemble des parties de S est noté $\mathcal{P}(S)$ et δ désigne un lien d'agrégation défini sur S, i.e. une application qui est définie sur le sous-ensemble de $\mathcal{P}(S) \times \mathcal{P}(S)$ contenant tous les couples de parties disjointes de S, et qui prend ses valeurs dans $\mathbb{R}^+$. Par la suite, nous supposons que le lien δ est défini à partir de d et vérifie :

$$\delta(\{x\}, \{y\}) = d(x, y),$$

pour toute paire d'éléments distincts x, y de S.

Notation 2.1 *On dira qu'une hiérarchie est une δ-hiérarchie si elle a été générée par une exécution (arbitraire) de l'algorithme CAH utilisant le lien δ.*
Si $\mathscr{C}$ est une δ-hiérarchie, nous noterons :

- *$\mathscr{C}_t$ l'ensemble des classes de $\mathscr{C}$ créées jusqu'à l'étape t de la CAH,*
- *$\mathscr{C}_{t,\max}$ l'ensemble des classes de $\mathscr{C}_t$ maximales au sens de l'ordre d'inclusion définie sur $\mathscr{C}_t$.*

Definition 2.2 Une partie C non vide de S est dite *δ-fermée minimalement* si pour tout $t \geq 0$ et toute paire $X, Y \in \mathscr{C}_{t,\max}$ qui minimise δ à l'étape t, on a :

$$X \subsetneq C \Rightarrow Y \subsetneq C.$$

Theorem 2.3 *Une partie non vide de S appartient à chaque δ-hiérarchie si et seulement elle est δ-fermée minimalement.*

Definition 2.4 Une partie non vide A de S est appelée *classe d'Apresjan de d* (cf. J. Apresjan [3]) si :

$$d(x, y) < \min\{d(x, z), d(y, z)\},$$

pour tout $x \in A, y \in A$ et $z \in (S \setminus A)$. Une partie non vide A de S sera dite *δ-Apresjan*, si :

$$\delta(X, Y) < \min\{\delta(X, Z), \delta(Y, Z)\},$$

pour toutes les parties non vides $X, Y, Z \in \mathcal{P}(S)$ telles que $X \in \mathcal{P}(A), Y \in \mathcal{P}(A \setminus X)$ et $Z \in \mathcal{P}(S \setminus A)$.

L'ensemble des classes d'Apresjan de d, forme une hiérarchie appelée *hiérarchie d'Apresjan* de d, et notée $\mathcal{H}_{\mathcal{A}}(d)$ par la suite.
Par définition, l'ensemble des parties δ-Apresjan forme un sous-ensemble de $\mathcal{H}_{\mathcal{A}}(d)$. Comme les parties triviales de S (i.e. S et ses singletons) sont δ-Apresjan, il en résulte que les parties δ-Apresjan forment une hiérarchie sur S, qui est contenue dans $\mathcal{H}_{\mathcal{A}}(d)$.

Proposition 2.5 *Toute partie δ-Apresjan est δ-fermée minimalement.*

On rappelle qu'un lien d'agrégation δ satisfait la *condition de la médiane* si pour tout $X, Y, Z \in \mathcal{P}(S)$, on a :

$$\delta(X, Y) \leq \min\{\delta(X, Z), \delta(Y, Z)\} \Rightarrow \delta(X \cup Y, Z) \geq \min\{\delta(X, Z), \delta(Y, Z)\}.$$

Proposition 2.6 *La hiérarchie d'Apresjan de d est incluse dans chaque hiérarchie $\mathcal{H}$ construite par la CAH avec le lien δ si chaque classe d'Apresjan de d est δ-Apresjan. En outre, si δ satisfait la condition de la médiane, alors aucune classe d'Apresjan n'est supprimée lors de l'élagage de $\mathcal{H}$ effectué afin que $(\mathcal{H}, f)$ soit une hiérarchie indicée.*

Le reste de cette section porte sur les liens d'agrégation les plus usuels. Nous utiliserons la notation suivante. Si A et B sont deux parties de S, la notation $d(A, B)$ désignera le multi-ensemble des valeurs $d(a, b)$ pour $a \in A$ et $b \in B$, i.e. la collection de toutes les valeurs $d(a, b)$ (avec $a \in A$ et $b \in B$), ces valeurs étant répétées autant de fois que leur ordre de multiplicité. Par exemple, si $A = \{a_1, a_2\}$ et $B = \{b_1, b_2, b_3\}$, avec $d(a_1, b_1) = (a_1, b_2) = (a_1, b_3) = 1$ et $d(a_2, b_i) = i$ pour $i \in \{1, 2, 3\}$, alors $d(A, B)$ est le multi-ensemble $\{1, 1, 1, 1, 2, 3\}$. Si X désigne un multi-ensemble fini et non vide de nombres réels, alors les notations $\min X$, $\mathrm{Me}X$ and $\max X$ désigneront, respectivement, le minimum, la médiane et le maximum de X. Rappelons tout d'abord la définition des liens d'agrégation usuels :

- *lien simple* :

$\delta_{\min}(A, B) = \min d(A, B)$

- *lien complet* :

$\delta_{\max}(A, B) = \max d(A, B)$

- *lien de la médiane* :

$\delta_{\mathrm{med}}(A, B) = \mathrm{Me}\, d(A, B)$, où $\mathrm{Me}X$ désigne la médiane d'une partie X de $\mathbb{R}^+$.

- *lien moyen* (UPGMA) :

$$\delta(A,B) = \frac{1}{|A||B|}\sum_{a\in A}\sum_{b\in B} d(a,b)$$

- *lien moyen pondéré* (WPGMA) :

$\delta(A,B) = \dfrac{\delta(A_1,B)+\delta(A_2,B)}{2}$, où A a été formée par union de A_1 et A_2

- *lien de Hausdorff* :

$$\delta_H(A,B) = \max\{\max_{a\in A}\delta_{\min}(a,B), \max_{b\in B}\delta_{\min}(b,A)\}$$

- *lien Minimax* :

$$\delta_{\text{minimax}}(A,B) = \min_{x\in A\cup B}\delta_{\max}(x,(A\cup B)\setminus\{x\})$$

- *lien de Ward* :

$\delta_{\text{Ward}}(A,B) = \dfrac{w_A\, w_B}{w_A+w_B}\, d^2(c_A,c_B)$, où w_X désigne le poids de la classe X et c_X son centre de gravité.

- *lien du centroïde* :

$$\delta_{CeL}(A,B) = d(c_A,c_B)$$

- *lien du centroïde pondéré* (WPGMC) :

$$\delta_{\text{wc}}(A,B) = \frac{1}{2}\Big[\sum_{i=1}^{2}\delta_{\text{wc}}(A_i,B)\Big] - \frac{1}{4}\delta_{\text{wc}}(A_1,A_2),$$

où la classe A a été créée par agrégation de A_1 et A_2.

Plusieurs auteurs (e.g J.-P. Benzécri [5], D. Bryant et V. Berry [8]) ont montré que la hiérarchie d'Apresjan d'une dissimilarité quelconque est incluse dans chaque hiérarchie construite par la CAH lorsque cet l'algorithme utilise l'un des liens suivants : liens simple, complet et moyen.
Le théorème 2.7 et la remarque 2.8 précisent ce résultat et l'étendent à d'autres liens standard.

Theorem 2.7 *La hiérarchie d'Apresjan est incluse dans chaque hiérarchie $\mathcal{H}$ construite par la CAH avec le lien δ si δ est l'un des liens suivants : lien simple, complet, moyen, de la médiane, de la moyenne pondérée, de Hausdorff et du minimax.*
Pour chacun de ces liens, aucune classe d'Apresjan n'est supprimée lors de l'élagage de $\mathcal{H}$ effectué afin que $(\mathcal{H}, f)$ soit une hiérarchie indicée.

Remark 2.8 Les hiérarchies de Ward, du centroïde et du centroïde pondéré ne contiennent pas nécéssairement toutes les classes d'Apresjan. L'exemple 2.9 illustre le cas du lien de Ward.

Example 2.9 [lien de Ward]
Soit d la distance euclidienne définie sur $\mathbb{R}$. Soit $S = X\cup Y\cup Z$ où X, Y, Z sont définis par $X = \{0.1, 0.2\}$, $Y = \{4\}$ and $Z = \{8\}$. On remarque que $C = X\cup Y$ est une classe d'Apresjan puisque,

$$\max_{x,\,y\in C} d(x,y) = 3.9 < \min_{x\,\in\, C, z\,\notin\, C} d(x,z) = 4.$$

Cependant, $C = X\cup Y$ ne peut pas appartenir à la hiérarchie de Ward car, en choisissant des poids unitaires, on obtient :

$$\delta_{\text{Ward}}(X,Y) \approx 9.88 > \delta_{\text{Ward}}(Y,Z) = 8,$$

ce qui prouve que X et Y ne peuvent pas être réunies avant Y et Z, par conséquent que $C = X\cup Y$ n'appartient pas à la hiérarchie de Ward.

3 Une relaxation du critère définissant les classes d'Apresjan

Plusieurs auteurs ont proposé de relaxer le critère définissant les classes d'Apresjan qui, en pratique, s'avère strict (e.g. [8, 11]). Dans ce qui suit, nous proposons d'assouplir ce critère, en suivant une approche fondée sur la notion de fonction d'intervalle. Une fonction $I : S\times S \mapsto \mathcal{P}(S)$ est dite *fonction d'intervalle* si pour tout $x, y \in S$ on a

$$x \in I(x,y) = I(y,x).$$

Une partie $X \subseteq S$ est dite *I-convexe* si $I(x,y)\subseteq X$ pour tout $x, y \in X$. Notons que l'ensemble des parties I-convexes, noté $conv(I)$, est une convexité au sens d'une famille finie de parties fermée par intersection. Cette convexité est d'un type particulier appelé *convexité d'intervalle* induite par I.
Soient I_1 et I_2 deux fonctions d'intervalle sur S. On notera $I_1 \preceq I_2$ si $I_1(x,y)\subseteq I_2(x,y)$ pour tout $x, y \in S$. De plus, on désignera par $I_1\cup I_2$ la fonction d'intervalle définie par $(I_1\cup I_2)(x,y) = I_1(x,y)\cup I_2(x,y)$, pour tout $x, y \in S$. La proposition suivant présente deux propriétés élémentaires des convexités d'intervalle.

Proposition 3.1 *Soient I_1 et I_2 deux fonctions d'intervalle sur S.*
(i) Si $I_1 \preceq I_2$, alors $conv(I_2)\subseteq conv(I_1)$.
(ii) $conv(I_1\cup I_2) = conv(I_2)\cap conv(I_1)$.

On peut remarquer que les hiérarchies et les hiérarchies faibles (cf. [4]) sont des convexités d'intervalle de types particuliers. Ainsi on a la caractérisation suivante des hiérarchies.

Proposition 3.2 ([6])) *Les hiérarchies coïncident avec les convexités induites par les fonctions d'intervalle I telles que pour tout $x, y, z \in S$,*

$$I(x,y)\subseteq I(x,z) \text{ ou } I(x,z)\subseteq I(x,y). \qquad (1)$$

On vérifie facilement que la hiérarchie d'Apresjan de d coïncide avec la convexité induite par la fonction d'intervalle D_d définie pour tout $x, y \in S$ par :

$$\begin{aligned} D_d(x,y) &= \{z \mid \min\{d(x,z), d(y,z)\} \le d(x,y)\},\\ &= B_d(x, d(x,y))\cup B_d(y, d(x,y)),\end{aligned}$$

où $B_d(x,\rho)$ désigne la boule fermée de centre x et de rayon $\rho \ge 0$ au sens de la dissimilarité d. En d'autres termes, une

partie A de S est une classe d'Apresjan au sens de d si et seulement si A est une boule fermée (au sens de d) telle que pour tout $x, y \in A$, on a :

$$B_d(y, d(x, y)) \subseteq A.$$

Remark 3.3 *On peut observer que la hiérarchie d'Apresjan de d est la convexité induite par D_d, mais qu'en général D_d ne vérifie pas la propriété (1). Par ailleurs, plusieurs fonctions d'intervalle distinctes peuvent induire la même convexité (cf. [9]). Ainsi d'après la proposition 3.2, il existe nécessairement une autre fonction d'intervalle I qui vérifie (1) et qui induit la hiérarchie d'Apresjan de d. Une telle fonction d'intervalle I peut être construite, de la façon canonique suivante :*
Pour tout $x, y \in S$, $I(x, y) = \bigcap\{A \in \mathcal{H}_{\mathcal{A}}(d) \mid x, y \in A\}$.

Si d est une ultramétrique, on peut vérifier que la hiérarchie d'Apresjan de d coïncide avec l'ensemble des boules fermées au sens de d. Il en résulte que si d est ultramétrique et comporte peu de triplets équilatéraux, alors les classes d'Apresjan de d non triviales sont nombreuses, plus précisément de l'ordre de $n = |S|$, et leurs tailles ne sont pas toujours petites devant n. Par l'argument de continuité, cette propriété reste vraie, si au lieu d'être ultramétrique, d est approximativement ultramétrique, ou même approximativement ultramétrique sur un assez grand sous-ensemble de S. Dans la majorité des autres cas, il est fréquent d'observer que les classes d'Apresjan non triviales sont peu nombreuses et de petite taille, ce qui limite l'intérêt et l'usage de la hiérarchie d'Apresjan en pratique (cf. [8, 6]).
Afin de dépasser cette limitation, nous proposons une relaxation du critère qui définit la notion de classe d'Apresjan. Pour cela, nous introduisons tout d'abord une famille de fonctions d'intervalle $\{D_{d,\alpha}\}_\alpha$, où α désigne un paramètre variant dans $[0, 1]$ qui peut être choisi a priori arbitrairement et où $D_{d,\alpha}$ est la fonction d'intervalle définie pour tout $x, y \in S$ par :

$$\begin{aligned} D_{d,\alpha}(x, y) &= \{z \mid \min\{d(x, z), d(y, z)\} \leq \alpha\, d(x, y)\} \\ &= B_d(x, \alpha\, d(x, y)) \cup B_d(y, \alpha\, d(x, y)). \end{aligned}$$

Soit $\mathcal{F} = conv(I)$ une convexité d'intervalle quelconque. Notons $\mathcal{F}_\alpha(d)$ la convexité d'intervalle définie par

$$\mathcal{F}_\alpha(d) = \mathcal{F} \cap conv(D_{d,\alpha}) = conv(I \cup D_{d,\alpha}).$$

La proposition suivante est une conséquence directe de la définition de $\mathcal{F}_\alpha(d)$ et de la proposition 3.1.

Proposition 3.4 *Soit $\mathcal{F}$ une convexité d'intervalle arbitraire, et d une dissmilarité propre (i.e. $d(x, y) \neq 0$ si $x \neq y$). Pour tout α, β tels que $0 \leq \alpha \leq \beta \leq 1$, on a :*

$$\mathcal{F}_1(d) = \mathcal{H}_A(d) \cap \mathcal{F} \subseteq \mathcal{F}_\beta(d) \subseteq \mathcal{F}_\alpha(d) \subseteq \mathcal{F}.$$

Par la suite, on note $\mathcal{T}(S)$ l'ensemble des parties triviales de S.

Definition 3.5 *Soit d une dissimilarité propre. On appelle* rapport d'isolation *d'une partie non vide C de S la quantité $\iota_d(C)$ définie par :*

$$\iota_d(C) = \begin{cases} \min\limits_{x \in C}\left(\dfrac{\min\limits_{z \notin C} d(x, z)}{\max\limits_{y \in C} d(x, y)}\right), & \text{si } C \notin \mathcal{T}(S) \\ +\infty, & \text{sinon.} \end{cases}$$

On vérifie aisément que, pour tout $\alpha \geq 0$, une partie C de S est D_α-convexe si et seulement si $\iota_d(C) > \alpha$. Donc C est une classe d'Apresjan si et seulement si $\iota_d(C) > 1$.

Proposition 3.6 *Soit $v_1 < \cdots < v_m$ les valeurs (positives) distinctes de ι_d prises sur une convexité d'intervalle $\mathcal{F}$.*
En notant $v_0 = 0$, $v_{m+1} = +\infty$, on a :
(i) $\mathcal{F}_{v_m}(d) = \mathcal{F} \cap \mathcal{T}(S)$.
(ii) Soit $\ell \in \{0, \ldots, m\}$ et $\alpha \in [v_\ell, v_{\ell+1}[$, on a : $\mathcal{F}_\alpha(d) = \mathcal{F}_{v_\ell}(d)$.
(iii) S'il existe $\ell \in \{0, \ldots, m\}$ avec $1 \in [v_\ell, v_{\ell+1}[$, alors :

$$\mathcal{H}_A(d) \cap \mathcal{F} = \mathcal{F}_{v_{\ell+1}}(d) \subsetneq \ldots \subsetneq \mathcal{F}_{v_1}(d) \subsetneq \mathcal{F}_0(d) = \mathcal{F}.$$

La condition (iii) met en évidence une suite de hiérarchies emboîtées allant de la hiérarchie d'Apresjan à toute hiérarchie $\mathcal{F}$ la contenant, chacune d'elles résultant d'une relaxation croissante du critère définissant les classes d'Apresjan. Soit C une classe d'une hiérarchie $\mathcal{F}$. On peut observer que $C \in \mathcal{F}_\alpha(d)$ pour tout $\alpha \in [v_{\ell_C}, v_{\ell_C+1}[$ avec $v_{\ell_C+1} = \iota_d(C)$ et que ℓ_C est alors la plus grande valeur de ℓ pour laquelle $C \in \mathcal{F}_\alpha(d)$ pour tout $\alpha \in [v_\ell, v_{\ell+1}[$. Plus la valeur de v_{ℓ_C+1} est grande, et donc plus ℓ_C est grand, plus la structure de la classe C se rapproche de celle d'une classe d'Apresjan.
On en déduit que le rapport d'isolation $\iota_d(C)$ et son rang dans la suite $v_1, \ldots, v_m$ de la condition (iii), peuvent être interprétés comme des indicateurs de qualité, respectivement absolus et relatifs, de la classe C au sens du modèle de classe d'Apresjan.

Références

[1] M. Ackerman, S. Ben-David, S. Brânzei, and D. Loker. Weighted clustering : Towards solving the user's dilemma. *Pattern Recognition*, 120:108152, 2021.

[2] Julien Ah-Pine. An efficient and effective generic agglomerative hierarchical clustering approach. *Journal of Machine Learning Research*, 19(42):1–43, 2018.

[3] J. D. Apresjan. An algorithm for constructing clusters from a distance matrix. *Mashinnyi perevod : prikladnaja lingvistika*, 9:3–18, 1966.

[4] H.-J. Bandelt and A.W.M. Dress. Weak hierarchies associated with similarity measures : an additive clustering technique. *Bull. Math. Biology*, 51:133–166, 1989.

[5] J.-P. Benzécri. *L'Analyse des Données*. Dunod, Paris, 1973.

[6] P. Bertrand and J. Diatta. An interval convexity-based framework for multilevel clustering with applications to single-linkage clustering. *Discrete Applied Mathematics*, (342):38–63, 2024.

[7] H. H. Bock. *Automatische Klassifikation*. Vandenhoeck & Ruprecht, Göttingen, 1974.

[8] David Bryant and Vincent Berry. A structured family of clustering and tree construction methods. *Advances in Applied Mathematics*, 27(4):705–732, 2001.

[9] J. Calder. Some elementary properties of interval convexities. *J. Lond. Math. Soc.*, s2-3(3):422—428, 1971.

[10] G. Carlsson and F. Mémoli. Characterization, stability and convergence of hierarchical clustering methods. *Journal of Machine Learning Research*, (11):1425–1470, 2010.

[11] Claudine Devauchelle, Andreas W. M. Dress, Alexander Grossmann, Stefan Grünewald, and Alain Henaut. Constructing hierarchical set systems. *Annals of Combinatorics*, 8(4):441–456, 2005.

[12] Yongkweon Jeon and Sungroh Yoon. Multi-threaded hierarchical clustering by parallel nearest-neighbor chaining. *IEEE Transactions on Parallel and Distributed Systems*, 26(9):2534–2548, 2015.

[13] A. F. Parker-Rhodes and R. M. Needham. A reduction method for non-arithmetic data, and its application to thesauric translation. In *Information Processing, Proceedings of the International Conference on Information Processing, Paris, 1960*, pages 321–325. UNESCO, 1960.

[14] Baris Sumengen, Anand Rajagopalan, Gui Citovsky, David Simcha, Olivier Bachem, Pradipta Mitra, Sam Blasiak, Mason Liang, and Sanjiv Kumar. Scaling hierarchical agglomerative clustering to billion-sized datasets. 05 2021.

Explorer les structures d'observations communes, partiellement communes et spécifiques à plusieurs blocs de variables

Stéphanie Bougeard[1], Jean Michel Galharet[2], Mohamed Hanafi[2]

[1] Anses, Unité d'Epidémiologie et Bien-Etre, Ploufragan
[2] Oniris VetAgroBio, Unité StatSC, Nantes

stephanie.bougeard@anses.fr

Résumé

Dans le contexte multibloc exploratoire en grande dimension, les utilisateurs sont intéressés pour explorer et interpréter les parts d'information communes, partiellement communes, spécifiques et résiduelles de chaque bloc de variables. Des méthodes orginales et récentes, telles que sparse-SCA, DISCO-SCA, PO-PLS, OnPLS, JIVE, COBE, SLIDE ou D-GCCA, apportent ces réponses. Cependant, l'ensemble de ces méthodes n'a jamais été comparé et les comparaisons (partielles) existantes ne sont pas associées à des structures simulées et interprétables des observations. Dans cette étude, nous proposons une comparaison méthodologique et pratique de ces méthodes.

Mots-clés

Analyse multibloc exploratoire, classification.

Abstract

In the high-dimensional exploratory multiblock framework, users are interested in exploring and interpreting common, partially common, specific and residual information parts of each block of variables. Original and recent methods, such as sparse-SCA, DISCO-SCA, PO-PLS, OnPLS, JIVE, COBE, SLIDE or D-GCCA, provide solutions. However, all of these methods have never been compared, and the (partial) comparisons that do exist are not associated with simulated observation-structures that can be interpreted. In this study, we propose a methodological and practical comparison of these methods.

Keywords

Exploratory multiblock analysis, clustering.

1 Introduction

Contexte. A l'heure des big data et de la fusion de données, les scientifiques — biologistes en particulier — recherchent des méthodes statistiques leur permettant d'étudier des liens complexes entre variables — issues de différentes sources et mesurées sur les mêmes observations — dans des espaces de dimensions réduites. Les méthodes multiblocs sont des outils adéquats pour y répondre. Nous nous plaçons dans un contexte exploratoire en grande dimension (méthodes multiblocs 'component-based' non supervisées). Pour aller plus loin dans l'exploration et l'interprétation des données, certaines méthodes multiblocs permettent de plus de quantifier les parts d'information communes (C), partiellement communes (PC), spécifiques (S) et résiduelles (R) de chaque bloc [7, 6]. Ces méthodes originales ont été appliquées, par exemple, dans les domaines de la chimiométrie [1], de l'analyse d'image [5] ou du 'multi-omique' [14].

Etat de l'art. Les méthodes multiblocs permettant de quantifier les parts d'information C/PC/S/R de chaque bloc peuvent être classées en deux catégories. (i) La première est basée sur des méthodes standards qui caractérisent *a posteriori* leurs composantes selon le statut C/PC/S/R. Parmi celles-ci, on peut citer les méthodes sparse-SCA [16] et DISCO-SCA [11]. L'attribution de tels statuts est aussi possible à partir de la factorisation canonique appliquée aux méthodes CPCA/SCA, MCOA, MFA, STATIS, CCSWA [3]. (ii) La seconde catégorie de méthodes — généralement associée à un critère additif de ces parts — recherche *a priori* des composantes de statut C/PC/S/R au sein des blocs de variables. Parmi ces méthodes, on peut citer PO-PLS [12], OnPLS [15], JIVE [13], COBE [10], SLIDE [8] ou D-GCCA [4]. L'ensemble de ces méthodes n'a jamais été comparé, bien que certaines comparaisons partielles existent, e.g., [1, 8, 9, 4, 2]. Ces comparaisons ne sont de plus pas associées à des structures simulées et interprétables des observations.

Contributions. (i) Notre premier objectif est de comparer les performances des méthodes multiblocs précedemment citées sur des données simulées dont les structures des observations sont maîtrisées et sur des indices clairs. (ii) Notre second objectif est d'interpréter ces parts relativement aux blocs, à la structure des observations (e.g., en classes) et aux variables. En effet, chaque bloc peut contenir à la fois des informations C/PC/S/R, chacune étant associée à différentes structures des mêmes observations (par ex., une structure commune en trois classes, superposée à une structure partiellement commune en deux classes) et à différentes variables de chaque bloc (par ex., certaines variables d'un bloc contribuent à la part commune, d'autres à la part spécifique).

2 Méthodes

Données. Soit K blocs de variables $(\mathbf{X}_1, \ldots, \mathbf{X}_K)$ mesurés sur les mêmes N observations et comportant respectivement $(P_1, \ldots, P_K)$ variables avec $P = \sum_{k=1}^{K} P_k$.

Définition. La définition des parts d'information spécifique ou partiellement commune est dissymétrique et nécessite une référence. On suppose ici que cette référence est l'information commune portée par les données concaténées $\mathbf{X} = [\mathbf{X}_1 | \ldots | \mathbf{X}_K]$.

Décomposition. Que la reconstitution des parts d'information C/PC/S/R de chaque bloc $\mathbf{X}_k$ (pour $k = 1, \ldots, K$) soit effectuée *a posteriori* ou *a priori*, celle-ci est considérée additive selon le modèle :

$$\mathbf{X}_k = \mathbf{C}_k + \mathbf{PC}_k + \mathbf{S}_k + \mathbf{R}_k \quad , \tag{1}$$

avec $\mathbf{C}_k$ la part d'information commune entre ce bloc et tous les autres blocs, $\mathbf{PC}_k$ la part d'information partiellement commune entre ce bloc et certains autres blocs, $\mathbf{S}_k$ la part d'information spécifique à ce bloc et $\mathbf{R}_k$ la part d'information résiduelle de ce bloc. Certaines méthodes multiblocs ne permettent pas d'identifier les parts d'information partiellement communes (par ex., PO-PLS, JIVE, COBE) ; elles sont dans ce cas (à tort) intégrées aux parts spécifiques [2].
Pour chaque bloc $\mathbf{X}_k$, chacune de ces parts est considérée dans un espace de dimension réduite (resp. $H_C < P$, $H_{PCk} < P$ et $H_{Sk} < P_k$) par des composantes (resp. $\mathbf{T}_C$, $\mathbf{T}_{PCk}$ et $\mathbf{T}_{Sk}$) associées à des loadings (resp. $\mathbf{P}_{Ck}$, $\mathbf{P}_{CPk}$ et $\mathbf{P}_{Sk}$) selon le modèle (2) équivalent au modèle (1) :

$$\mathbf{X}_k = \mathbf{T}_C \mathbf{P}_{Ck}^T + \mathbf{T}_{PCk} \mathbf{P}_{PCk}^T + \mathbf{T}_{Sk} \mathbf{P}_{PSk}^T + \mathbf{R}_k \quad . \tag{2}$$

Ces éléments sont résumés par la Figure 1.
Pour assurer une séparation optimale entre les parts d'information C/PC/S/R de chaque bloc, le choix de la dimension de chaque sous-espace (H_C, H_{PCk}, H_{Sk}) se revèle crucial et est abordé différemment selon les méthodes. De plus, les orthogonalités entre composantes — au sein de chaque sous-espace et entre les sous-espaces — posent de fortes contraintes, abordées aussi différemment selon celles-ci.

Simulations. Afin de comparer l'efficacité de la séparation en parts d'information C/PC/S/R par différentes méthodes multiblocs (sparse-SCA, DISCO-SCA, PO-PLS, OnPLS, JIVE, COBE, SLIDE, D-GCCA), plusieurs scénarios sont simulés et associés à des structures en classes des observations. $K = 3$ blocs de variables $(\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3)$ — mesurés sur les mêmes $N = 90$ observations selon respectivement $P = (50, 50, 50)$ variables — sont simulés. Trois scénarios de contrôle sont d'abord proposés : le premier (**S1**) ne comporte qu'une structure des observations en $G = 3$ classes communes à tous les blocs dans un espace de dimension $H_C = 2$ (e.g., $70\% - 20\%$ d'inertie) ainsi qu'une part résiduelle. Le second scénario (**S2**) comporte trois structures des observations en $G = 2$ classes différentes selon les blocs (e.g., $70\% - 20\%$ d'inertie) ainsi qu'une part résiduelle. Le troisième scénario (**S3**), considéré comme une structure résiduelle, ne comporte aucune structure des observations. Puis plusieurs scénarios mêlant, au sein de chaque bloc, des structures communes, partiellement communes et spécifiques sont évalués.

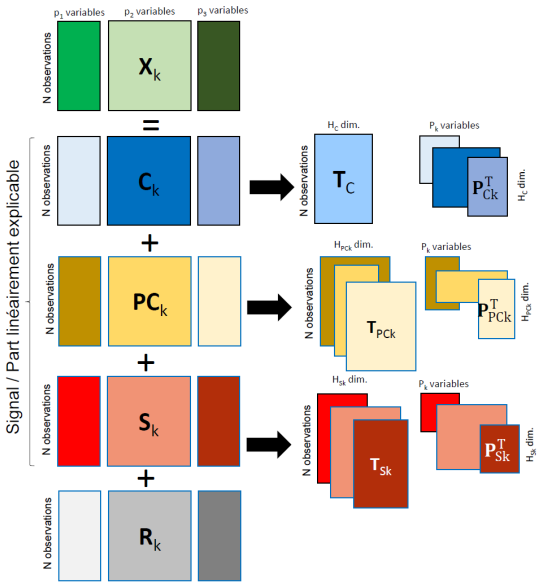

FIGURE 1 – Illustration de K blocs de variables $\mathbf{X}_k$ (ici $K = 3$) décomposés selon une structure (additive) en information commune aux K blocs ($\mathbf{C}_k$), partiellement commune à certains blocs ($\mathbf{PC}_k$), spécifique à ce bloc ($\mathbf{S}_k$) et résiduelle ($\mathbf{R}_k$). Chacune de ces parts est décomposée selon des composantes et des loadings dans une dimension réduite spécifique.

Indices de performance. Les méthodes multiblocs présentant des sorties différentes, les composantes $\mathbf{T}_C$, $\mathbf{T}_{PCk}$ et $\mathbf{T}_{Sk}$ sont tout d'abord récupérées (parfois recalculées, e.g., JIVE). A partir de ces composantes, plusieurs indices sont calculés. (i) Les dimensions de chaque sous-espace (H_C, H_{PCk}, H_{Sk}) sont comparées aux dimensions simulées. (ii) Les variances expliquées de chaque bloc pour chaque dimension et chaque part C/PC/S sont comparées aux variances expliquées simulées. (iii) La capacité à retrouver la(es) structure(s) simulée(s) des observations de chaque part C/PC/S en comparant la(es) structure(s) simulée(s) et celle(s) observée(s) par indice de Rand ajusté. Plusieurs autres critères sont évalués : l'unicité de la solution, l'orthogonalité des composantes, la prise en compte de l'information partiellement commune et le temps de calcul.

3 Conclusion et discussion

La décomposition — dans des espaces de dimensions réduites — de blocs de variables mesurés sur les mêmes observations, apporte des informations originales et d'intérêt aux utilisateurs pour l'interprétation des blocs et des liens entre ceux-ci. Plusieurs méthodes multiblocs —

sparse-SCA, DISCO-SCA, PO-PLS, OnPLS, JIVE, COBE, SLIDE, D-GCCA, pour les principales — apportent ces réponses. Bien que ces méthodes aient déjà été comparées (partiellement), elles ne l'ont jamais été toutes ensemble. De plus, leur comparaison n'a jamais été associée à des structures d'observations en classes dans des dimensions fixées, ni à des indices de performance associés à ces structures simulées. Cette étude apporte des réponses claires aux utilisateurs pour le choix de la méthode la plus performante et la mieux adaptée à leurs données.

L'application de ces méthodes originales et récentes pose encore de nombreuses questions méthodologiques et pratiques. Du point de vue méthodologique tout d'abord, la définition de ce qui est spécifique est dissymétrique et nécessite une référence, question qui n'est jamais discutée clairement. De plus, pour toutes ces méthodes, la décomposition des blocs est supposée additive et est généralement déterminée dans un ordre donné (part résiduelle, puis commune, puis spécifique / partiellement commune). La décomposition obtenue serait-elle la même si les décompositions étaient construites différemment ? Les questions cruciales de la dimension de chaque sous-espace ainsi que des orthogonalités entre composantes méritent d'être discutées et optimalement solutionnées. Du point de vue pratique, l'interprétation de chaque sous-espace mérite d'être mieux exploitée, notamment par l'étude des structures d'observations (e.g., par reduced/factorial K-means) et par celles des variables du bloc qui leur sont associées.

Références

[1] A.K. Smilde I. Mage T. Naes T. Hankemeier M.A. Lips H.A.L. Kiers E. Acar R. Bro. Common and distinct components in data fusion. *Journal of Chemometrics*, 31, 2007.

[2] S. Yi R.K.W. Wong I. Gaynanova. Hierarchical nuclear norm penalization for multi-view data. *Biometrics*, 79(4), 2013.

[3] S. Bougeard C. Peltier B. Jaillais J.C. Boulet M. Hanafi. Benchmarking multiblock methods with canonical factorization. *Chemometrics and Intelligent Laboratory Systems*, 254 :105240, 2024.

[4] H. Shu Z. Qu H.Zhu. D-GCCA : Decomposition-based generalized canonical correlation analysis for multi-view high-dimensional data. *Journal of Machine Learning Research*, 23, 2022.

[5] N. Shi S. Fattahi R. Al Kontar. Triple component matrix factorization : Untangling global, local, and noisy components. *Journal of Machine Learning Research*, 25, 2024.

[6] M. Hanafi R. Lafosse. Généralisation de la régression simple pour analyser la dépendance de K ensembles de variables avec un K+1ième. *Revue de statistique appliquée*, XLIX(1) :5–30, 2001.

[7] R. Lafosse. Analyse de concordance de deux tableaux : monogamie, simultanéités et découpages. *Revue de statistique appliquée*, XLV(3) :45–72, 1997.

[8] I. Gaynanova G. Li. Structural learning and integrative decomposition of multi-view data. *Biometrics*, 75(4) :1121–1132, 2019.

[9] J.Y. Park E.F. Lock. Integrative factorization of bidimensionally linked matrices. *Biometrics*, 76(1) :61–74, 2019.

[10] G. Zhou A. Cichocki Y. Zhang D. Mandic. Group component analysis for multi-block data : Common and individual feature extraction. *IEEE transactions on neural networks and learning systems*, 2017.

[11] M. Schouteden K. Van Deun S. Pattyn I. Van Mechelen. SCA and rotation to distinguish common and distinctive information in linked data. *Behavior Research Methods*, 45, 2013.

[12] I. Mage M. Bjorn-Helge T. Naes. Regression models with process variables and parallel blocks of raw material measurements. *Journal of Chemometrics*, 22, 2008.

[13] E.F. Lock K.A. Hoadley A.B. Nobel. Joint and individual variation explained (JIVE) for integrated analysis of multiple data types. *The Annals of Applied Statistics*, 7(1) :523–542, 2013.

[14] T. Lofstedt D. Hoffman J. Trygg. Global, local and unique decompositions in onPLS for multiblock data analysis. *Analytica Chimica Acta*, 791, 2013.

[15] T. Lofstedt J. Trygg. OnPLS : a novel multiblock method for the modelling of predictive and orthogonal variation. *Journal of Chemometrics*, 25 :441–455, 2011.

[16] K. Van Deun T.F. Wilderjans R.A. van den Berg et al. A flexible framework for sparse simultaneous component based data integration. *BMC Bioinformatics*, 12, 2011.

Problèmes de sériation dans les graphes

François Brucker[1,2], Pascal Préa[1,2]

[1] Aix-Marseille Université, CNRS, Université de Toulon, LIS, Marseille, France
[2] École Centrale Méditerranée, Marseille, France

{francois.brucker, pascal.prea}@lis-lab.fr

Résumé

La sériation *consiste à chercher un ordre linéaire sous-jacent à un ensemble de données. Nous présentons une extension de ce problème aux graphes. Nous obtenons ainsi une grande variété de problèmes intéressants, de diverses complexités.*

Mots-clés

Sériation, graphes, dissimilarités, complexité.

Abstract

The goal of seriation *is to find a linear order subjacent to a data set. We present here an extension of this problem to graphs. We thus get numerous interesting problems, of various complexities.*

Keywords

Seriation, graphs, dissimilarities, complexity.

1 Introduction

Le problème de classification que nous nous proposons d'étudier est une variation du problème initialement posé par [7] : comment ordonner un ensemble d'objets décrit par une distance selon un *ordre compatible*.
De façon formelle, le problème est ainsi posé dans [7] : soit X un ensemble d'objet muni d'une distance $d : X \times X \to \mathbb{R}^+$. On cherche s'il existe un ordre total $\leq$ entre les éléments de X tel que $x \leq y \leq z$ implique $d(x,z) \geq \max(d(x,y), d(y,z))$. Un tel ordre est dit compatible.
Un tel ordre n'existant pas forcément, on cherche à minimiser les incompatibilités : c'est à dire trouver un ordre $\leq$ entre les éléments de X minimisant la somme :

$$\sum_{(x,y,z)\in I} (\max(d(x,y), d(y,z)) - d(x,z))$$

Avec $I \subseteq X^3$ l'ensemble des triplets (x,y,z) tels que $x \leq y \leq z$ et $d(x,z) < \max(d(x,y), d(y,z))$.
On peut noter que si trouver un ordre compatible (s'il existe) est polynomial (voir par exemple [6] pour le premier algorithme optimal résolvant ce problème), trouver la meilleur approximation est NP-difficile (voir [2] pour une formulation du problème en norme L_1 et [4] pour sa formulation en norme L_∞).

Ce problème d'ordonnancement a été initialement posé pour résoudre des problèmes archéologiques où l'on cherche à trouver un ordre compatible "temporel" entre plusieurs artefacts (l'ensemble X) créés à des époques différences en un même lieu (une même "civilisation"). Les différences entres les artefacts (la distance entre les éléments de X) sera d'autant plus faible qu'ils ont été créés à des époques proches. Nous nous intéressons dans cette communication au problème dual : on cherche un ordre compatible "spacial" entre des objets produit au même moment.
On considère un ensemble X d'éléments (*ie.* des artefacts) reliés entre eux par un graphe $G = (X, E)$ (*ie.* les artefacts ont été créés par des civilisations reliés entres elles par des nœuds de communications, dont on prend un arbre couvrant) et munis d'une distance $d : X \times X \to \mathbb{R}^+$ (*ie.* la différences entre les caractères). On cherche alors une orientation de $\overrightarrow{E}$ maximale de G telle que tous les chemins soient compatibles [1] avc la dissimilarité d.
Après avoir défini formellement le problème, nous donnons les principaux résultats obtenus puis concluons sur quelques perspectives algorithmiques.
Nous utilisons pour cela des travaux préliminaires [1] en les replaçant dans le cadre général de l'orientation de graphes qui abordent des problématiques connexes [5, 3].

2 Définition du Problème

Le problème de la sériation peut se généraliser de plusieurs manières selon le type de contraintes que l'on veut généraliser :

1. Nature de la relation (ordre total, partiel, classes d'équivalence),
2. Nature du critère : global ou local, extrinsèque ou intrinsèque,...
3. Existence, optimisation ou approximation d'une solution,

Nous proposons dans cette présentation un cadre général permettant de traiter (séparément ou conjointement) ces différents problèmes.

1. Un chemin orienté $(v_1, \ldots, v_n)$ est compatible avec d si $i < j < k \implies d(x_i, x_k) \geq \max\{d(x_i, x_j), d(x_j, x_k)\}$, où $\forall u \in \{1, \ldots, n\}, v_u = \phi(x_u)$.

Nos données de départ sont un ensemble X muni d'une dissimilarité d et un graphe $G = (V, E)$ avec $|V| = |X|$. On cherche alors une orientation de G [2] et une fonction ϕ (qui sera le plus souvent une bijection) entre X et V satisfaisant certains critères.

Par exemple, déterminer si (X, d) est Robinson revient à considérer le chemin P_n comme graphe G et à chercher une bijection entre X et $[n]$ tel que le chemin orienté $\overrightarrow{P_n}$ soit compatible avec d.

Selon le type de graphe considéré ainsi que le critère à optimiser, on aura des complexités radicalement différentes.

3 Résultats

(X, d) et G sont toujours connus. Selon le problème, on fixera ϕ ou l'orientation de G. Selon la nature de G, on a différentes classes de problèmes.

3.1 G est un chemin

On montre ici que même sur les graphes support les plus simples, la détermination de la bijection ϕ rend le problème NP-complet (3.1.2). Pour les graphes plus complexes, on supposera donc ϕ connue, *i.e.* X est l'ensemble des sommets du graphe support.

3.1.1 Orientation et ϕ inconnues

N'importe quelle orientation sans chemin de longueur > 1 (obtenue en orientant les arêtes tête-bêche) est une solution du problème, mais celle-ci ne maximise généralement pas le nombre de chemins compatibles.

3.1.2 Orientation connue et ϕ inconnue

Le problème est NP-complet, sauf si l'orientation correspond à un chemin (auquel cas, on est ramené à la reconnaissance des dissimilarités de Robinson).

Cela se démontre par réduction à partir du problème SUBSET-ROB suivant (qui lui-même se réduit à partir du Chemin Hamiltonien) : étant donné un espace (X, d) et un entier $k < n$, existe-il $Y \subset X$, tel que $|Y| = k$ et (Y, d) soit Robinson ?

3.1.3 Orientation inconnue et ϕ connue

On peut résoudre ce problème en temps polynomial par programmation dynamique.

3.2 G est un arbre

3.2.1 d constant

Le problème revient à trouver une orientation d'un arbre maximisant le nombre de chemins. Une caractérisation d'une telle orientation est qu'il existe un sommet c tel que pour tout sommet x du graphe, il existe un chemin de x vers c ou de c vers x. À partir de cette caractérisation, il est facile de trouver un algorithme polynomial pour ce problème.

2. Étant donné un *graphe support* $G = (X, E)$, une *orientation* $\overrightarrow{E}$ de G est un sous-ensemble de $X \times X$ où :
   — chaque élément $(x, y) \in \overrightarrow{E}$ est associé à une arête $\{x, y\}$,
   — pour toute arête $\{x, y\} \in E$, soit $(x, y) \in \overrightarrow{E}$, soit $(y, x) \in \overrightarrow{E}$.

3.2.2 d non constant

Le problème est NP-complet, même sur les graphes de diamètre 4 (on montre ceci par réduction à partir de 3-SAT). Sur les étoiles (les graphes de diamètres 2), le problème est polynomial. Le cas des graphes de diamètre 3 reste ouvert.

3.3 G est quelconque

Si d est constant, on peut se ramener au cas des arbres en décomposant le graphe en composantes 2-connexes. Si d n'est pas constant, ce problème est une généralisation de 3.2.2, il est donc NP-complet.

4 Conclusion

Dans cette présentation, nous avons donné un cadre formel pour généraliser le problème de la sériation aux graphes. On obtient ainsi une grande variété de problèmes, certains polynomiaux, d'autres NP-durs, et pour quelques uns d'entre eux, la classe de complexité reste inconnue. Déterminer ces complexités fait partie de nos perspectives.

Références

[1] F. Brucker, P. Préa, and C. Thraves Caro. Arbres & chemins. *SFC 2024, Marseille*, 2024.

[2] François Brucker. *Modèles de classification en classes empiétantes*. PhD thesis, École des Hautes Études en Sciences Sociales, 2001.

[3] R.M. Casablanca, P. Dankelmann, W. Goddard, L. Mol, and O. Oellermann. The maximum average connectivity among all orientations of a graph. *Journal of Combinatorial Optimization*, 43 :543–570, 2022.

[4] V. Chepoi, B. Fichet, and M. Seston. Seriation in the presence of errors : Np-hardness of l_∞-fitting robinson structures to dissimilarity matrices. *Journal of Classification*, 26 :279–296, 2009.

[5] F. Hörsch. On orientations maximizing total arc-connectivity. *Theoretical Computer Science*, 978 :114176, 2023.

[6] P. Préa and D. Fortin. An optimal algorithm to recognize robinsonian dissimilarities. *Journal of Classification*, 31 :351–385, 2014.

[7] W. S. Robinson. A model for chronological ordering of archeological deposits. *American antiquity*, 16 :295–301, 1951.

Dissimilarités de Robinson multi-voies

Victor Chepoi[1] Guyslain Naves[1] Pascal Préa[1,2]

[1] Aix-Marseille Université, CNRS, Université de Toulon, LIS, Marseille, France
[2] École Centrale Méditerranée, Marseille, France

{victor.chepoi, guyslain.naves, pascal.prea}@lis-lab.fr

Résumé

Dans [12], M.J Warrens et J.W. Heiser ont donné deux généralisations des dissimilarités de Robinson [11] aux dissimilarités 3-voies.
Nous étendons ici ces travaux en d'une part proposant deux autres extensions des dissimilarités de Robinson, et d'autre part en les définissant pour $k > 3$. De plus, nous donnons des algorithmes efficaces, voire optimaux, pour trois de ces définitions.

Mots-clés

Sériation, dissimilarités de Robinson, dissimilarités mult-voies.

Abstract

In [12], M.J Warrens and J.W. Heiser introduced two generalizations of Robinson dissimilarities [11] to three-way dissimilarities.
We extend here this work by, on one hand, giving two other generalizations of Robinson dissimilarities and, on the other hand, extending these definitions to $k > 3$. In addition, we give efficient, or even optimal, algorithms for three of these definitions.

Keywords

Seriation, Robinson dissimilarities, multi-way dissimilarities.

1 Introduction

Une dissimilarité (*i.e.* une distance sans l'inégalité triangulaire ni axiome de séparation) d sur un ensemble X est *Robinson* [11] si il existe un ordre total (dit *compatible*) sur X tel que $x < y < z \implies d(x,z) \geq \max\{d(x,y), d(y,z)\}$.
Les dissimilarités de Robinson ont été introduites pour résoudre le problème de la sériation en archéologie et sont depuis un outil fondamental pour la sériation dans n'importe quel domaine. De plus, elles généralisent les ultramétriques et sont équivalentes aux pyramides, le modèle standard pour la classification en classes empiétantes [6, 7]. Enfin, elle jouent un rôle important dans la reconnaissance de cas polynomiaux pour le problème du voyageur de commerce [2, 3].

Une dissimilarité k-voies sur un ensemble X est une fonction $d : X^k \longmapsto \mathbb{R}^+$ à diagonale nulle ($\forall x \in X, d(x,x,\ldots,x) = 0$) et telle que $\forall x_1,\ldots,x_k \in X, d(x_1,\ldots,x_k) = d(x_{\sigma(1)},\ldots,x_{\sigma(k)})$ pour toute permutation σ de $[k]$. Un *cube* est une dissimilarité 3-voies. Ces dissimilarités ont souvent été étudiées dans le cadre de la classification [9, 4, 8, 5].
M.J Warrens et J.W. Heiser ont défini [12] les *cubes de Robinson* et les *cubes réguliers de Robinson*. La définition des cubes de Robinson ne fait intervenir que n^2 valeurs (sur n^3) alors que la définition des cubes réguliers de Robinson impose à n^2 valeurs d'être nulles (et pas seulement la diagonale).

Nous présentons ici deux autres extensions des dissimilarités de Robinson aux cubes, qui se situent "entre" les deux définitions de [12]. De plus, nous étendons toutes ces définitions à $k > 3$ et nous donnons des algorithmes efficaces (et même optimal pour une extension) pour trois de ces définitions.

2 Définitions

On représente généralement une dissimilarité sur un ensemble X à n éléments par une matrice $n \times n$ à diagonale nulle. Une matrice à diagonale nulle est *Robinsonienne* si toutes ses lignes et colonnes sont non décroissantes en s'éloignant de la diagonale (une dissimilarité d sur un ensemble X est Robinson si X peut être ré-ordonné de telle sorte que d soit représentée par une matrice Robinsonienne). Un vecteur $(x_1,\ldots,x_n)$ est *Robinsonien* si il existe $1 \leq \ell \leq n$ tel que $1 \leq i < \ell \implies x_i \geq x_{i+1}$ et $\ell < i \leq n \implies x_i \geq x_{i-1}$. Toutes les lignes et colonnes d'une matrice Robinsonienne sont des vecteurs Robinsoniens ; l'inverse est faux.
Étant donnée une dissimilarité k-voies d sur un ensemble X, que l'on prend ordonné en $(x_1,\ldots,x_n)$:

- Si $(t_1,\ldots,t_{k-1})$ est un vecteur d'éléments de X, la *droite* de rang ℓ indexée par $(t_1,\ldots,t_{k-1})$ est le vecteur de $\mathbb{R}^n$ $(v_1,v_2,\ldots,v_n)$ où $v_i = d(t_1,\ldots,t_{\ell-1},x_i,t_\ell,\ldots,t_{k-1})$ pour tout $1 \leq i \leq n$ (quand $k = 2$, les droites sont les lignes et les colonnes de la matrice).
- Si $(t_1,\ldots,t_{k-2})$ est vecteur d'éléments de X, le *plan* de rangs $\ell < \ell'$ indexé par $(t_1,\ldots,t_{k-2})$ est la matrice $n \times n$ M où $M[i,j] = d(t_1,\ldots,t_{\ell-1},x_i,t_\ell,\ldots,t_{\ell'-2},x_j,t_{\ell'-1},\ldots,t_{k-2})$

— La *diagonale* de d est l'ensemble de points de $\mathbb{R}^k$ $\{(x_i, \ldots, x_i), 1 \leq i \leq n\}$.

De part les symétries d'une dissimilarité k-voies, on a :

Proposition 1 *Étant donnée une dissimilarité k-voies d sur un ensemble X. (le tableau k-dimensionel représentant d est construit en supposant X ordonné en $(x_1, \ldots, x_n)$). On a :*

1. *Deux droites indexées par $(t_1, \ldots, t_{k-1})$ et $(t_{\sigma(1)}, \ldots, t_{\sigma(k-1)})$ sont égales pour toute permutation σ de $\{1, \ldots, k-1\}$, quels que soient leurs rangs.*
2. *Deux plans indexés par $(t_1, \ldots, t_{k-2})$ et $(t_{\sigma(1)}, \ldots, t_{\sigma(k-2)})$ sont égaux pour toute permutation σ de $\{1, \ldots, k-2\}$, quels que soient leurs rangs.*

Soit $\mathbf{a} = (a_1, \ldots, a_k)$ un k-uplet à valeurs dans $[n]$. La *norme* de $\mathbf{a}$, notée $\|\mathbf{a}\|$ est la distance Euclidienne entre $\mathbf{a}$ et la diagonale. Deux k-uplets $\mathbf{a} = (a_1, \ldots, a_k)$ et $\mathbf{b} = (b_1, \ldots, b_k)$ sont *voisins* si $\sum |a_i - b_i| = 1$ (*i.e.* ils sont voisins selon la norme L_1).

Soit $X = (x_1, \ldots, x_n)$ un ensemble ordonné. La *norme* $\|\mathfrak{a}\|$ d'un k-uplet $\mathfrak{a} = (x_{i_1}, x_{i_2}, \ldots, x_{i_k})$ d'éléments de X est égale à $\|\mathbf{i}\|$, où $\mathbf{i} = (i_1, \ldots i_k)$. De même, deux k-uplets $\mathfrak{a} = (x_{i_1}, x_{i_2}, \ldots, x_{i_k})$ et $\mathfrak{b} = (x_{j_1}, x_{j_2}, \ldots, x_{j_k})$ d'éléments de X sont *voisins* si $(i_1, \ldots, i_k)$ et $(j_1, \ldots, j_k)$ sont voisins.

Nous pouvons maintenant définir les extensions des dissimilarités de Robinson aux dissimilarités k-voies.

— Une dissimilarité k-voies d sur un ensemble X est *diagonale-Robinson* si X peut être ordonné de telle sorte que toute droite de d coupant la diagonale est Robinsonienne. Pour $k = 3$, ces dissimilarités sont appelées *Robinson Cubes* dans [12].

— Une dissimilarité k-voies d sur un ensemble X est *droite-Robinson* si X peut être ordonné de telle sorte que toute droite de d est Robinsonienne.

— Une dissimilarité k-voies d sur un ensemble X est *Robinson* si X peut être ordonné de telle sorte que, si $\mathfrak{a}$ et $\mathfrak{b}$ sont deux k-uplets voisins d'éléments de X, $\|\mathfrak{a}\| < \|\mathfrak{b}\| \implies d(\mathfrak{a}) \leq d(\mathfrak{b})$.

— Une dissimilarité k-voies d sur un ensemble X est *plan-Robinson* si X peut être ordonné de telle sorte que tout plan de d est Robinsonien. Pour $k = 3$, ces dissimilarités sont appelées *Regular Robinson Cubes* dans [12].

Ces définitions ne sont pas indépendantes :

Proposition 2 *Soit d une dissimilarité k-voies sur un ensemble X.*

1. *Si d est plan-Robinson, alors d est Robinson.*
2. *Si d est Robinson, alors d est droite-Robinson.*
3. *Si d est droite-Robinson, alors d est diagonale-Robinson.*

On peut remarquer que, si on avait défini la norme d'un k-uplet en utilisant la norme L_1, l'implication *2* aurait été fausse.

3 Algorithmes

3.1 Dissimilarités plan-Robinson

Il est possible de reconnaitre les dissimilarités de Robinson en temps optimal $O(n^2)$ [10]. Cet algorithme donne en plus l'ensemble des permutations compatibles sous la forme d'un PQ-tree [1].

D'autre part, l'algorithme de [10] peut facilement être adapté, avec la même complexité, pour qu'il prenne en entrée une dissimilarité d et un ensemble de permutations $\mathcal{S}$ représenté par un PQ-tree et rende en sortie un PQ-tree représentant toutes les permutations de $\mathcal{S}$ qui sont compatibles avec d.

En lançant cet algorithme successivement pour tous les plans de d, on construit le PQ-tree représentant tous les ordres compatibles de d (si d n'est pas plan-Robinson, le PQ-tree obtenu est vide).

Par la Proposition 1, il suffit de tester les plans de rang 1 et 2 (ou n'importe quel autre couple d'indices). Comme il y a n^{k-2} tels plans ($n := |X|$), o)n en déduit :

Proposition 3 *Il est possible de déterminer si une dissimilarité k-voies d sur un ensemble X à n éléments est plan-Robinson en temps optimal $O(n^k)$.*

3.2 Dissimilarités droite et diagonale Robinson

Réordonner une droite de d pour la rendre Robinsonienne se fait en deux étapes :

1. Dans un premier temps, on trie ses valeurs. On obtient alors des ensembles imbriqués les uns dans les autres (chaque ensemble correspond à un seuil). Ceci prend un temps $O(n \log n)$
2. On construit le PQ-tree représentant toutes les permutations pour lesquelles les ensembles imbriqués obtenus à l'étape 1 sont des intervalles. Ceci prend $O(n)$.

On peut là encore adapter l'étape 2 de manière à prendre aussi en entrée un ensemble S de permutations représenté par un PQ-tree. On peut donc ainsi reconnaître les dissimilarités diagonal-Robinson et droite-Robinson. Par la proposition 1, il suffit de tester les droites de rang 1.

Comme il y a n droites de rang 1 croisant la diagonale, on a :

Proposition 4 *Il est possible de déterminer si une dissimilarité k-voies d sur une un ensemble X à n éléments est diagonale-Robinson en temps $O(n^2 \log n)$.*

Il y a n^{k-1} droites de rang 1 en tout. On en déduit :

Proposition 5 *Il est possible de déterminer si une dissimilarité k-voies d sur une un ensemble X à n éléments est droite-Robinson en temps $O(n^k \log n)$.*

4 Conclusion

On a ici étendu les définitions les définitions de Warren et Heiser, d'une part d'un point de vue formel et d'autre part en les donnant pour n'importe quelle dimension (et non pas seulement 3). Nous obtenons ainsi quatre définitions qui étendent les dissimilarités de Robinson aux dimensions quelconques.
Nous avons également donné des algorithmes optimaux, ou presque (avec un écart logarithmique par rapport à l'optimal) pour trois de ces définitions. Il nous reste à trouver un algorithme efficace pour la quatrième définition, d'autant plus que celle-ci nous parait la plus fidèle à la définition d'origine (en dimension 2).

Références

[1] Booth, K.S. & G.S. Lueker, Testing for the Consecutive Ones Property, interval graphs and graph planarity using PQ-tree algorithm, *Journal of Computer and System Sciences* **13**, 335–379 (1976).

[2] Çela, E., V. Deineko & G.J. Woesinger, Recognising permuted Demidenko matrices, *Operations Research Letters*, **51**, 494–500 (2023).

[3] Deineko, V., B. Klinz, A. Tiskin & G.J. Woesinger, Four-point conditions for the TSP : The complete complexity classification, *Discrete optimization* **14**, 147–159 (2014).

[4] Diatta, J., Dissimilarités multivoies et généralisations d'hypergraphes sans triangles, *Mathématiques Informatique et Sciences humaines* **138**, 57–73 (1997).

[5] Diatta, J., Galois closed entity sets and k-balls of quasi-ultrametric multi-way dissimilarities, *Advances in Data Analysis and Classification* **1**, 53–65 (2007).

[6] Diday, E., Orders and overlapping clusters by pyramids, in *Multidimensionnal Data Analysis*, J. de Leeuw, W. Heiser, J. Meulman & F. Critchley Eds., 201–234 (1986).

[7] Durand, C. & B. Fichet, One-to-one correspondences inpyramidal representation : an unified approach, in *Classification and Related Methods of Data Analysis*, H.H. Bock Ed., 85–90 (1988).

[8] Heiser, W.J. & M. Bennani, Triadic distance models : axiomatization and least squares representation, *Journal of Mathematical Psychology* **41**, 189–206 (1997).

[9] Joly, S. & G. Le Calvé, Three-Way Distances, *Journal of Classification* **12**, 191–205 (1995).

[10] Préa, P. & D. Fortin, An optimal algorithm to recognize Robinsonian dissimilarities, *Journal of Classification* **31**, 351–385 (2014).

[11] Robinson, W.S., A method for chronologically ordering archeological deposits, *American Antiquity* **16**, 293–301 (1951).

[12] Warrens, M. J. & W. J. Heiser, Robinson Cubes, in *Selected Contributions in Data Analysis and Classification*, P. Brito, P. Bertrand, G. Cucumel & F. De Carvalho Eds, 515–523 (2007).

Kendall's tau and copula-based active learning algorithm

Chourouk Elokri[1], Tayeb Ouaderhman[1], Hasna Chamlal[1] ...

[1] Computer Science and Systems Laboratory (LIS), Department of Mathematics and Informatics, Faculty of Sciences Ain Chock, Hassan II University of Casablanca, Morocco

April 26, 2025

Résumé

L'apprentissage supervisé est une technique puissante qui implique la construction d'un modèle à partir de données annotées pour prendre des décisions sur de nouvelles données. Cependant, la disponibilité des données annotées n'est pas toujours garantie, car la rareté des données étiquetées et l'abondance des données non étiquetées sont courantes dans de nombreux domaines, ce qui entrave l'utilisation fiable des modèles supervisés. Pour relever ce défi, l'apprentissage actif est une méthode qui sélectionne les instances les plus informatives à partir des données non étiquetées, les présente à un expert humain pour qu'il les étiquette et les incorpore à l'ensemble de données étiquetées pour renforcer l'apprentissage du modèle. Les approches existantes de la sélection d'instances se concentrent souvent sur l'amélioration d'un modèle pré-spécifié, ce qui peut introduire des biais, tandis que les méthodes d'apprentissage actif cherchent à intégrer les instances les plus incertaines pour enrichir la distribution des données étiquetées. Cet article propose une approche alternative, où le processus d'apprentissage actif est traité comme une étape de prétraitement, intégrant les instances les plus informatives avant que le modèle ne soit appris. L'idée est de sélectionner les instances présentant le plus grand déséquilibre par rapport à l'ordre des caractéristiques initialement déterminées, en se basant uniquement sur les données étiquetées. Les fonctions de corrélation de rang de Kendall et de copule sont utilisées pour identifier ce déséquilibre. Pour évaluer l'efficacité de la méthode, le document évalue les performances de divers modèles statistiques, tels que la régression linéaire, les réseaux neuronaux et les machines à vecteurs de support, formés sur l'ensemble de données sélectionné, à l'aide des mesures de performances telles que: l'accuracy et le F1-Score.

Mots-clés

L'apprentissage supervisé, L'apprentissage actif, Fonction copule, Kendall, Poids des variables, Stratégie de sélection.

Abstract

Supervised learning is a powerful technique that involves constructing a model from annotated data to make decisions on new data. However, the availability of annotated data is not always guaranteed, as the scarcity of labeled data and the abundance of unlabeled data are common in many domains, hindering the reliable use of supervised models. To address this challenge, active learning is a method that selects the most informative instances from the unlabeled data, presents them to a human expert for labeling, and incorporates them into the labeled dataset to strengthen the model's learning. Existing approaches to instance selection often focus on improving a pre-specified model, which can introduce bias, while active learning methods seek to integrate the most uncertain instances to enrich the distribution of labeled data. This paper proposes an alternative approach, where the active learning process is treated as a preprocessing step, integrating the most informative instances before the model is learned. The idea is to select the instances with the greatest imbalance with respect to the order of the features initially determined, based solely on the labeled data. The Kendall rank correlation and copula functions are used to identify this imbalance. To assess the effectiveness of the method, the paper evaluates the performance of various statistical models, such as linear regression, neural networks and support vector machines, trained on the selected dataset, using measures such as accuracy and F1 score.

Keywords

Supervised learning , Active learning , Copula , Kendall , Feature weighting , Query strategy.

1 Introduction

Artificial intelligence has experienced substantial growth across its various subfields. Machine learning, the science of enabling machines to learn from existing data to inform subsequent decisions, has allowed machines to intervene and solve problems without the need for direct human involvement in numerous sectors, although human participation remains essential for its development [16].

Supervised learning, a domain within machine learning, is an area where humans must prepare the data before training and using it to make decisions. Human intervention is crucial in this case to annotate as much of the data as possible [22].

In many sectors, labeled data is scarce, and annotating unlabeled data is laborious, so passively selecting unlabeled

data to integrate with labeled data may not yield significant benefits [2].

Active learning is a technique that aims to select the unlabeled instances that provide the most information to the labeled data set by optimizing the cost and time of annotating the instances [11]. Multiple active learning strategies seek to maximize the performance of one or more models by selecting the most significant batch of instances and adding them to the initially labeled dataset [18]. These techniques enable the reliable and optimal utilization of a combination of labeled and unlabeled data.

Feature selection, on the other hand, is a field developed to minimize the time and cost of learning. It aims to select non-redundant and relevant features based on specific criteria. There are three types of feature selection: filter methods, wrapper methods, and embedded approaches. [1].

Filter methods assess the relevance of features independently of the model, using information theory or statistical measures. This involves assigning each feature a weight, ordering them based on their weights, and selecting those with the maximum weight [21].

In this article, the active learning method employs the order of the feature weights to select the instances that provide the most information to the labeled data set. The instances selected by this method represent the most uncertain instances, the ones that, when assigned to the initial labeled dataset, order the features as differently as possible under different labels.

To measure the difference in order, this article utilizes the Relief feature selection method [25] to determine the weights, the Kendall rate [5] to measure the noise contributed by each instance associated with a label in the class, and the copula function [8] to assess the uncertainty between the different labels.

The key contributions of this article are as follows:

- An approach that optimizes the allocation of human and computational resources by prioritizing informative data, enabling more effective utilization of the abundant unlabeled data available.
- The proposed method selects the most informative instances irrespective of the model, and focuses on enhancing the labeled dataset without pre-specifying the model.
- The method can be considered a preprocessing step that provides a dataset ready for learning by the chosen model, thereby optimizing the learning cost and time.

The paper is structured as follows: in the second section, a concise overview of existing approaches that we use to develop the proposed method is provided, in the third section, the method introduced by this work is described, in the fourth section, experimental considerations are discussed, and a conclusion along with future research directions are presented in the final section.

2 Preliminaries

This section outlines the primary methodology that enabled us to define our proposed active learning approach.

2.1 Relief and ReliefF algorithm

The active learning method proposed in this article uses the Relief technique introduced by Kira[13] to calculate feature weights. The main idea behind Relief is to estimate the quality of features in terms of their ability to distinguish between instances of different classes. This involves calculating a weight for each feature, reflecting its relevance to the classification task. These weights are then used to rank the features, and the best ranked features are selected. It should be noted that the motivation for using Relief as part of the weight calculation framework for active learning is the fact that it uses Euclidean distance for numeric features and Hamming distance for non-numeric features [26]. The most important strength of the Relief algorithm is that it is an agnostic technique, meaning that it can be used with a variety of machine learning algorithms and is not tied to any specific model [27]. Releif is also robust to noisy data because it takes into account the difference between instances of different classes [6].

Algorithm 1: Relief pseudo code (Feature Weighting using Nearest Neighbors)

Input: Dataset with n samples
Output: Weights $W[a]$ for each feature a

foreach *feature* a **do**
  $W[a] \leftarrow 0$;
for $i \leftarrow 1$ **to** n **do**
  Select a random sample s_i from the dataset;
  Find nearest hit s_h (same class);
  Find nearest miss s_m (different class);
  foreach *feature* a **do**
    $\Delta W_i[a] \leftarrow \text{diff}(a, s_i, s_m) - \text{diff}(a, s_i, s_h)$;
    $W[a] \leftarrow W[a] + \Delta W_i[a]$;
foreach *feature* a **do**
  $W[a] \leftarrow \frac{W[a]}{n}$;
where $\text{diff}(a, s_i, s_j) = \begin{cases} 0 & \text{if } s_i[a] = s_j[a] \\ 1 & \text{if } s_i[a] \neq s_j[a] \end{cases}$

The ReliefF algorithm, introduced by Kononenko (1994) [14], improves upon the original Relief method by addressing its limitations. It is designed to handle multi-class classification problems, is more robust to noise, and can deal effectively with incomplete data.

As in the original Relief approach, the algorithm begins by randomly selecting an instance s_i from the dataset. It then finds k nearest neighbors s_{h_j} that share the same class as s_i (called *nearest hits*) and, for each class $C \neq \text{class}(s_i)$, it selects k nearest neighbors $s_{m_j}^{(C)}$ (called *nearest misses*).

The relevance weight $W[a]$ of each feature a is updated based on the differences in feature values between s_i and

the selected neighbors. Specifically, the algorithm averages the contributions of all misses and all hits. The update rule is a generalization of the original Relief formula and is computed as follows:

- The average contribution from hits s_{h_j} is subtracted.
- The contribution of each miss $s_{m_j}^{(C)}$ is weighted by the prior probability $P(C)$ of its class C.

To ensure that the updates remain normalized and symmetric within the interval $[0, 1]$, the weights assigned to the misses are rescaled so that they sum to one. Since the class of s_i is excluded from the sum, the normalization factor becomes $1 - P(\text{class}(s_i))$.
This process is repeated for m randomly selected instances. The parameter k controls the locality of the neighborhood and can be set to $k = 10$ in most practical scenarios, as suggested by Kononenko.
To handle missing values, the difference function $\text{diff}(a, s_i, s_j)$ is redefined based on class-conditional probabilities. If the value of feature a is missing for s_i but known for s_j, then:

$$\text{diff}(a, s_i, s_j) = 1 - P(s_j[a] \mid \text{class}(s_i))$$

If the value is missing for both s_i and s_j, the difference becomes:

$$\text{diff}(a, s_i, s_j) = 1 - \sum_{v \in \text{values}(a)} P(v \mid \text{class}(s_i)) \cdot P(v \mid \text{class}(s_j))$$

The probabilities $P(v \mid \text{class})$ are estimated empirically from the training data using relative frequencies.

2.2 Kendall measurement

Kendall's τ rank correlation coefficient is a widely used non-parametric measure for assessing the strength and direction of association between two ranked variables. It is particularly appreciated for its ability to detect monotonic relationships, even when the underlying connection is non-linear, and it remains reliable in the presence of outliers [28]. Given two sequences of observations, Kendall's τ evaluates the concordance and discordance between all possible pairs:

$$\tau = \frac{(N_c - N_d)}{\frac{1}{2}n(n-1)}$$

where N_c is the number of concordant pairs, N_d is the number of discordant pairs, and n is the total number of observations. This coefficient provides a robust estimation of similarity in the ordering of values between two variables.

2.3 Copula function

Copula theory offers a powerful and flexible framework for modeling dependence structures between random variables, especially in high-dimensional settings where traditional correlation measures often fall short [17]. Sklar's Theorem (1959) [23] forms the theoretical foundation of this approach, stating that any multivariate joint distribution can be uniquely decomposed into its marginal distributions and a copula function that captures the dependence among variables [8] Formally, for a random vector $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ with continuous marginal distribution functions $F_1, F_2, \ldots, F_n$, there exists a unique copula C such that:

$$F(x_1, x_2, \ldots, x_n) = C(F_1(x_1), F_2(x_2), \ldots, F_n(x_n))$$

This decomposition isolates the dependence structure in the copula C, allowing it to be studied independently of the marginals. Copulas are particularly well-suited for capturing non-linear and tail dependencies, which are common in real-world data but typically overlooked by linear measures such as Pearson or Kendall correlations. In recent studies, copula-based methods have been successfully applied in various domains, including machine learning, finance, and bioinformatics, to model complex multivariate dependencies (see [10], [19], [12]). Their ability to describe intricate joint behaviors makes them a valuable tool in modern statistical modeling and feature selection processes.

Gaussian Copula

Among the most widely used copulas in practice is the Gaussian copula [29], primarily due to its mathematical simplicity and compatibility with the multivariate normal distribution. The Gaussian copula is derived from a multivariate normal distribution and captures linear dependence structures through a correlation matrix, without requiring the margins themselves to be Gaussian.
Formally, let $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ be a random vector following a multivariate normal distribution with zero mean and correlation matrix $\boldsymbol{R} \in \mathbb{R}^{n \times n}$. The Gaussian copula C_{Gauss} associated with $\boldsymbol{R}$ is defined as:

$$C_{\text{Gauss}}(u_1, \ldots, u_n) = \Phi_{\boldsymbol{R}}\left(\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_n)\right)$$

where Φ^{-1} denotes the inverse cumulative distribution function (quantile function) of the standard univariate normal distribution, and $\Phi_{\boldsymbol{R}}$ is the joint CDF of the n-dimensional normal distribution with correlation matrix $\boldsymbol{R}$ and standard normal margins.
This construction allows the Gaussian copula to separate the modeling of dependence from the modeling of marginals. That is, the marginal distributions $F_1, \ldots, F_n$ of each variable can be arbitrary and modeled independently, while the copula captures the dependence structure through $\boldsymbol{R}$. This makes the Gaussian copula particularly attractive in fields like finance, insurance, and machine learning, where the variables may exhibit non-Gaussian behavior individually, yet share an approximately Gaussian dependence pattern [15].
The Gaussian copula remains a foundational and computationally efficient tool for modeling multivariate

dependence, particularly when linear relationships dominate or when a first approximation of complex dependence is needed.

Algorithm 2: Calculate Dependencies Using Gaussian Copula

Input: A dataset $\{s_1, s_2, \ldots, s_m\}$ with m samples and n variables, correlation matrix $\boldsymbol{R}$

Output: Copula-based dependence structure between variables

foreach *sample* s_i *in the dataset* **do**
  foreach *variable* X_j *in* s_i **do**
    Compute the empirical marginal distribution function F_j for X_j;
  foreach *pair of variables* (X_j, X_k) **do**
    Apply the inverse CDF of the standard normal to the empirical marginal distributions:

$$u_j = \Phi^{-1}(F_j(x_j)), \quad u_k = \Phi^{-1}(F_k(x_k))$$

  Compute the Gaussian copula for s_i using the formula:

$$C_{\text{Gauss}}(u_1, \ldots, u_n) = \Phi_{\boldsymbol{R}}\left(\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_n)\right)$$

return *The dependence structure encapsulated in the Gaussian copula.*

3 Method

The active learning approach described in this paper combines the strengths of three key techniques:

- Relief: used to calculate feature weights and rank them based on their relevance [25].
- Kendall's tau: employed to measure the agreement between the weight vector derived from the initially labeled data and the weight vector after incorporating an unlabeled instance assigned to a label [28].
- Gaussian Copula: utilized to identify the linear relationship among the different Kendall's tau values applied to the same unlabeled instance assigned to various classes [29].

The active learning strategy then selects the instance with the smallest value in the vector rotated by the copula function.

4 Perspectives and experiments

Dans cette section, nous introduisons les différents ensemble de données envisagée pour l'étude de cette approche, les modèles utilisés pour visualiser les résultats, et les mesures de performances adoptées.

4.1 Datasets

The various datasets used to experiment with this method are described in Table 1. Table 1 includes four datasets, sourced from the UCI Machine Learning Repository [4]. These datasets are a mix of binary and multi-class classification problems, which allows for a comprehensive evaluation of the method across diverse data types. This diversity ensures that the method is robust and can be evaluated under different conditions, making it possible to assess its generalizability to various types of classification tasks.

4.2 Performance Measurements

The performance metrics used in this study are explained below. These metrics were selected based on their ability to provide a comprehensive evaluation of the classification models, particularly in cases where class imbalance or non-ideal classification scenarios may exist.

- **Accuracy** [20]: This metric measures the proportion of correct predictions (both true positives and true negatives) out of all predictions made. It is a commonly used metric for evaluating classification models. However, accuracy can be misleading when the dataset is imbalanced, as it does not take the distribution of the classes into account. Despite this, accuracy remains useful in situations where the classes are evenly distributed and when we aim for a general measure of model effectiveness.
- **F1-Score** [24]: The F1-Score is the harmonic mean of precision and recall, providing a balanced evaluation of a model's ability to correctly classify instances of the positive class. This metric is particularly useful in the case of imbalanced datasets, where one class is much more prevalent than the other. F1-Score ensures that both false positives and false negatives are penalized, giving a more nuanced understanding of model performance in such scenarios.

Both metrics were chosen because they address different aspects of model performance. Accuracy is straightforward and provides a quick overview of a model's performance, while the F1-Score provides a more detailed picture, especially in the case of imbalanced data or when the costs of false positives and false negatives are significant.

4.3 Models

The models used for the experiments are as follows:

- **Logistic regression** [9]: is a widely used statistical model for binary classification tasks. It estimates the probability that an instance belongs to a particular class based on input features. It is simple, interpretable, and effective, especially when the relationship between the input features and the target variable is linear. Logistic regression serves as a baseline model for comparison in the experiments.

Dataset Name	Type of Classes	Number of Features	Number of Instances
Iris	Multi-class	4	150
Breast Cancer	Binary	30	569
Diabetes	Binary	8	768
Heart Disease	Binary	13	303

Table 1: Datasets for experiments

- **Support Vector Machine (SVM)** [3]: is a powerful classifier that works well on both linear and non-linear problems by finding an optimal hyperplane that separates the classes in the feature space. It is known for its effectiveness in high-dimensional spaces and for its robustness against overfitting, especially when combined with the appropriate kernel. SVM is included to evaluate the method's performance on more complex, non-linear decision boundaries.
- **Neural Networks** [7]: are highly flexible models that can capture complex, non-linear relationships between features and the target variable. They are especially useful for large datasets with intricate patterns, such as those found in image recognition or speech processing. In this experiment, neural networks are included to evaluate how the method performs when applied to more advanced, deep learning models, which are capable of modeling high-level abstractions in the data.

Each model was chosen to provide a comprehensive comparison, covering a range of simple to more complex approaches for binary and multi-class classification. The choice of these models ensures that the method can be evaluated across different levels of model complexity.

5 Conclusion and futur work

In this study, we presented an instance selection strategy for active learning that operates independently of the classification model, making it suitable as a preprocessing step prior to training. The proposed method is grounded in a combination of three complementary approaches: first, Relief is used to rank features according to their relevance, helping to prioritize features that contribute most to class discrimination. Second, Kendall's rank correlation coefficient is applied to capture concordance relationships between variables, which is particularly valuable in detecting non-linear dependencies that often go unnoticed by linear measures. Finally, we leverage the copula function, known for its strong theoretical foundations in statistical dependence modeling, to identify the most informative unlabeled instances within the dataset. This step is essential for effectively guiding the labeling process while minimizing annotation effort.
The integration of these three methods allows the system to capture different aspects of the data structure: feature importance, inter-variable dependencies, and global multivariate relationships. Our experiments on several datasets—binary and multi-class—demonstrated that this approach can effectively reduce the number of labeled instances needed to reach high predictive performance. Because the method is model-agnostic, it can be applied across various learning algorithms, such as logistic regression, SVMs, and neural networks.
As promising as these results are, future work may enhance the method in several directions. One avenue involves refining the instance selection process by incorporating dynamic query strategies based on uncertainty, diversity, or expected model change. Another promising direction is to extend the framework to streaming environments or multi-label settings, where data is continuously generated and labeling challenges are amplified. Furthermore, the scalability of the approach could be improved through parallel processing or sampling-based approximations, especially for large-scale, high-dimensional datasets. By pursuing these enhancements, the proposed method could become a robust and flexible tool for real-world active learning applications across various domains.

References

[1] Younes Benhlima, Younes Amahjour, and Said Abou El Haj. Feature selection: A review and comparative study. In *E3S Web of Conferences*, volume 351, page 01046. EDP Sciences, 2022.

[2] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. MIT Press, 2006.

[3] Corinna Cortes and Vladimir Vapnik. *Support-vector networks*, volume 20. Springer, 1995.

[4] Dheeru Dua and Casey Graff. Uci machine learning repository. https://archive.ics.uci.edu/ml/index.php, 2019. Accessed: 2024-04-26.

[5] T Espana, V Le Coz, and M Smerlak. Kendall correlation coefficients for portfolio optimization. *arXiv preprint arXiv:2410.17366*, 2024.

[6] Salvador García, Julián Luengo, and Francisco Herrera. *Data Preprocessing in Data Mining*. Springer, 2015.

[7] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[8] Stefan Größer. Copulae: An overview and recent developments. *Wiley Interdisciplinary Reviews: Computational Statistics*, 14(2):e1557, 2022.

[9] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2nd edition, 2009.

[10] Harry Joe. *Dependence Modeling with Copulas*. CRC Press, 2014.

[11] Tejaswi Kasarla, Abhishek Jha, Faye Tervoort, Rita Cucchiara, and Pascal Mettes. Maximally separated active learning. *arXiv preprint arXiv:2411.17444*, 2024.

[12] Malte Killiches, Daniel Kraus, and Claudia Czado. Modeling high-dimensional dependencies using vine copulas. *Computational Statistics*, 36:1577–1613, 2021.

[13] Kenji Kira and Larry A. Rendell. The feature selection problem: Traditional methods and a new algorithm. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 129–134. AAAI Press, 1992.

[14] Igor Kononenko. Estimating attributes: Analysis and extensions of relief. In *Machine Learning: ECML-94*, pages 171–182. Springer, 1994.

[15] X. Liu, Y. Zhang, and Z. Wang. Gaussian copula embeddings. In *Advances in Neural Information Processing Systems*, volume 35, pages 12345–12356, 2022.

[16] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[17] Roger B. Nelsen. *An Introduction to Copulas*. Springer Science & Business Media, 2nd edition, 2006.

[18] Yuta Ono, Till Aczel, Benjamin Estermann, and Roger Wattenhofer. Supclust: Active learning at the boundaries. *arXiv preprint arXiv:2403.03741*, 2024.

[19] Andrew J Patton. A review of copula models for economic time series. *Journal of Multivariate Analysis*, 110:4–18, 2012.

[20] David M. W. Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2:37–63, 2011.

[21] Mehrdad Rostami, Kamal Berahmand, and Saman Forouzandeh. Review of swarm intelligence-based feature selection methods. *arXiv preprint arXiv:2008.04103*, 2020.

[22] Burr Settles. *Active Learning*, volume 6 of *Synthesis Lectures on Artificial Intelligence and Machine Learning*. Morgan & Claypool Publishers, 2012.

[23] Abraham Sklar. Fonctions de répartition à n dimensions et leurs marges. *Publ. Inst. Statist. Univ. Paris*, 8:229–231, 1959.

[24] Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437, 2009.

[25] Ryan J Urbanowicz, Melissa Meeker, William LaCava, Randal S Olson, and Jason H Moore. Relief-based feature selection: Introduction and review. *Journal of Biomedical Informatics*, 85:189–203, 2018.

[26] Ryan J Urbanowicz, Michael Meeker, William LaCava, Randal S Olson, and Jason H Moore. Relief-based feature selection: Introduction and review. *Journal of Biomedical Informatics*, 85:189–203, 2018.

[27] Ryan J Urbanowicz, Randal S Olson, Patrick Schmitt, Michael Meeker, and Jason H Moore. Benchmarking relief-based feature selection methods for bioinformatics data mining. *Journal of Biomedical Informatics*, 85:168–182, 2018.

[28] Dalia Valencia, Rosa E. Lillo, and Juan Romo. A kendall correlation coefficient between functional data. *Advances in Data Analysis and Classification*, 13(4):1083–1103, 2019.

[29] Ke Wan and Alain Kornhauser. On the properties of gaussian copula mixture models. *arXiv preprint arXiv:2305.01479*, 2023.

Vers une meilleure exploitation du clustering textuel : clustering pondéré et LLM

A. Ferdjaoui[1,2], S. Affeldt[2], M. Nadif[2]

[1] SogetiLabs, Capgemini, France
[2] Centre Borelli UMR 9010, Université Paris Cité, France

amine.ferdjaoui@etu.u-paris.fr | severine.affeldt@u-paris.fr | mohamed.nadif@u-paris.fr

Résumé

Dans le domaine de l'apprentissage non supervisé, l'analyse de vastes quantités de données textuelles est un sujet qui attire beaucoup d'attention. Dans cet article, nous présentons une application facile à utiliser pour explorer de grands volumes de données textuelles en utilisant le clustering et les modèles génératifs. Nous démontrons comment adapter l'algorithme Lasso-Weighted k-means pour traiter les données textuelles. De plus, nous présentons en détail un package très facile à utiliser qui montre comment exploiter efficacement les LLM pour décrire les clusters de documents.

Mots-clés

Clustering, données textuelles, pondération, exploration de clusters.

Abstract

In unsupervised learning, the exploration of large volumes of textual data is a topic of significant interest. In this article, we present our compact and easy-to-use application to explore large volumes of textual data using clustering and generative models. We demonstrate how to adapt the Lasso weighted k-means algorithm to handle textual data. In addition, we present in detail a user-friendly package that shows how to use LLMs effectively to describe document classes.

Keywords

Clustering, Text Mining, Feature Weighting, Cluster Interpretation.

1 Introduction

La classification non supervisée, ou *clustering*, est un sujet de grand intérêt et une démarche cruciale pour l'analyse de grandes quantités de données textuelles. De nombreuses méthodes ont été proposées, chacune présentant ses propres avantages et spécificités, en fonction du type de données. Néanmoins, l'interprétation des résultats reste l'un des principaux défis. Récemment, le clustering pondéré a suscité beaucoup d'intérêt. Il consiste à attribuer des poids aux variables, afin de mieux capturer les caractéristiques distinctives des objets, ce qui permet d'améliorer l'interprétation des clusters. Cependant, peu de méthodes de clustering pondéré se concentrent sur les données textuelles. Dans cet article, nous proposons un nouveau cadre d'exploration, basé sur un algorithme de clustering pondéré, adapté aux données textuelles. L'algorithme proposé est intégré dans un package Python, fournissant une application web intelligible et personnalisable pour l'analyse et l'exploration de grands volumes de données textuelles.

2 Travaux Connexes

Le clustering est un sujet de recherche de longue date, ayant conduit au développement de nombreuses techniques au cours des dernières années. Malgré la diversité des méthodes disponibles, l'algorithme k-means [9, 8] reste un choix populaire et simple pour le clustering. De nombreuses variantes ou extensions des k-means ont été développées, notamment pour traiter des données sparses. On peut citer le *sparse clustering* avec pondération [14, 7, 3] ou encore le co-clustering [1, 12, 11], ainsi que les modèles d'apprentissage profond [10, 2].

Les techniques de *sparse clustering* excellent dans l'identification des caractéristiques les plus significatives d'une matrice sparse. Witten et Tibshirani [14] ont introduit l'une des premières versions parcimonieuses utilisant la régularisation ℓ_1 et ℓ_2 pour la sélection de variables, connue sous le nom d'algorithme Sparse k-means. Des travaux ultérieurs ont poursuivi cette approche de régularisation, notamment avec *Robust and Sparse k-means Clustering* (RSKC) [3].

Plus récemment, Chakraborty et Das [15, 4] ont introduit un cadre de *sparse clustering* plus simple, basé sur un algorithme k-means pondéré, nommé Lasso Weighted k-means (Lwk). Cette méthode applique directement une pénalisation Lasso sur les poids des variables, permettant une solution analytiuqe pour la mise à jour des poids. L'algorithme Lwk a démontré son efficacité par rapport aux autres approches de l'état de l'art, tout en conservant une simplicité et une efficacité computationnelle.

3 Variante pondérée de K-means

L'algorithme Lwk offre une méthode simple et robuste basée sur l'algorithme k-means pour calculer les poids. Soit $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ une matrice de taille $n \times p$ que nous

souhaitons partitionner en g clusters. L'algorithme k-means classique minimise la variance intra-cluster à travers la fonction objectif :

$$f_{\text{k-means}}(\Theta) = \sum_{i=1}^{n}\sum_{k=1}^{g} u_{ik}\|\mathbf{x}_i - \theta_k\|_2^2 \tag{1}$$

où $\Theta = \{\theta_1, \ldots, \theta_g\}$ représente l'ensemble des g centroïdes, la matrice $\mathbf{U}$ contient les éléments $u_{ik} \in \{0,1\}$ satisfaisant $\sum_{k=1}^{g} u_{ik} = 1$, indiquant si le point i est le plus proche du centre θ_k, et $\|.\|_2$ est la norme Euclidienne classique donnée par : $\|\mathbf{x}_i\|_2^2 = \sum_{j=1}^{p} x_{ij}^2$.
L'algorithme Lwk [15, 4] est formulé comme une minimisation alternée de la fonction objectif suivante :

$$f_{\text{LWK-means}}(\Theta, \mathbf{w}) = \sum_{i=1}^{n}\sum_{k=1}^{g} u_{ik}\left\{\|\mathbf{x}_i - \theta_k\|_{\mathbf{w}^\beta}^2 + \lambda\|\mathbf{x}_i - \theta_k\|_{|\mathbf{w}|}^2\right\} - \alpha\mathbf{w}^\top\mathbb{1}_p \tag{2}$$

où $\|\mathbf{y}\|_{|\mathbf{w}|}^2 = \sum_{j=1}^{p}|w_j|y_j^2$, $\mathbf{w} = [w_1, w_2, ..., w_p]$ est le vecteur de poids des p variables, et $\lambda > 0$, $\alpha > 0$, $\beta > 1$ sont des paramètres fixés par l'utilisateur. Cette fonction est minimisée par rapport à Θ et $\mathbf{w}$. L'algorithme Lwk se distingue des autres algorithmes de pondération par l'ajout d'un terme qui pénalise les variables à forte variance. Pour illustrer cela, nous définissons :

$$D_j = \sum_{i=1}^{n}\sum_{k=1}^{g} u_{ik}(x_{ij} - \theta_{kj})^2$$

qui représente la dispersion au sein du cluster le long de la variable j. Une valeur élevée de D_j indique que la variable j a une grande variance et n'est pas utile pour le regroupement, car les données ne sont pas homogènes au sein de la partition. Fixer Θ dans l'équation 2 :

$$f(w) = \sum_{j=1}^{p} w_j^\beta D_j + \lambda\sum_{j=1}^{p}|w_j|D_j - \alpha\mathbf{w}^\top\mathbb{1}_p \tag{3}$$

Le premier terme, $\sum_{j=1}^{p} w_j^\beta D_j$, représente un clustering pondéré standard avec un vecteur de poids $\mathbf{w}$ de taille p. Cependant, l'algorithme diffère en incluant un second terme, $\lambda\sum_{j=1}^{p}|w_j|D_j$, qui pénalise les variables à forte variance. Comme l'algorithme n'utilise pas d'autres formes de régularisation, le terme $\alpha\mathbf{w}^\top\mathbb{1}_p$ est ajouté pour éviter les solutions dégénérées où $\mathbf{w} = 0_p$.
Suivant le cadre LWK, la minimisation de ce problème est donnée par la résolution des trois problèmes de minimisation suivants :

Problème P1

Étant donné $\mathbf{\Theta} = \mathbf{\Theta_0}$ et $\mathbf{w} = \mathbf{w_0}$ fixés, minimiser $f(\mathbf{U}, \mathbf{\Theta_0}, \mathbf{w_0})$ par rapport à $\mathbf{U}$:

$$u_{ik} = \begin{cases} 1, & \text{si } k = \arg\min\limits_{1\le k\le g}\sum\limits_{j=1}^{p}(w_j^0 + \lambda|w_j|)(x_{ij} - \theta_{kj})^2, \\ 0, & \text{sinon.} \end{cases}$$

Problème P2

Étant donné $\mathbf{U} = \mathbf{U_0}$ et $\mathbf{w} = \mathbf{w_0}$ fixés, minimiser $f(\mathbf{U_0}, \mathbf{\Theta}, \mathbf{w_0})$ par rapport à $\mathbf{\Theta}$:

$$\theta_k = \frac{\sum_{i=1}^{n} u_{ik}\mathbf{x}_i}{\sum_{i=1}^{n} u_{ik}} \quad \forall k \in \{1, \ldots, g\}.$$

Problème P3

Étant donné $\mathbf{\Theta} = \mathbf{\Theta_0}$ et $\mathbf{U} = \mathbf{U_0}$ fixés, minimiser $f(\mathbf{U_0}, \mathbf{\Theta_0}, \mathbf{w})$ par rapport à $\mathbf{w}$. La solution $\mathbf{w}^*$ est donnée par :

$$w_j^* = \left[\frac{1}{\beta}S\left(\frac{\alpha}{D_j}, \lambda\right)\right]^{\frac{1}{\beta-1}}, \quad \forall j \in \{1, \ldots, p\},$$

où $S(x, \gamma)$ est la fonction de soft-threshold définie comme :

$$S(x,\gamma) = \begin{cases} x - \gamma, & \text{si } x \ge \gamma, \\ x + \gamma, & \text{si } x \le -\gamma, \\ 0, & \text{sinon.} \end{cases}$$

3.1 Régularisation proposée pour les données textuelles

Les algorithmes de clustering pondérés doivent leur efficacité à leur capacité à identifier les caractéristiques clés. En particulier, la grande supériorité de l'algorithme Lwk repose à la fois sur sa rapidité – il reproduit la complexité de k-means – et sur la simplicité de sa fonction objective. Cependant, l'approche Lwk a principalement été testée sur des données génomiques qui sont généralement denses, tandis que les données textuelles, sous forme de matrices document-terme, sont très sparses et souvent déséquilibrées. Nous proposons ci-dessous une régularisation adaptée aux données textuelles.

3.1.1 Défi de la sparsité

Le K-means sphérique, souvent appelé S-Kmeans, est l'une des méthodes de clustering basées sur K-means les plus adaptées pour traiter les données textuelles organisées sous forme de matrices document-terme [1]. S-Kmeans est particulièrement bien adapté aux données de grande dimension et sparses, en particulier lorsque les données se trouvent sur une hypersphère unitaire. Ce scénario est courant en fouille de textes et en traitement automatique du langage naturel, où les documents sont fréquemment représentés sous forme de vecteurs de longueur unitaire.
Appliquer cette normalisation à l'algorithme Lwk implique d'intégrer une normalisation par ligne. Cependant, cela introduit un autre problème : la distribution des données à travers les colonnes. L'algorithme Lwk, via son deuxième terme de régularisation $\lambda\sum_{j=1}^{p}|w_j|D_j$, recherche les variables présentant des zones denses et renforce ainsi leur poids. Par conséquent, bien que cela ne soit pas explicitement mentionné par les auteurs, la normalisation est cruciale pour les performances de l'algorithme. Bien que la normalisation ℓ_2 projette les données sur une sphère unitaire, elle ne préserve pas la normalité au sein de chaque

variable. Cela suggère que la double normalisation, à la fois en ligne et en colonne, serait plus appropriée.
L'une des métriques les plus adaptées dans ce contexte est la métrique du chi-2 ; voir par exemple [6]. Lorsqu'elle est calculée entre deux documents, elle prend en compte non seulement le poids de chaque document $x_{i.} = \sum_{j=1}^{p} x_{ij}$ mais aussi celui de chaque mot $x_{.j} = \sum_{i=1}^{n} x_{ij}$. La distance du chi-2 entre deux documents est donnée par :

$$\begin{aligned} d_{\chi^2}(x_i, x_{i'}) &= \sum_j \frac{1}{x_{.j}} \left(\frac{x_{ij}}{x_{i.}} - \frac{x_{i'j}}{x_{i'.}} \right)^2 \\ &= \sum_j \left(\frac{x_{ij}}{x_{i.}\sqrt{x_{.j}}} - \frac{x_{i'j}}{x_{i'.}\sqrt{x_{.j}}} \right)^2 \end{aligned} \tag{4}$$

$$x_{ij} \leftarrow \frac{x_{ij}}{x_{i.}\sqrt{x_{.j}}}.$$

Cela implique d'adopter ce type de normalisation afin de rendre l'optimisation de (2) beaucoup plus réaliste pour le type de données que nous traitons.

3.1.2 Initialisation

En raison de la sparsité, l'initialisation de l'algorithme est cruciale. Nous adoptons une initialisation basée sur le partitionnement donné par l'algorithme S-Kmeans. Ce choix est justifié par l'intégration des vecteurs de documents dans une hypersphère de rayon 1. En d'autres termes, une seule normalisation des documents est effectuée, tandis que la métrique du chi-2 agit à la fois sur l'ensemble des documents et des mots.

3.2 Résultats de clustering

Afin de tester la variante pondérée de l'algorithme K-means, nommé VPK-means, nous évaluons d'abord les résultats de clustering de notre algorithme sur divers jeux de données de référence (Tableau 1), en le comparant à l'algorithme original LWK, K-means, S-Kmeans et Deep k-means (Tableau 2). Ensuite, nous effectuons une analyse exploratoire des poids afin d'étudier la structure et le contenu des clusters, que nous présentons dans la section suivante.
Les résultats du clustering sont évalués à l'aide de l'Information Mutuelle Normalisée (NMI) [13]. Nous constatons que l'algorithme R-LWK surpasse l'algorithme LWK original, montrant ainsi que l'adaptation proposée est efficace pour les données textuelles. Nous observons également que R-LWK surpasse l'algorithme K-means et en particulier S-Kmeans, qui est l'un des meilleurs algorithmes de clustering pour les textes.

4 Présentation de l'application web

Afin de faciliter l'utilisation de l'algorithme R-LWK, nous proposons un package python qui permet de réaliser le traitement des données textuelles, le clustering et l'exploration des résultats. Nous fournissons également une interface web, très facile à utiliser (Figure 1).

TABLE 1 – Résumé des jeux de données

Jeu de données	Documents	Mots	g	Sparsité (%)	Équilibre
Sports	8580	14870	7	99.14	0.03
TR45	690	8261	10	96.60	0.08
Classic4	7094	5896	4	96.60	0.32
CSTR	475	1000	4	96.60	0.39
OHSCALE	11162	11465	10	99.47	0.43
NG20	19949	43586	20	99.50	0.62
BBC news	2225	15555	5	97.67	0.75

TABLE 2 – Comparaison des résultats de clustering en terme NMI.

Jeu de données	VPK-means	Lwk	S-Kmeans	K-means	Deep KM
Sports	**0.65**	0.47	0.64	0.45	0.60
TR45	**0.70**	0.66	0.66	0.63	0.61
Classic4	**0.79**	0.69	0.68	0.55	0.67
CSTR	**0.76**	0.66	0.69	0.60	0.64
OHSCALE	**0.45**	0.42	0.43	0.37	0.38
NG20	**0.59**	0.47	0.55	0.39	0.49
BBC	**0.90**	0.78	0.87	0.71	0.76

4.1 Package python

Les trois principales fonctionnalités de notre package Python sont les suivantes :

- *Traitement des données textuelles* : tokenisation et nettoyage des données textuelles (chargées localement par l'utilisateur, ou sélectionnées à partir d'une liste de jeux de données de référence).
- *Clustering* : lancement de l'algorithme VPK-means, collecter les labels, les poids et les critères internes.
- *Exploration des résultats* : fonctions utilitaires permettant d'explorer les résultats du clustering (variables importantes, mots fréquents, scores de clustering, etc.).

4.2 Application web

L'application web comprend un pipeline pour générer et explorer les clusters prédits par l'algorithme VPK-means. Il permet d'analyser de grands volumes de données textuelles de manière simple, sans que l'utilisateur ne soit contraint d'effectuer une quelconque intervention dans le code. L'interface est divisée en trois parties, décrites dans les sections suivantes.

4.2.1 Paramètres

La première partie des paramètres concerne les données (Figure 1 - Data). Deux choix de données sont disponibles. Le premier est le chargement à partir de données de référence (*Benchmark*) déjà présentes en base de données. Un résumé du jeu de données choisi est fourni dynamiquement, les résultats de clustering sont affichés aussi en comparant les labels de référence aux labels prédits par le modèle. Le deuxième choix proposé est l'utilisation des articles de PubMed pour créer un corpus de données textuelles. L'utilisateur peut extraire des données de la base de données en ligne - avec plus de 37 millions de citations pour la littéra-

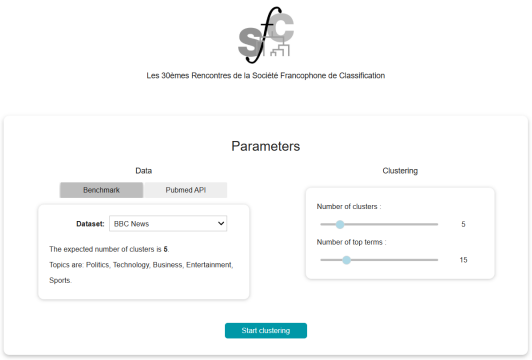

FIGURE 1 – Cluster Insight web application.

ture biomédicale - sur la base d'un mot-clé et du nombre de documents. Les données sont ensuite traitées avant de lancer le clustering, en procédant à une tokenisation et à un nettoyage. La deuxième partie des paramètres concerne les paramètres du modèle (Figure 1 - Clustering), à savoir le choix du nombre de clusters et le nombre de top termes à afficher.

4.2.2 Résultats de clustering

Des indicateurs globaux de clustering sont affichés à la convergence du modèle. Nous fournissons un tableau récapitulatif comprenant le nombre de documents, la taille du vocabulaire (nombre total de mots dans le corpus), ainsi que les scores de clustering NMI et ARI pour les jeux de données de référence.

L'interface propose également :

- un graphique représentant la proportion de documents et de termes (nombre unique de termes) par cluster,
- une vue d'ensemble en nuage de mots des thématiques de l'ensemble des clusters,
- et enfin, les mots les plus importants, correspondant aux variables ayant le poids le plus faible retourné par VPK-means.

4.2.3 Exploration des clusters

4.2.4 Paramètres

La section d'exploration est la partie la plus importante de l'application (Figure 2), car elle permet d'analyser les thématiques de chaque cluster et ainsi de mieux comprendre la manière dont les documents ont été regroupés. L'interface affiche d'abord un nuage de mots basé sur la matrice tf-idf des documents de chaque cluster. Ensuite, elle présente les termes les plus fréquents de ces matrices. La sélection des termes principaux de chaque cluster est guidée par les poids de VPK-means. Il est important de noter que les poids retournés étant globaux à l'ensemble du jeu de données, une pré-sélection des mots est effectuée pour les restreindre à ceux du cluster. Cette sélection est basée sur le score tf-idf des mots. Enfin, les mots sont organisés par ordre croissant de leur poids R-LWK.

L'application est personnalisable à l'aide d'un simple no-

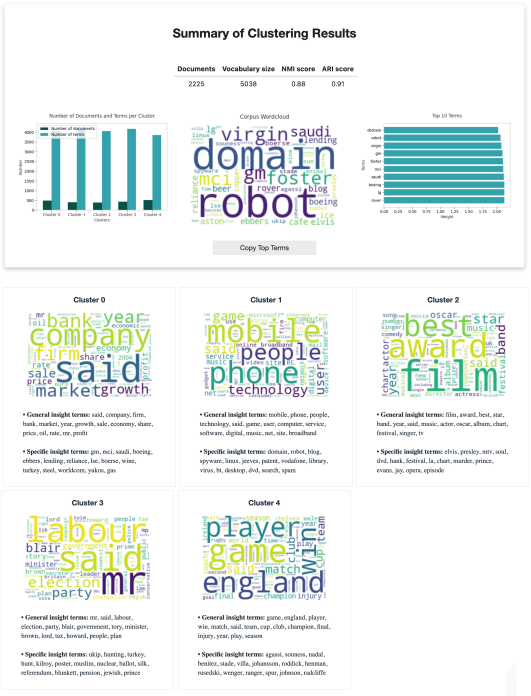

FIGURE 2 – Cluster Insight web application.

tebook jupyter et est dockerisé, ce qui permet de l'utiliser sans avoir à installer de bibliothèques ou à gérer des dépendances.

4.3 Application sur le jeu de données BBC News

Pour fournir un exemple concret de notre application, nous avons utilisé le célèbre jeu de données BBC News. Nous avons exécuté l'algorithme de clustering sur ce jeu de données, et les résultats sont présentés dans la Figure 2. Si nous regardons les termes généraux, nous pouvons voir qu'ils donnent une idée globale du contenu du cluster, ce qui nous aide à identifier le sujet principal. Par exemple, dans le premier cluster, des termes comme film, award et star indiquent que le cluster concerne le domaine du divertissement. En revanche, les termes spécifiques permettent d'identifier des sujets plus détaillés au sein des clusters, tels que des acteurs spécifiques, des sports ou des questions politiques. Cette analyse des top termes est utile pour découvrir les thèmes principaux, et l'ajout de termes supplémentaires permet d'approfondir les détails de chaque cluster. Cependant, ce processus peut devenir complexe avec un grand nombre de termes, en particulier en l'absence d'experts pour les interpréter. C'est pourquoi nous avons décidé d'utiliser des modèles génératifs pour assister cette tâche. La première approche consistait à fournir une liste de tous les termes principaux (généraux et spécifiques) afin d'aider le modèle à identifier les clusters. Nous avons testé cette approche avec GPT-4o, et les noms de clusters prédits étaient : Business/Economy, Technology/Gadgets, En-

tertainment/Awards, Politics/Government et Sports/Games. Ces labels correspondaient en fait aux labels d'origine et nous ont même donné une idée des sous-thèmes présents dans chaque cluster. La deuxième approche consistait à demander au modèle génératif d'expliquer un cluster spécifique. Nous avons testé cela sur le premier cluster, et le résultat suivant a été généré :

General Topic : Global Economic Trends and Corporate Performance
In recent years, major **companies** and **firms** have experienced fluctuating growth due to **market** conditions. **Boeing**, **Reliance**, and **Saudi oil** giants faced changes in **share prices** influenced by **oil** and **gas rates**. The **banking** sector, including **lending** institutions like **MCI** and **GM**, saw **profits rise** and **fall**, impacted by economic trends. High-profile **companies** like **WorldCom** and **Yukos** faced legal challenges, while **steel** producers and **wine** exporters, notably in **Turkey**, navigated shifting demand. Stock exchanges like the **LSE** and **Boerse** track these dynamics. Corporate leaders like **Mr. Ebbers** have been pivotal in shaping the **economy** through strategic decisions and acquisitions.

Cette analyse montre que le modèle génératif explique bien le cluster en faisant le lien entre les termes les plus importants et en expliquant chaque mot dans son contexte.

5 Conclusion

Dans cet article, nous avons présenté un outil permettant d'analyser et d'explorer de grands volumes de données textuelles à l'aide du clustering pondéré, en utilisant l'algorithme VPK-means que nous avons adapté aux données textuelles. À travers nos expériences, nous avons démontré que les améliorations apportées à l'algorithme Lwk original ont significativement amélioré les résultats du clustering. En exploitant les poids retournés par l'algorithme, nous sommes en mesure d'explorer les résultats du clustering de deux manières : d'une part, par l'analyse des termes les plus représentatifs, et d'autre part, en l'intégrant à un modèle génératif afin d'interpréter ces termes. Ce processus d'extraction de mots-clés est très prometteur et pourrait être étendu à d'autres applications, telles que l'extraction de relations causales entre termes, à l'image de *WordGraph* [5]. Enfin, nous identifions des pistes d'amélioration pour l'interface, notamment la possibilité de connecter directement les termes les plus représentatifs à *ChatGPT* via une API, permettant ainsi un affichage en temps réel des résultats d'analyse directement dans l'application.

Références

[1] Séverine Affeldt, Lazhar Labiod, and Mohamed Nadif. Regularized bi-directional co-clustering. *Statistics and Computing*, 31(3) :32, 2021.

[2] Séverine Affeldt, Lazhar Labiod, and Mohamed Nadif. Caeclust : A consensus of autoencoders representations for clustering. *IPOL*, 12 :590–603, 2022.

[3] Sarka Brodinova, Peter Filzmoser, Thomas Ortner, Christian Breiteneder, and Maia Rohm. Robust and sparse k-means clustering for high-dimensional data. *Advances in Data Analysis and Classification*, 1 :1–28, 2019.

[4] Saptarshi Chakraborty and Swagatam Das. Detecting meaningful clusters from high-dimensional data : A strongly consistent sparse center-based clustering approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(6) :2894–2908, 2022.

[5] Amine Ferdjaoui, Séverine Affeldt, and Mohamed Nadif. Wordgraph : A python package for reconstructing interactive causal graphical models from text data. In *ACM WSDM*, page 1046–1049, 2024.

[6] M.J. Greenacre. *Theory and Applications of Correspondence Analysis*. Academic Press, 1984.

[7] Joshua Huang, Michael Ng, Hongqiang Rong, and Zichen Li. Automated variable weighting in k-means type clustering. *IEEE transactions on pattern analysis and machine intelligence*, 27 :657–68, 06 2005.

[8] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2) :129–137, 1982.

[9] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. Le Cam and J. Neyman, editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.

[10] Maziar Moradi Fard, Thibaut Thonet, and Eric Gaussier. Deep k-means : Jointly clustering with k-means and learning representations. *Pattern Recognition Letters*, 138 :185–192, 2020.

[11] Aghiles Salah, Melissa Ailem, and Mohamed Nadif. Word co-occurrence regularized non-negative matrix tri-factorization for text data co-clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[12] Aghiles Salah and Mohamed Nadif. Directional co-clustering. *Advances in Data Analysis and Classification*, 13 :591–620, 2019.

[13] Alexander Strehl and Joydeep Ghosh. Cluster ensembles - a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3 :583–617, 2002.

[14] Daniela M. Witten and Robert Tibshirani. A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490) :713–726, 2010.

[15] Daniela M. Witten and Robert Tibshirani. A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490) :713–726, 2010.

Associative Feature-Driven Undersampling for Imbalanced Data Classification
Sous-échantillonnage associatif basé sur les caractéristiques pour la classification de données déséquilibrées

Hajar Kamel[1], Hasna Chamlal[1], Tayeb Ouaderhman[1]

[1] Computer Science and Systems Laboratory (LIS), Faculty of sciences
Ain Chock, Hassan II University of Casablanca, Morocco

Abstract

As machine learning advances, imbalanced datasets remain a major challenge, often reducing model effectiveness. To address this, we propose Feature-Guided Associative Undersampling (FGAU), a framework that combines associative classification with adaptive undersampling. FGAU consists of three key components: Double-Side Filtering (DSF) for noise removal, Feature-Projected Associative Learning (FPAL) for mapping data into a compact space while preserving relationships, and Feature-Guided Undersampling (FGU) for selecting informative samples based on feature importance. Experiments on benchmark and real-world datasets show that FGAU significantly enhances accuracy and precision, effectively mitigating data imbalance.

Keywords

Imbalanced Data, Classification, Associative Learning, Association Rule, Adaptive Undersampling.

Résumé

La classification des données déséquilibrées constitue un défi majeur en apprentissage automatique. Nous proposons FGAU, un cadre combinant classification associative et sous-échantillonnage adaptatif. Ce modèle intègre trois étapes clés : un filtrage double face pour éliminer le bruit, un apprentissage associatif projeté pour préserver les relations entre caractéristiques, et un sous-échantillonnage guidé par les caractéristiques pour sélectionner les échantillons informatifs. Les expérimentations sur divers jeux de données réels démontrent une nette amélioration des performances de classification.

Mots-clés

Données déséquilibrées, Classification, Apprentissage associatif, règles d'association, Sous-échantillonnage adaptatif.

1 Introduction

In an era defined by the exponential growth of high-dimensional data, the pervasive challenge of imbalanced class distributions has transcended a mere technical inconvenience to emerge as a critical barrier that necessitates innovative methodologies capable of reconciling predictive precision with equitable representation [19]. The imbalanced datasets issue, where the majority class dominates the training dataset, leads to biased models and poor generalization for the minority class [9]. This problem is prevalent in various real-world applications such as fraud detection, medical diagnosis, and predictive maintenance, where the minority class instances are often the most crucial but least represented. Traditional classification algorithms assume balanced class distributions, which causes skewed decision boundaries and suboptimal performance on minority class predictions [4].

To address this challenge, several approaches have been proposed, including data-level, algorithm-level, and hybrid solutions. Among them, resampling techniques such as oversampling the minority class or undersampling the majority class are widely used to balance class distributions [13]. However, existing undersampling methods often discard valuable information, leading to potential loss of critical decision boundaries [15]. Additionally, noise and outliers in both majority and minority classes can further degrade classification performance. In this study, we propose Feature-Guided Associative Undersampling (FGAU), a novel framework that combines associative classification principles [1] with an adaptive undersampling strategy [14]. Our approach enhances imbalanced classification by filtering noise, preserving essential feature relationships, and guiding sample selection based on feature importance and local neighborhood structures. Through extensive experimental evaluations, we demonstrate that FGAU significantly improves classification stability and precision while mitigating the adverse effects of class imbalance.

2 Related Works

Class imbalance has been extensively studied, and various methods have been proposed to address the issue. These methods can be broadly categorized into data-level, algorithm-level, and ensemble-based approaches [12]. Resampling techniques modify the training data distribution

to alleviate imbalance [2]. Oversampling methods such as Synthetic Minority Over-sampling Technique (SMOTE) generate synthetic minority samples to improve class representation [7]. Undersampling methods, on the other hand, reduce the majority class instances to balance the dataset, but they often risk losing valuable information [3]. Hybrid approaches combine both oversampling and undersampling to achieve a more balanced representation while minimizing information loss.

Algorithmic solutions focus on modifying the learning process to improve classification on imbalanced datasets. Cost-sensitive learning assigns higher misclassification costs to minority class instances, encouraging models to improve their prediction accuracy for these cases. Additionally, modifications to standard machine learning algorithms, such as adjusting decision thresholds or modifying loss functions, have been explored to enhance classification performance on imbalanced data [16].

As for Ensemble methods, including bagging and boosting, have been widely adopted to address class imbalance issues. Techniques such as Balanced Random Forest [6] and Adaptive Boosting [18] incorporate resampling strategies and cost-sensitive learning to improve classification performance. These methods leverage multiple weak classifiers to enhance model generalization and robustness against imbalance. Over the past two decades, numerous ensemble-based methods have been developed to enhance conventional bagging, particularly in addressing CI challenges. Hido Kashima et al. [8] introduced a novel undersampling bagging based technique that employs a negative binomial distribution to perform bagging on imbalanced data. Later, Lango and Stefanowski [11] further extended the bagging framework to accommodate a larger number of attributes and multiple minority classes. The famous ensemble learner SMOTEBoost introduced by Chawla et al. [5], tackles the imbalanced data problem by seamlessly integrating the Synthetic Minority Over-sampling Technique with conventional boosting procedures. Ensemble methods are also increasingly applied to boost metrics like precision and recall. A good example is HML [4], which performs well in classifying minority instances in the imbalanced contexts.

Another interesting aspect is associative classification, an alternative approach that employs association rule mining to uncover relationships between features and class labels, offering promising potential for addressing imbalanced data challenges by preserving minority class information and enhancing overall model interpretability. Some studies have attempted to incorporate feature selection and rule pruning techniques to improve minority class representation. However, these methods often overlook the impact of noise and redundant samples in both majority and minority classes.

Motivation: Despite the advancements in imbalance learning, existing methods face challenges in preserving minority class structure while mitigating noise. Our proposed FGAU framework aims to address these gaps by integrating associative learning with an adaptive undersampling mechanism guided by feature importance. Unlike traditional undersampling methods that randomly discard majority class instances, FGAU selectively removes redundant samples while retaining critical decision boundaries. Additionally, the hybrid framework leverages a novel feature-projected learning approach to enhance class separation in high-dimensional data spaces.

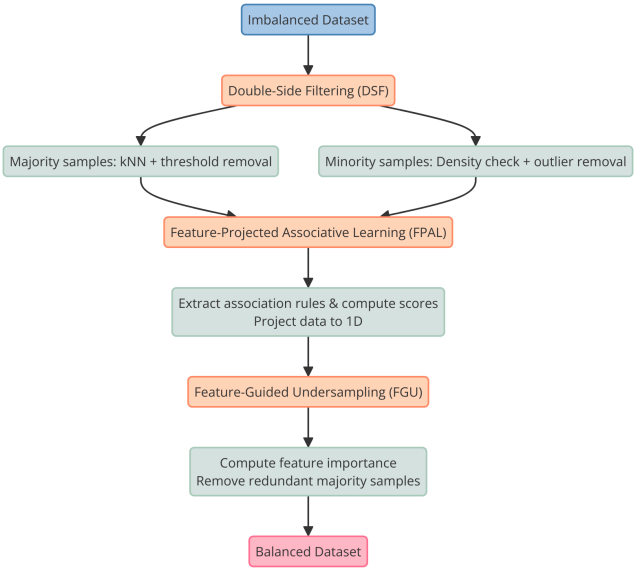

Figure 1: The flowchart of the proposed FGAU

3 Proposed Method

The FGAU framework enhances imbalanced data classification by integrating associative learning with a feature-guided undersampling strategy, Figure 1 represents a clear flowchart of the proposed procedure. Our approach operates in three stages. In the first stage, Double-Side Filtering (DSF) reduces noise and redundancy by filtering out irrelevant samples from both the majority and minority classes. For the majority class, each sample's contribution is evaluated by computing its k-nearest neighbors using the Euclidean distance

$$d(x_i, x_j) = \sqrt{\sum_{l=1}^{m} \left(x_i^{(l)} - x_j^{(l)}\right)^2},$$

and samples with distances exceeding a preset threshold are removed; for the minority class, local density measures are applied to detect and eliminate outliers. In the second stage, Feature-Projected Associative Learning (FPAL) maps high-dimensional data into a one-dimensional (1D) representation that retains key feature interactions. Association rules [10] linking features to class labels are extracted and evaluated using the metrics support and confidence, and each sample is assigned a 1D score by aggregating the contribu-

tions of activated rules:

$$s_i = \sum_{r \in R} w_r \cdot I(x_i, r),$$

where $I(x_i, r) = \begin{cases} 1 & \text{if sample } x_i \text{ satisfies rule } r, \\ 0 & \text{otherwise} \end{cases}$ In the final phase, Feature-Guided Undersampling selectively reduces the majority class by computing feature importance scores using mutual information (MI) [17], and performing local neighborhood analysis in the projected space; redundant majority samples are removed and the refined majority samples are merged with the filtered minority samples to form the balanced training dataset $D_{balanced}$. This integrated approach reduces noise, preserves discriminative feature interactions, and ultimately improves classification performance on imbalanced datasets. Algorithm 1 summarizes the complete FGAU framework :

Algorithm 1 Feature-Guided Associative Undersampling (FGAU)

Require: Training dataset D with imbalanced classes.
Ensure: Balanced dataset $D_{balanced}$.

1: **DSF:** For each sample in D, evaluate neighbor distances and mark redundant/outlier samples.
2: Remove marked samples to obtain $D_{filtered}$.
3: **FPAL:** Extract association rules from $D_{filtered}$ and compute a one-dimensional score s_i for each sample.
4: Project each sample using s_i.
5: **FGU:** For majority class samples in the projected space, compute feature importance and local distances; remove redundant samples.
6: Merge the retained majority samples with the filtered minority samples to form $D_{balanced}$.
7: **return** $D_{balanced}$

4 Experimental Analysis

We evaluated our proposed Feature-Guided Associative Undersampling (FGAU) method on a diverse collection of benchmark datasets. These datasets, detailed in Table 1, include several E. coli datasets, the `cleveland0vs4` dataset, and two Glass datasets. For each dataset, we report the total number of samples, the counts of minority and majority class instances, and the imbalance ratio.

To assess the effectiveness of FGAU, we compared its performance against state-of-the-art techniques such as SMOTEBoost, Roughly Balanced Bagging (RBB), and Baseline approaches. Our primary evaluation metrics are F1-score and G-mean.

The results in Table 2 and Table 3 clearly indicate that FGAU outperforms the competing methods in terms of both F1-score and G-mean across all datasets, the results are visualised in Figure 2 and Figure 3 . For example, on the `ecoli0146vs5` dataset (Total = 280, Imbalance Ratio = 13), FGAU achieves an F1-score of 91.4% and a G-mean of 90.0%, significantly surpassing the scores obtained by

Table 1: Dataset Characteristics

Dataset	Total	Minority	Majority	Imbalance Ratio
ecoli067vs5	220	20	200	10.00
ecoli0147vs2356	336	30	306	10.59
ecoli067vs35	222	22	200	9.09
ecoli0234vs5	203	20	183	9.15
ecoli046vs5	202	20	182	9.10
ecoli01vs235	244	24	220	9.17
ecoli0267vs35	203	20	183	9.15
ecoli0346vs5	205	20	185	9.25
ecoli0347vs56	257	25	232	9.28
ecoli01vs5	240	20	220	11.00
ecoli0146vs5	280	20	260	13.00
ecoli0147vs56	336	30	306	10.20
cleveland0vs4	173	13	160	12.31
glass1	214	76	138	1.82
glass0	214	70	144	2.06

Table 2: F1-Score (%) Comparison

Dataset	FGAU	SMOTEBoost	RBB	Baseline
ecoli067vs5	92.5	88.7	87.0	86.5
ecoli0147vs2356	91.8	87.2	86.5	85.8
ecoli067vs35	93.0	89.0	88.0	87.0
ecoli0234vs5	92.3	88.0	87.0	86.5
ecoli046vs5	92.7	88.2	87.3	86.8
ecoli01vs235	93.2	88.9	88.0	87.5
ecoli0267vs35	92.6	88.1	87.3	86.7
ecoli0346vs5	92.9	88.5	87.8	87.0
ecoli0347vs56	93.1	88.7	88.0	87.2
ecoli01vs5	92.8	88.4	87.7	87.0
ecoli0146vs5	91.4	87.0	86.2	85.7
ecoli0147vs56	92.0	87.5	87.0	86.5
cleveland0vs4	93.8	89.5	89.0	88.5
glass1	89.5	84.5	84.0	83.5
glass0	89.7	84.8	84.2	83.8

SMOTEBoost, RBB, and Baseline methods. Similarly, on the `cleveland0vs4` dataset, FGAU records an F1-score of 93.8% and a G-mean of 92.0%, demonstrating its robust performance in highly imbalanced scenarios. These findings confirm the efficacy of FGAU as a powerful solution for imbalanced data classification.

5 Conclusion

The FGAU framework introduces an innovative strategy for tackling imbalanced data classification by seamlessly integrating feature-guided undersampling with associative learning techniques. Unlike conventional methods, FGAU intelligently filters out noisy instances while preserving the most crucial relationships within the data, ensuring that essential patterns are not lost in the process. By prioritizing feature importance, it refines the dataset in a way that enhances both classification stability and precision, leading to more reliable and interpretable results. This makes FGAU a powerful and adaptable tool for addressing the persistent challenges of imbalanced learning, particularly in domains where data scarcity and class imbalance compromise model performance.

References

[1] Neda Abdelhamid and Fadi Thabtah. Associative classification approaches: review and comparison. *Journal of Information & Knowledge Management*, 13(03):1450027, 2014.

[2] Azwaar Khan Azlim Khan and Nurul Hashimah

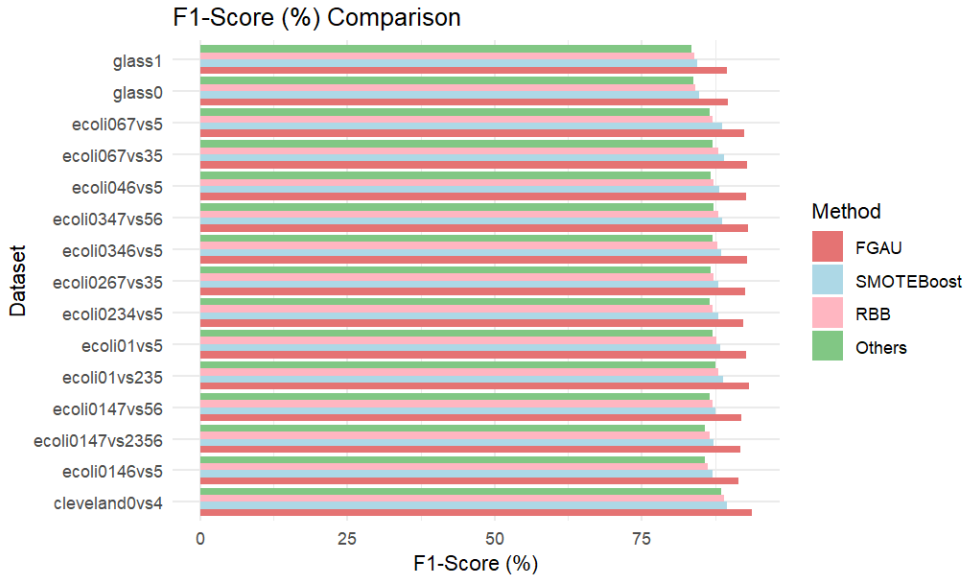

Figure 2: Comparison of F1-Scores (%) across methods, showing FGAU outperforming others

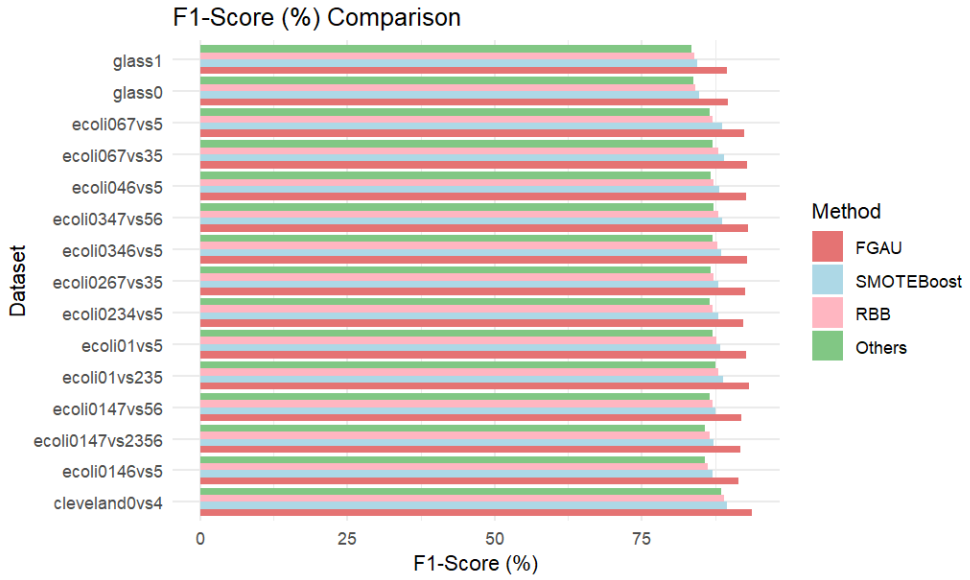

Figure 3: Comparison of G-mean values (%) across methods, showing FGAU outperforming others

Ahamed Hassain Malim. Comparative studies on resampling techniques in machine learning and deep learning models for drug-target interaction prediction. *Molecules*, 28(4):1663, 2023.

[3] Malgorzata Bach, Aleksandra Werner, J Żywiec, and Wojciech Pluskiewicz. The study of under-and over-sampling methods' utility in analysis of highly imbalanced data on osteoporosis. *Information Sciences*, 384:174–190, 2017.

[4] Hasna Chamlal, Hajar Kamel, and Tayeb Ouaderhman. A hybrid multi-criteria meta-learner based classifier for imbalanced data. *Knowledge-based systems*, 285:111367, 2024.

[5] Nitesh V Chawla, Aleksandar Lazarevic, Lawrence O Hall, and Kevin W Bowyer. Smoteboost: Improving prediction of the minority class in boosting. In *European conference on principles of data mining and knowledge discovery*, pages 107–119. Springer, 2003.

[6] Abdul Waheed Dar and Sheikh Umar Farooq. Handling class overlap and imbalance using overlap

Table 3: G-Mean Comparison

Dataset	FGAU	SMOTEBoost	RBB	Baseline
ecoli067vs5	91.0	87.0	86.0	85.5
ecoli0147vs2356	90.5	86.0	85.5	84.8
ecoli067vs35	91.2	87.5	86.8	86.0
ecoli0234vs5	90.8	86.5	86.0	85.2
ecoli046vs5	91.0	86.8	86.2	85.7
ecoli01vs235	91.3	87.2	86.5	86.0
ecoli0267vs35	90.9	86.5	86.0	85.5
ecoli0346vs5	91.1	87.0	86.5	85.8
ecoli0347vs56	91.2	87.2	86.8	86.0
ecoli01vs5	91.0	87.0	86.5	85.8
ecoli0146vs5	90.0	86.0	85.0	84.5
ecoli0147vs56	90.7	86.5	86.0	85.5
cleveland0vs4	92.0	88.0	87.5	87.0
glass1	88.0	83.5	83.0	82.5
glass0	88.2	83.8	83.2	82.8

driven under-sampling with balanced random forest in software defect prediction. *Innovations in Systems and Software Engineering*, pages 1–21, 2024.

[7] Dina Elreedy and Amir F Atiya. A comprehensive analysis of synthetic minority oversampling technique (smote) for handling class imbalance. *Information Sciences*, 505:32–64, 2019.

[8] Shohei Hido, Hisashi Kashima, and Yutaka Takahashi. Roughly balanced bagging for imbalanced data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 2(5-6):412–426, 2009.

[9] Justin M Johnson and Taghi M Khoshgoftaar. Survey on deep learning with class imbalance. *Journal of big data*, 6(1):1–54, 2019.

[10] Sotiris Kotsiantis and Dimitris Kanellopoulos. Association rules mining: A recent overview. *GESTS International Transactions on Computer Science and Engineering*, 32(1):71–82, 2006.

[11] Mateusz Lango and Jerzy Stefanowski. Multi-class and feature selection extensions of roughly balanced bagging for imbalanced data. *Journal of Intelligent Information Systems*, 50:97–127, 2018.

[12] Joffrey L Leevy, Taghi M Khoshgoftaar, Richard A Bauder, and Naeem Seliya. A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5(1):1–30, 2018.

[13] Chun-Yang Peng and You-Jin Park. A new hybrid under-sampling approach to imbalanced classification problems. *Applied Artificial Intelligence*, 36(1):1975393, 2022.

[14] Min Qian and Yan-Fu Li. A novel adaptive undersampling framework for class-imbalance fault detection. *IEEE Transactions on Reliability*, 72(3):1003–1017, 2022.

[15] Amirreza Salehi and Majid Khedmati. Hybrid clustering strategies for effective oversampling and undersampling in multiclass classification. *Scientific Reports*, 15(1):3460, 2025.

[16] Nguyen Thai-Nghe, Zeno Gantner, and Lars Schmidt-Thieme. Cost-sensitive learning methods for imbalanced data. In *The 2010 International joint conference on neural networks (IJCNN)*, pages 1–8. IEEE, 2010.

[17] Jorge R Vergara and Pablo A Estévez. A review of feature selection methods based on mutual information. *Neural computing and applications*, 24:175–186, 2014.

[18] Wenyang Wang and Dongchu Sun. The improved adaboost algorithms for imbalanced data classification. *Information Sciences*, 563:358–374, 2021.

[19] Yuanting Yan, Yuanwei Zhu, Ruiqing Liu, Yiwen Zhang, Yanping Zhang, and Ling Zhang. Spatial distribution-based imbalanced undersampling. *IEEE Transactions on Knowledge and Data Engineering*, 35(6):6376–6391, 2022.

Multitrajectory Analysis in Finite Mixture Models

Cédric Noel[1], Jang Schiltz[2]

[1] Université de Lorraine, IUT Thionville-Yutz
[2] Université du Luxembourg, Département de Finance

cédric.noel@univ-lorraine; jang.schiltz@uni.lu

Résumé

Dans son livre "Group-Based Modeling of Development", Nagin (2005) introduit un modèle multitrajectoire qui permet d'analyser deux séries temporelles conjointement et qui a l'avantage de contenir aussi des paramètres fournissant de l'information sur la possible structure relationnelle entre les deux séries. Le désavantage du modèle multitrajectoire de Nagin est qu'il est numériquement lourd à calculer. Nous présentons dans cet article un modèle multitrajectoire alternatif qui suit une logique différente et peut être calibré à l'aide de la librairie R trajeR, conçue par Noel (Noel and Schiltz 2022).

Mots-clés

Modèle de mélanges finis, analyse conjointe de plusieurs trajectoires, librairie R

Abstract

In his book "Group-Based Modeling of Development", Nagin (2005) introduces a multitrajectory model which allows to analyze two time series simultaneously, which has the advantage of exhibiting also some information about the possible relationship structure between the two time series. Nagin's model necessitates to compute a high number of parameters and is numerically quite difficult to handle. In this paper, we present an alternative multitrajectory model which needs less parameters and can be calibrated with the R package trajeR programmed by Noel (Noel and Schiltz 2022).

Keywords

Finite mixture models, multitrajectory analysis, R package

1 Introduction

In his book "Group-Based Modeling of Development", Nagin (2005) provides a systematic exposition of a group-based statistical method for analyzing longitudinal data that he calls finite mixture models. The aim of these models is to divide the population into homogenous groups and to evaluate a typical developmental trajectory for each of these groups at the same time. Usually finite mixture models are applied to one time series only. Nagin (2005) however introduces also a multitrajectory model which allows to analyze two time series simultaneously, which has the advantage of exhibiting also some information about the possible relationship structure between the two time series. The supplemental parameter of interest of Nagin's multitrajectory model are the conjoint membership probability to be in a given group for the first time series and in another given group for the second time series. Nagin and coauthors also generalized his model to the case of more than just two time series (Burckhardt et ali 2016), but to this day, there are no practical applications with more than two. This could be due to the fact that Nagin's model necessitates to compute a high number of parameters and is numerically quite difficult to handle.

In this paper, we present an alternative multitrajectory model which needs less parameters and can be calibrated with the R package trajeR programmed by Noel (Noel and Schiltz 2022).

The rest of the paper is organized as follows. In section 2, we present the general theory of finite mixture models. Section 3 is then dedicated to multitrajectory modeling. We first present Nagin's multitrajectory model and then our alternative model and compute the number of parameters for both models. In sections four and five, we present the numerical algorithms to resolve our multitrajectory model. Section four shows the algorithms needed for direct optimization and section five the resolution using the EM algorithm. Section six introduces the R package trajeR and explains how to use it in order to work with multitrajectory models.

2 Finite Mixture Models

In group based trajectory modeling, we consider a population of size N divided into K latent classes. The assignment of the individuals into classes is based on the degree of similarity of the developmental trajectories.

More precisely, consider a time-varying variable of interest Y defined in a population Ω of size N. Let $Y_i = y_{i_1}, \cdots, y_{i_T}$ be T measures of the variable Y, taken at times $t_1, \cdots, t_T$ for subject number i.

The aim of the analysis is to divide the population into K sub-populations $G_1, ...G_K$, which are homogeneous in the sense that two subjects in the same group have similar trajectories for the variable of interest Y and two subjects in different groups have quite different trajectories for the variable of interest Y.

Let $P^k(Y_i)$ be the probability of Y_i given membership in group G_k and $P(Y_i)$ the unconditional probability of observing the realization Y_i of Y. Furthermore, for a given group G_k, we suppose conditional independence for the sequential realizations of the elements y_{i_t} over the T periods of measurements. Then,

$$P(Y_i) = \sum_{k=1}^{K} P\left(G_i = k\right) P^k(Y_i). \tag{1}$$

By definition of a finite mixture model [**?**], the density f of Y is given by

$$f(y_i;\psi) = \sum_{k=1}^{K} \pi_k g_k(y_i;\Theta_k). \tag{2}$$

The role of the parameters Θ_k is to describe the shape of the trajectories in group k.

Moreover, the group size $\pi_k > 0$ denotes the probability of a given subject to belong to group number k and thus

$$\sum_{k=1}^{K} \pi_k = 1.$$

Since in practice, it is difficult to constraint the π_k to be numbers between 0 and 1, we link the π_k to a set of parameters $\theta_1, ..., \theta_K$ such that

$$\pi_k = \frac{e^{\theta_k}}{\sum_{k=1}^{K} e^{\theta_k}}.$$

On top of the parameters defining the underlying distribution, the model depends thus on the parameter set $\psi = (K, \theta_1, \cdots, \theta_{K-1}, \Theta_1, \cdots, \Theta_K)$.

There are two interesting generalizations of this basic finite mixture model. First, we can test if group membership of the individuals is influenced by a static set of R risk variables $X = (X_1 \cdots X_R)$, that are typically measured before the start of the trajectories Y. Second, we can investigate the relationship between the trajectories Y and a time-dependent covariate W which is independent of X. Let's point out, that this is basically regression analysis, so there is no automatic proof of causality here. We test if there is a significant relationship between Y and W and can only conclude that W influences the trajectories of Y if we get the causal status of the relationship by other means.
Combining these two extensions, the conditional density of Y given X and W is given by
$\mathrm{f}(y_i|x_i, w_i) = \sum_{k=1}^{K} \Big(P(G_i = k|X_i = x_i$
$\times \prod_{t=1}^{T} P(Y_{i_t} = y_{i_t}|X_i = x_i, W_i = w_i, G_i = k) \Big)$, which can be written as $\mathrm{f}\Big(\mathrm{y}_i|x_i, w_i \Big) =$
$\sum_{k=1}^{K} \left(\sum_{j=1}^{R} \frac{e^{x_i^j \theta_k^j}}{1+e^{x_i^j \theta_k^j}} \prod_{t=1}^{T} P\left(Y_{i_t} = y_{i_t}|W_i = w_i, G_i = k\right) \right)$.

Nagin (2005) defined this model for underlying logit, (censored) normal and zero inflated Poisson distributions, whereas Elmer et ali (2018) and Noel and Schiltz (2024) extended the model to an underlying Beta distribution.

3 Group-based multitrajectory modeling

In case of several time series that are related to each other somehow, it makes sense to analyze them conjointly rather than separately and to take their dependency structure into account. Here the different trajectories can evolve contemporaneously or over different time periods. Nagin (2005) proposes an extension of the basic finite mixture model to deal with this situation. Burckhardt et ali (2016) extend this dual trajectory model it to a multiple setting. In this paper, we propose an alternative model with parameters that are easier to interpret in some situations, especially in the case of more than just two time series, where the number of parameters in Nagin's model is blowing up very fast.

3.1 Nagin's model

We suppose that we want to analyze J dependent time series Y^j, $1 \leq j \leq J$. Let Y_i^j denote the value of Y^j for the ith individual in the sample and let K_j be the number of homogenous groups the population is divided into, according to the time series Y^j. We suppose further that the time series $(Y^j)_{1 \leq j \leq J}$ are independently distributed, conditional on the group memberships of the different time series, so that $P^{k_1 \ldots k_J}(Y_i^1, \ldots, Y_i^J) = \prod_{j=1}^{J} P^{k_j}(Y_i^j)$.
Thus, the likelihood function of the data can we written as
$\mathrm{P}(\mathrm{Y}_i^1, \ldots, Y_i^J|A_i, W_i, X_i)$
$= \sum_{(k_1,\ldots,k_J) \in K_1 \times \cdots \times K_J} \pi_{k_1 \ldots k_J} \prod_{j=1}^{J} P^{k_j}(Y_i^j|A_i, W_i, \Theta_k^j)$
$= \sum_{(k_1,\ldots,k_J) \in K_1 \times \cdots \times K_J} \pi_{k_J|k_1 \ldots k_{J-1}} \times \ldots \times \pi_{k_2|k_1} \times \pi_{k_1}$
$\times \prod_{j=1}^{J} \prod_{t=1}^{T^j} g^{k_j}(y_{it}^j|A_i, W_i, \Theta_k^j)$.
For problems in which the different time series follow a temporal order, there is of course a natural way to order the time series. In general, the order in which the time series Y^j are considered is however arbitrary and there are actually $J!$ different formulas for the likelihood function. In practice, one chooses the order of the time series in a way that allows for an easy computation and interpretation of the conditional group membership probabilities.

For the computation of the conditional membership probabilities, we again link the group memberships to parameters $\theta_{k_i}^{k_1 \ldots k_{i-1}}$. For a model without covariate, we write

$$\pi_{k_1} = \frac{e^{\theta_{k_1}}}{\sum_{k_1=1}^{K_1} e^{\theta_{k_1}}}, \quad \pi_{k_2|k_1} = \frac{e^{\theta_{k_2}^{k_1}}}{\sum_{k_2=1}^{K_2} e^{\theta_{k_2}^{k_1}}}, \quad ,$$

$$\pi_{k_J|k_1 \ldots k_{J-1}} = \frac{e^{\theta_{k_J}^{k_1 \ldots k_{J-1}}}}{\sum_{k_J=1}^{K_J} e^{\theta_{k_J}^{k_1 \ldots k_{J-1}}}},$$

where $\theta_{k_1}, \theta_{k_2}^{k_1}, \ldots, \theta_{k_J}^{k_1 \ldots k_{J-1}} \in R$.

This implies the computation of $(K_1-1)+(K_1-1)\times(K_2-1)+...+(K_J-1)\times(K_{J-1}-1)\times\ldots\times(K_1-1)$ parameters.

3.2 An alternative model

In this section, we present an alternative model, inspired by the multinomial logit model of Bel & Paap (2014).

We denote by $Z=(Z_{i1},\ldots,Z_{iJ})$ the latent variable which contains the group membership of the individuals for each of the outcomes $Y^1,\ldots,Y^J$. Thus $Z_i\in[\![1;K_1]\!]\times\cdots\times[\![1;K_J]\!]$.
Let S be the set of all possible realizations of Z. Note that $|S|=\Pi_{k=1}^{J}K_k$. We suppose furthermore that group membership for the j-th time series may be influenced by covariate X^j. To simplify notations, we take X^j one-dimensional here, but it is of course easy to extend this to a multidimensional setting.
Following Bel & Paap (2014), we can then write the membership probability for the j-th time series conditional on group memberships of all the other time series as

$$P\left(Z_{ij}=k|z_{ih}for h\neq j,X_i^j\right)=\frac{e^{B_{ij,k}}}{\sum_{h=1}^{K_j}e^{B_{ij,h}}},\qquad(3)$$

where $B_{ij,k}=\alpha_{j,k}+\beta_{j,k}X_i^j+\sum_{h\neq j}\psi_{jh,kz_{ih}}$.
Here,

- $\alpha_{j,k}$ is a group specific intercept for outcome Y^j ;
- $\beta_{j,k}$ is a parameter vector associated to the covariate X^j ;
- z_{ih} gives the group membership of the individual i for the h-th outcome ;
- $\psi_{jh,kz}$ is an association parameter linked to membership in group k for the j-th outcome and membership in group z for the h-th outcome.

Let's point out that if all the association parameters $\psi_{jh,kz}$ are zero, we recover the classical formula for π_k for independent outcomes. For the parameters to be identifiable, we have to impose the standard identification restrictions by choosing one group for which the parameters values are 0. Thus, we choose $\alpha_{j,1}=0$ and $\beta_{j,1}=0$ for all j. Moreover, we choose $\psi_{jh,1z}=\psi_{hj,z1}=0$ for all h and z. This choice allows a direct interpretation of the association parameters via odds ratios.
Furthermore, we want the parameters $\psi_{jh,kz}$ to describe a symmetric relationship between the groups, thus we impose $\psi_{jh,kz}=\psi_{hj,zk}$ for all k and z.
As shown in Bel & Paap (2014), we then get that

$$P\left(Z_i=z_i|X_i\right)=\frac{e^{\mu_{z_i}}}{\sum_{s_i\in S}e^{\mu_{s_i}}},\qquad(4)$$

where $\mu_{z_i}=\sum_{j=1}^{J}\left(\alpha_{j,z_{ij}}+\beta_{j,z_{ij}}X_i^j+\sum_{h>j}\psi_{jh,z_{ij}z_{ih}}\right)$.

The parameters $\psi_{jh,kz}$ are theoretically unbounded and do thus not directly resemble correlation between memberships in groups k and z. Bel & Paap (2014) show however that it is possible to give a direct interpretation to these associations through log odds ratios. A positive $\psi_{jh,kz}$ implies that membership in groups k and z move together more often than apart.

The alternative multitrajectory finite mixture model is then defined by its likelihood function of the data
$\mathrm{P}(\mathrm{Y}_i^1,\ldots,Y_i^J|A_i,W_i,X_i)=\sum_{(k_1,\ldots,k_J)\in K_1\times\cdots\times K_J}$
$\left(\frac{e^{\mu_k}}{\sum_{\tilde{k}\in S}e^{\mu_{\tilde{k}}}}\right)\prod_{j=1}^{J}\prod_{t=1}^{T^j}g^{k_j}(y_{it}^j|A_i,W_i,\Theta_k^j)$,

where $\mu_k=\sum_{j=1}^{J}\left(\alpha_{j,k_j}+\beta_{j,k_j}X_i^j+\sum_{h>j}\psi_{jh,k_jk_h}\right)$.

We can then easily compute the number of parameters that needed to be estimated for this model. The numbers of parameters in the alternative multitrajectory model that need to be estimated to compute the joint group memberships is

$$\sum_{j=1}^{J}(K_j-1)\times(ncol(X^j)+1)+\sum_{1\leq j<j'\leq J}(K_j-1)(K_{j'}-1).\qquad(5)$$

The result follows easily from the above mentioned standard identification restrictions.
In the next two sections, we will present two methods to estimate the parameters of the alternative multitrajectory model, direct maximum likelihood estimation on the one hand and the use of the EM algorithm on the other.

4 Parameter estimation

The parameters of the model can be estimated either by direct differentiation of the likelihood function or by using the expectation-maximization (EM) algorithm. In both cases, the formulas for the computation are too long to put them in a two-column format, so we do not present them here.

5 The R package trajeR

Jones, Nagin and Roeder (2007) have developed a **SAS** procedure **Proc Traj** to calibrate finite mixture models, as well as a **Stata** version (Jones and Nagin 2012). Our **R** package **trajeR** allows to calibrate finite mixture models for all commonly used densities (Noel and Schiltz 2022). The proofs of all algorithms inside this package can be found in Noel's PhD thesis (Noel 2025).
The function `multitrajeR` in **trajeR** fits the multitrajectory model and computes its parameters for given degrees of the polynomial trajectories in the different groups. The function signature for `multitrajeR` is

```
R> multitrajeR(Y, degre.Y, A,
+     RISK = NULL, TCOV = NULL,
```

```
+     degre, degre.phi=0, Model = "BETA",
+     Method = "L", ssigma = FALSE,
+     itermax = 100, ymax = max(Y) + 1,
+     ymin = min(Y) - 1,hessian = TRUE,
+     paraminit = NULL,diffct = NULL,
+     control = list(fnscale=-1, trace=1),
+     ProbIRLS = TRUE, refgr = 1,
+     nbvar = NULL, nls.limiter = 50)
```

Some of these arguments are mandatory others optional.

The mandatory arguments are the main data matrices `Y`, `A`, as well as `degre`, `degre.phi`, `Model` and `Method`.
Here `Y` is the matrix containing the values of the variable of interest and `A` is the matrix containing the age or time variable. In most applications, this matrix just contains times of measurement that are the same for each individual in the sample, implying that all lines of the matrix `A` are equal, but this is not necessarily the case. `A` can for instance contain the age of the different individuals at the times of measurement, which is generally different for each individual in the sample.
`degre` is a vector indicating the degree of the polynomials describing the typical trajectories in the different groups. Implicitly, the dimension of this vector also indicates the number of groups into which we want to divide the population, so there is no no need for a seperate argument for the number of groups.
`degre.phi` is a vector indicating the degree of the polynomials describing the precision parameters in the different groups.
`Model` is a string defining the underlying distribution used in the model. The possible choices are LOGIT for the Logistic Regression Mixture Model, CNORM for the Censored Normal Mixture Model, ZIP for the Zero Inflated Poisson Mixture model and BETA for the BETA model.
`Method`, finally, is a string to decide which algorithm is used for estimating the model parameters. In case of the BETA model, only `L` for direct optimization is possible.

The optional arguments are `Risk`, `TCOV`, `degre.nu`, `ssigma`, `ymax`, `ymin`, `hessian`, `itermax`, `paraminit`, `ProbIRLS`, `refgr`, `fct`, `diffct`, `nls.lmiter`, `ng.nl` and `nbvar`.
`Risk` is a data matrix that contains the values of the covariate X modifying the group membership probability. By default, there is no such variable and `Risk` is a one-column matrix with value 1.
`ProbIRLS` allows to decide which method is used to compute the predictor probabilities. If its value is TRUE (default setting) we use the IRLS method and if it is FALSE we use the optimization method.
`TCOV` is an optional data matrix containing a time-dependent covariate W that influences the trajectories themselves. By default its value is NULL.
To ensure the identifiability of the parameters of the predictor, we have to fix a reference group. This can be done by the `refgr` command. It's default value is 1.
`hessian` indicates if we want to calculate the Hessian matrix, the default value being FALSE. If the method used is direct optimization, the Hessian matrix is computed by inverting the Fisher Information Matrix.
`itermax` gives the maximal number of iterations for the `optim` function or for the EM algorithm.
The choice of the initial parameters is very important in optimization problems. We can specify these initial parameters by `paraminit`. By default `trajeR` calculates the initial value based on the range or the standard deviation of the data (for the details, see Noel (2025)).

Références

[1] K. Bel and R. Paap, "A Multivariate Model for Multinomial Choices," Econometric Institute Report, 26, 2014.

[2] P. Burckhardt, D.S. Nagin and R. Padman, "Multi-Trajectory Models of Chronic Kidney Disease Progression," AMIA Annual Symposium Proceedings, pp. 1737-1746, 2016.

[3] J. Elmer, B. L. Jones and D.S. Nagin, "Using the Beta distribution in group-based trajectory models," BML Medical Research Methodology, 18 (152), pp. 1-5, 2018.

[4] B. L. Jones and D. S. Nagin, "Advances in Group-Based Trajectory Modeling and an SAS Procedure for Estimating them," Sociological Methods & Research, 35 (4), pp.542-571, 2007.

[5] B. L. Jones and D. S. Nagin, "A Stata Plugin for Estimating Group-Based Trajectory Models." Heinz College Research Working Paper, 2012.

[6] B. L. Jones, D. S. Nagin and K. Roeder, "A SAS Procedure Based on mixture Models for Estimating Developmental Trajectories," Sociological Methods & Research, 29 (3), pp. 374-393, 2001.

[7] D.S. Nagin, Group-Based Modeling of Development. Cambridge, Massachusets : Harvard University Press, 2005.

[8] C. Noel, On a generalisation of Nagin's finite mixture model. PhD Thesis, University of Luxembourg, Luxembourg, 2025.

[9] C. Noel and J. Schiltz. "trajeR, an R package for cluster analysis of time series." Working Paper, University of Luxembourg, Luxembourg, 2022.

[10] C. Noel and J. Schiltz, "Finite Mixture Models for an underlying Beta distribution with an application to COVID-19 data," in M. Stemmler, W. Wiedermann and F. L. Huong (eds.), Dependent Data in Social Sciences Research. Second Edition, New York : Springer, 2024.

[11] J. Schiltz J. "A generalization of Nagin's finite mixture model, in : M. Stemmler, A. Von Eye and W. Wiedermann (eds.), Dependent Data in Social Sciences Research. New York : Springer, pp. 107-126. 2015.