HAL open science

Pénalisation l 1 pour un mélange de lois de von Mises-Fisher

Florian Barbaro, Fabrice Rossi

To cite this version:

Florian Barbaro, Fabrice Rossi. Pénalisation l 1 pour un mélange de lois de von Mises-Fisher. JDS 2021, Jun 2021, Nice, France. hal-03285717

HAL Id: hal-03285717

https://hal.science/hal-03285717v1

Submitted on 13 Jul 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Pénalisation l_1 pour un mélange de lois de von Mises-Fisher

Florian Barbaro [1] & Fabrice Rossi [2]

[1] *Université Paris 1 Panthéon-Sorbonne - Laboratoire SAMM EA 4543, florian.barbaro@etu.univ-paris1.fr*
[2] *Université Paris Dauphine-PSL - CEREMADE UMR 7534, rossi@ceremade.dauphine.fr*

Résumé. Les mélanges de lois de von Mises-Fisher permettent de construire des classifications (non supervisées) de données sur la sphère unité. Ces mélanges sont bien adaptés aux données directionnelles de grande dimension comme les textes. Pour améliorer la qualité des classes et leur interprétabilité, nous proposons dans cet article de pénaliser la vraisemblance par un terme l_1, ce qui conduit à des centroïdes parcimonieux. Nous dérivons un algorithme EM pour ce modèle et nous illustrons l'intérêt de notre approche sur un jeu de données réelles.

Mots-clés. Mélanges de lois de von Mises-Fisher, pénalisation l_1, données de grande dimension.

Abstract. Mixtures of von Mises-Fisher distributions can be used to cluster data on the unit hypersphere. This is particularly adapted for high dimensional directional data such as texts. We propose in this article to estimate a von Mises mixture using a l_1 penalised likelihood. This leads to sparse prototypes that improve both clustering quality and interpretability. We introduce an EM algorithm for this estimation and show the advantages of the approach on real data benchmark.

Keywords. Mixtures of von Mises-Fisher distributions, l_1 penalty, high-dimensional data.

1 Introduction

Beaucoup de modèles de mélanges classiques sont peu adaptés aux données de grande dimension, par exemple issues de la représentation vectorielle de textes. Quand les données sont directionnelles [Mardia and Jupp, 2009], c'est-à-dire quand c'est plutôt leur corrélation que leur distance euclidienne qui importe, les modèles de type Gaussien sont encore moins adaptés. Pour de telles données, il est naturel d'opérer à une normalisation qui les place sur la sphère unité. On montre alors que les mélanges de lois de von Mises-Fisher (vMF) sur cette sphère sont bien adaptées pour la classification (non supervisée), cf [Banerjee et al., 2005, Gopal and Yang, 2014].

Dans cet article, en s'inspirant de [Pan and Shen, 2007], nous proposons une pénalisation l_1 pour un mélange de distributions vMF pour augmenter la parcimonie des moyennes

directionnelles et ainsi améliorer la compréhension des résultats de classification des données de grande dimension. Notre solution s'appuie sur une modification de l'algorithme espérance-maximisation proposé par [Banerjee et al., 2005].

Notations Les matrices sont indiquées en gras et majuscules, les vecteurs en minuscules et en gras. La norme l_1 est notée par $\|.\|_1$ et la norme l_2 par $\|.\|_2$. La sphère unité de dimension $(d-1)$ intégrée dans $\mathbb{R}^d$ est noté $\mathbb{S}^{d-1}$. Les données sont représentées par une matrice $\boldsymbol{X} = (x_{ij})$ de dimension $n \times d$ avec $x_{ij} \in \mathbb{R}$ et la i^{eme} ligne de cette matrice est représentée par un vecteur $\boldsymbol{x_i} = (x_{i1}, ..., x_{id})^T$, où T dénote la transposée. La partition de l'ensemble des lignes I en K classes peut être représentées par une matrice de classification $\boldsymbol{Z}$ d'éléments z_{ih} dans $\{0, 1\}$ satisfaisant $\sum_{h=1}^{K} z_{ih} = 1$. On note $\mathbb{I}$ la fonction caractéristique.

2 Mélange de lois de von Mises-Fisher

On rappelle tout d'abord le modèle de mélange proposé dans [Banerjee et al., 2005]. La densité de la loi de von Mises en un point $\boldsymbol{x_i} \in \mathbb{S}^{d-1}$ est donnée par

$$f(\boldsymbol{x_i}|\boldsymbol{\mu}, \kappa) = C_d(\kappa) \exp\left(\kappa \boldsymbol{\mu}^T \boldsymbol{x_i}\right). \tag{1}$$

où $\boldsymbol{\mu}$ est la moyenne directionnelle et κ le paramètre de concentration de la loi, tels que $\|\boldsymbol{\mu}\|_2 = 1$ et $\kappa \geq 0$. Le terme de normalisation $C_d(\kappa)$ est donné par $C_d(\kappa) = \frac{\kappa^{d/2-1}}{(2\pi)^{d/2} I_{d/2-1}(\kappa)}$ où I_r est la fonction de Bessel modifiée du premier type d'ordre r.

On considère un mélange de K lois de van Mises, chacune avec ses propres paramètres, avec la densité [Banerjee et al., 2005]

$$f(\boldsymbol{x_i}|\boldsymbol{\Theta}) = \sum_{h=1}^{K} \alpha_h f(\boldsymbol{x_i}|\boldsymbol{\mu_h}, \kappa_h). \tag{2}$$

où $\Theta = \{\boldsymbol{\mu_1}, \ldots, \boldsymbol{\mu_K}, \alpha_1, \ldots, \alpha_K, \kappa_1, \ldots, \kappa_K\}$. Les observations sont supposées indépendantes. En introduisant les variables latentes $\boldsymbol{Z}$ indiquant (sous forme des indicatrices z_{ih}) la composante du mélange responsable de chaque observation, on obtient la log-vraisemblance suivante pour les données complétées :

$$l(\Theta|\boldsymbol{X}, \boldsymbol{Z}) = \sum_{h=1}^{K} z_{.h}[\log \alpha_h + \log c_d(\kappa_h)] + \sum_{i,h} z_{ih} \kappa_h \boldsymbol{\mu_h}^T \boldsymbol{x_i}, \tag{3}$$

où $z_{.h}$ représente la cardinalité de la classe h. En s'appuyant sur cette vraisemblance complétée, l'utilisation d'un algorithme EM pour l'estimation des paramètres ne pose pas de problème spécifique, excepté l'estimation des κ_h, cf [Banerjee et al., 2005] pour des détails.

3 Modèle proposé

3.1 Vraisemblance pénalisée

Nous proposons de pénaliser la vraisemblance par la norme l_1 permettant ainsi d'augmenter la parcimonie de la représentation des moyennes directionnelles. Plus précisément, nous cherchons à estimer Θ en maximisant la log-vraisemblance pénalisé :

$$l_p(\Theta|\boldsymbol{X}) = l(\Theta|\boldsymbol{X}) - \beta \sum_{h=1}^{K} \|\boldsymbol{\mu_h}\|_1 , \tag{4}$$

où β règle le compromis entre la vraisemblance et la parcimonie. Comme le montre [Pan and Shen, 2007], cette pénalisation n'a pas d'effet sur l'étape E de l'algorithme EM pour un modèle de mélange.

3.2 Phase M de l'algorithme EM

En revanche, la phase est modifiée. Notons $\tau'_{i,h} = \mathbb{P}(z_{ih} = 1|\boldsymbol{x_i}, \Theta')$, où Θ' désigne l'estimation actuelle des paramètres. L'espérance par rapport à $\mathbb{P}(\boldsymbol{Z}|\boldsymbol{X}, \Theta')$ de la log-vraisemblance pénalisée s'écrit alors

$$Q_P(\Theta|\Theta') = \sum_{h=1}^{K} \tau'_{.h}[\log \alpha_h + \log c_d(\kappa_h)] + \sum_{i,h} \tau'_{ih} \kappa_h \boldsymbol{\mu_h}^T \boldsymbol{x_i} - \beta \sum_{h=1}^{K} \|\boldsymbol{\mu_h}\|_1 , \tag{5}$$

où $\tau'_{.h} = \sum_i \tau'_{ih}$. On introduit le Lagrangien

$$\mathcal{L}(\Theta, \boldsymbol{\lambda}|\Theta') = Q_P(\Theta|\Theta') + \sum_h \lambda_h (1 - \boldsymbol{\mu_h}^T \boldsymbol{\mu_h}). \tag{6}$$

Par rapport aux dérivations de [Banerjee et al., 2005], la différence principale vient du calcul du sous-gradient de $\mathcal{L}$ par rapport à $\mu_{h,j}$. On a en effet

$$\partial_{\mu_{hj}} \mathcal{L}(\Theta, \boldsymbol{\lambda}|\Theta') = \kappa_h \sum_i \tau'_{ih} x_{ij} - 2\lambda_h \mu_{hj} - \beta \partial_{\mu_{hj}} |\mu_{hj}|. \tag{7}$$

Dans la dérivation qui suit, nous nous restreignons au cas où les μ_{hj} sont positifs ou nuls, pour une application à des données positives de type textes. Cette dérivation s'étend sans difficulté au cas général.

La condition d'optimalité du premier ordre est $0 \in \partial_{\mu_{hj}} \mathcal{L}(\Theta, \boldsymbol{\lambda}|\Theta')$. En notant $r'_{hj} = \sum_i \tau'_{ih} x_{ij}$, on a :

$$\partial_{\mu_{hj}} \mathcal{L}(\Theta, \boldsymbol{\lambda}|\Theta') = \begin{cases} \kappa_h r'_{hj} - 2\lambda_h \mu_{hj} - \epsilon\beta, \epsilon \in [-1; 1] & \text{si } \mu_{hj} = 0 \\ \kappa_h r'_{hj} - 2\lambda_h \mu_{hj} - \beta & \text{si } \mu_{hj} > 0 \end{cases} \tag{8}$$

On en déduit que $\mu_{hj} = \max\left(\frac{\kappa_h r'_{hj} - \beta}{2\lambda_h}, 0\right)$. En réinjectant cette formule dans la contrainte $\|\boldsymbol{\mu}_h\|_2 = 1$, on trouve

$$\lambda_h = \frac{1}{2}\sqrt{\sum_j (\max(\kappa_h r'_{hj} - \beta, 0))^2}., \tag{9}$$

ce qui permet de conclure que

$$\mu_{hj} = \max\left(\frac{\kappa_h r'_{hj} - \beta}{\sqrt{\sum_l (\max(\kappa_h r'_{hl} - \beta, 0))^2}}, 0\right). \tag{10}$$

Notons que l'ajout de la pénalisation introduit un couplage entre κ_h et $\boldsymbol{\mu}_h$ qui n'existe pas en son absence (on voit que si on fixe $\beta = 0$, κ_h n'intervient plus dans la définition de $\boldsymbol{\mu}_h$). On doit donc résoudre

$$\frac{c'_d(\kappa_h)}{c_d(\kappa_h)} = -\frac{\boldsymbol{\mu}_h \sum_i \tau'_{ih} \boldsymbol{x}_i}{\sum_i \tau'_{ih}}. \tag{11}$$

Nous reprenons l'approximation proposée dans [Banerjee et al., 2005]. Si on pose $\bar{r}'_h = \frac{\boldsymbol{\mu}_h \sum_i \tau'_{ih} \boldsymbol{x}_i}{\sum_i \tau'_{ih}}$, on estime κ_h par

$$\kappa_h = \frac{\bar{r}'_h d - (\bar{r}'_h)^3}{1 - (\bar{r}'_h)^2}. \tag{12}$$

On propose d'estimer $\boldsymbol{\mu}_h$ à partir de κ'_h, puis de mettre à jour κ_h.

3.3 Sélection de modèle

Nous proposons de sélectionner le modèle retenu pour un jeu de données en utilisant le critère BIC. Seuls les paramètres non nuls pour μ_{hj} sont considérés comme des paramètres effectifs. On a donc

$$BIC = -2 \times l(\hat{\Theta}|\boldsymbol{X}) + C \times \log(n), \tag{13}$$

avec pour C le nombre de paramètres la valeur $C = (K - 1 + K) + \sum_h \sum_j \mathbb{I}_{\mu_{hj} \neq 0}$.

4 Résultats expérimentaux

Pour obtenir les résultats expérimentaux, les équations 9, 10 et 13 sont implémentées à l'aide du *package* R *movMF*[1]. L'algorithme est ainsi testé sur un jeu de données textuelles en comparaison avec le modèle initial. Pour comparer les modèles nous avons choisi d'utiliser le Adjusted Rand Index.

1. https ://cran.r-project.org/web/packages/movMF/index.html

Nous avons sélectionné un jeu de données populaires pour tester notre algorithme à savoir CSTR [Li, 2005] [2] avec les caractéristiques suivantes $(n, d, g) = (475, 1000, 4)$. Il est composé de résumés de rapports techniques (TR) publiés au Département d'informatique de l'Université de Rochester entre 1991 et 2002. De plus, il a été divisé en quatre catégories qui sont *Natural Language Processing(NLP)*, *Robotics/Vision*, *Systems*, et *Theory*.

Pour commencer l'analyse, il est intéressant de s'attarder sur les modèles sélectionnés par le BIC. Pour le modèle movMF original, le BIC a été calculé sur le modèle qui maximise la vraisemblance pour chaque classe. Pour celui pénalisé, les méta-paramètres β et K sont estimés avec le BIC. Les résultats sont visibles sur les figures 1 et 2 où l'on remarque que le BIC a sélectionné dans les deux cas, les modèles avec quatre classes. De plus, sur la figure 2, on distingue que les β sont très différents selon les modèles et que la parcimonie évolue en conséquence où elle atteint un maximum pour le modèle sélectionné.

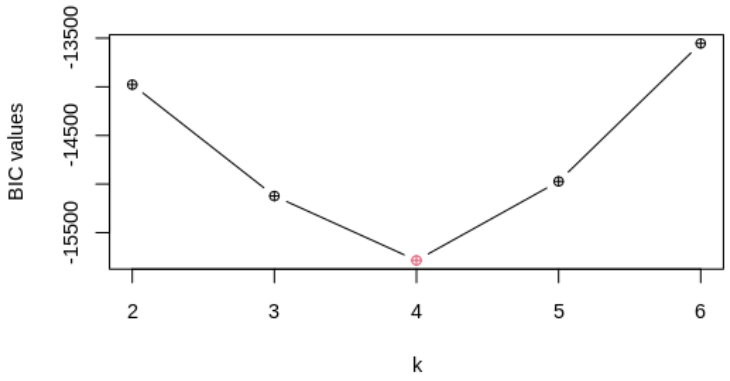

FIGURE 1 – Valeur du BIC pour movMF selon k.

La table 1 montre quant à elle les résultats obtenus avec les modèles sélectionnés par le BIC. Le modèle pénalisé avec un $\beta = 142$ obtient un ARI supérieur au movMF et permet une grande parcimonie de la moyenne directionnelle.

TABLE 1 – Résultats.

Algo	ARI	Parcimonie
movMF	63%	0%
movMF pénalisé	**72%**	**67%**

2. Disponible ici : https ://github.com/dbmovMFs/DirecCoclus/tree/master/Data

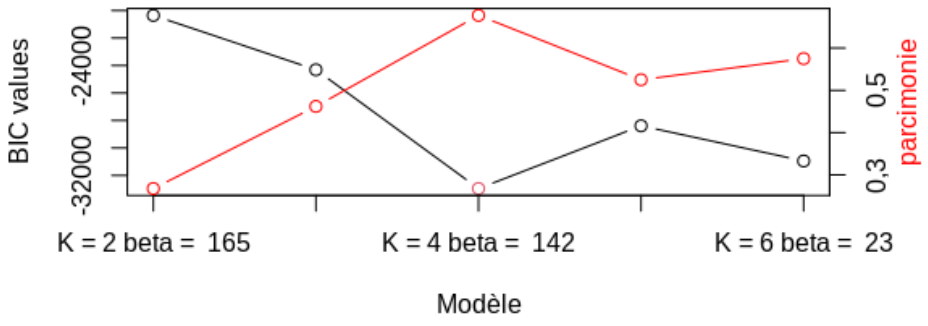

FIGURE 2 – Valeurs du BIC et de la parcimonie pour movMF pénalisé selon k et β.

Références

[Banerjee et al., 2005] Banerjee, A., Dhillon, I. S., Ghosh, J., and Sra, S. (2005). Clustering on the unit hypersphere using von mises-fisher distributions. *Journal of Machine Learning Research*, 6(46) :1345–1382.

[Gopal and Yang, 2014] Gopal, S. and Yang, Y. (2014). Von mises-fisher clustering models. In Xing, E. P. and Jebara, T., editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 154–162, Bejing, China. PMLR.

[Li, 2005] Li, T. (2005). A general model for clustering binary data. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, page 188–197, New York, NY, USA. Association for Computing Machinery.

[Mardia and Jupp, 2009] Mardia, K. and Jupp, P. (2009). *Directional Statistics*. Wiley Series in Probability and Statistics. Wiley.

[Pan and Shen, 2007] Pan, W. and Shen, X. (2007). Penalized model-based clustering with application to variable selection. *Journal of Machine Learning Research*, 8(41) :1145–1164.