Contributions à l'apprentissage statistique en grande dimension, adaptatif et sur données atypiques

Charles Bouveyron

Université Paris 1 Panthéon–Sorbonne

Mémoire

présenté par

Charles BOUVEYRON

pour l'obtention de

l'Habilitation à Diriger des Recherches

Spécialité : Mathématiques Appliquées

Contributions à l'apprentissage statistique en grande dimension, adaptatif et sur données atypiques

---

mémoire soutenu publiquement le 29 novembre 2012

JURY

Christophe	AMBROISE	Professeur, Université d'Evry	Examinateur
Gilles	CELEUX	Directeur de Recherche, INRIA	Examinateur
Marie	COTTRELL	Professeur, Université Paris 1	Examinateur
Adrian	RAFTERY	Professeur, Washington University, USA	Rapporteur
Stéphane	ROBIN	Directeur de Recherche, INRA & AgroParisTech	Rapporteur
Fabrice	ROSSI	Professeur, Université Paris 1	Rapporteur
Jérôme	SARACCO	Professeur, Université Bordeaux 1	Examinateur

Habilitation préparée au sein du laboratoire SAMM, Université Paris 1 Panthéon–Sorbonne

Remerciements

Le travail présenté dans ce mémoire est le fruit de collaborations et interactions avec de nombreuses personnes que je souhaite vivement remercier.

Je souhaite en premier lieu remercier Stéphane Girard qui m'a initié de la meilleure des manières, en tant que directeur de thèse, au métier de la recherche et qui reste aujourd'hui un co-auteur avec qui il est très enrichissant de travailler. Je souhaite également remercier Cordelia Schmid qui m'a permis de cultiver mon goût pour les applications et de découvrir la vision par ordinateur.

Je tiens à remercier Fabrice Rossi qui, en plus d'être un collègue avec qui le travail devient vite passionnant, a bien voulu être mon directeur de recherche dans le cadre de cette HDR et a été le premier à se confronter à mon manuscrit. Je remercie également Adrian Raftery et Stéphane Robin qui m'ont fait l'honneur de rapporter mon mémoire d'HDR et dont les conseils scientifiques sont toujours d'une grande justesse. Mes remerciements vont aussi à Christophe Ambroise, Gilles Celeux et Jérôme Saracco qui m'ont fait l'honneur et l'amitié de participer à mon jury d'HDR. Je remercie enfin Marie Cottrell pour sa participation à mon jury, qui fut une excellente directrice de laboratoire et qui demeure une collègue des plus agréable.

Je veux également exprimer ma gratitude envers les collègues des différents laboratoires par lesquels je suis passé ces dernières années. Par ordre chronologique, je remercie Michel Dojat de l'U594 de l'INSERM pour son accueil lors de mon stage de DEA. Un grand merci à Florence Forbes et tous les membres de l'équipe MISTIS, qui fut mon laboratoire INRIA d'accueil durant ma thèse, pour tous ces bon moments durant la thèse. Je remercie également les membres de l'équipe SMS du laboratoire LJK pour leur agréable accueil durant ma thèse. Je veux également remercier chaleureusement Hugh Chipman pour m'avoir offert l'opportunité d'un post-doctorat au Canada. Mes remerciements vont également à tous les membres du SAMM (et son annexe d'historiens du Pireh) pour ces (déjà) 6 années très agréables et ponctuées de pots mémorables. Je souhaite également remercier les collègues bordelais (Marie, Vanessa, Jérôme et Benoît), rennais (Julie et François) et lillois (Christophe, Alain et Serge) pour les très agréables moments (scientifiques ou non) passés lors de conférences.

Je souhaite aussi adresser un remerciement spécial à l'ensemble de mes co-auteurs qui ont bien évidement contribué de façon directe à cette habilitation. En particulier, je tiens à remercier à nouveau Stéphane Girard qui est un « co-auteur en or ». Je remercie également Julien Jacques, qui est devenu au fil des années tout autant un ami qu'un co-auteur, et Camille Brunet, qui fut ma première doctorante et à qui je souhaite beaucoup de réussites dans sa nouvelle vie américaine. Merci aussi à Pierre Latouche, Mathieu Fauvel et Laurent Bergé pour les bons moments passés autour d'un article.

Il m'est bien sûr impossible de ne pas remercier mes amis qui ont contribué à ce mémoire de part les moments de détentes : Xavier, Mylène, Sébastien et Timéo, Laure et Caroline, Frédéric et Maxime, Julien, Carole et Louis (pour les week-ends lillois et franc-comtois) et Olivier (pour les sorties escalade et à ski). J'adresse également un remerciement affectueux à mes parents, ma famille et ma belle-famille pour leur présence et leur soutien. J'ai en outre une pensée émue pour mon beau-père, José, qui nous a quitté beaucoup trop tôt.

Mon ultime remerciement va à mon épouse adorée, Nathalie, qui par son soutien et son amour m'a tant aidé dans cette tâche et qui a fait de moi l'heureux papa d'Alexis et Romain. Je lui dédie ce mémoire.

Table des matières

1 Introduction

Ce mémoire rend compte de mes activités de recherche depuis ma thèse de doctorat. Mes travaux s'inscrivent dans le cadre de l'apprentissage statistique et s'articulent plus précisément autour des quatre thématiques suivantes :

- apprentissage statistique en grande dimension,
- apprentissage statistique adaptatif,
- apprentissage statistique sur données atypiques,
- applications de l'apprentissage statistique.

Mes contributions à ces quatre thématiques sont décrites en autant de chapitres, numérotés de 2 à 5, pouvant être lus indépendamment. Ce mémoire se veut également être, en quelque sorte, un plaidoyer pour l'usage des méthodes génératives (reposant sur un modèle probabiliste) en apprentissage statistique moderne. Il sera en effet démontré dans ce document, je l'espère de façon convaincante, que les méthodes génératives peuvent résoudre efficacement les problèmes actuels de l'apprentissage statistique tout en présentant l'avantage de l'interprétabilité des résultats et de la connaissance du risque de prédiction.

Le chapitre 2 est consacré au problème de l'apprentissage statistique en grande dimension. Mes contributions à cette thématique portent en particulier sur la classification supervisée et non supervisée des données de grande dimension avec des modèles latents, sur l'estimation de la dimension intrinsèque de données de grande dimension et sur la sélection de variables discriminantes pour le clustering. L'objet du chapitre 3 est l'apprentissage statistique adaptatif. J'ai notamment contribué à cette thématique par la proposition de méthodes adaptatives pour la régression quand les populations d'apprentissage et de prédiction diffèrent. J'ai également proposé des approches adaptatives pour la classification supervisée avec labels incertains ou avec classes non observées. Le chapitre 4 s'intéresse au problème de l'apprentissage statistique sur données atypiques, *i.e.* avec des données non quantitatives ou non vectorielles. Dans ce contexte, j'ai en particulier travaillé sur la classification supervisée des nœuds d'un réseau et sur le clustering de données fonctionnelles. Une contribution récente aborde ce problème d'un point de vue plus général et la méthodologie proposée permet de faire la classification de données de types très variés, incluant notamment les données qualitatives, les réseaux et les données fonctionnelles. Enfin, le chapitre 5 présente mes contributions à l'application des techniques d'apprentissage statistique à des problèmes réels. Je présenterai notamment des applications à l'analyse d'images, au traitement de données bio-médicales, à l'analyse d'images hyper-spectrales, à la chimiométrie et au *health monitoring* en aéronautique. Les logiciels développés, implantant les méthodes théoriques proposées, sont

également présentés à la fin de ce chapitre.

L'ensemble de ces travaux a donné lieu, à ce jour, à 16 articles dans des journaux internationaux, 1 chapitre d'ouvrage collectif, 3 prépublications et 26 communications dans des conférences internationales (dont 4 invitations). Une partie de ces travaux a été en outre réalisée dans le cadre de deux thèses que j'ai co-encadré ou co-encadre actuellement : la thèse de Camille Brunet, débutée en octobre 2008 et soutenue en décembre 2011, et la thèse CIFRE d'Anastasios Belllas, débutée en janvier 2011. La liste de mes publications est donnée ci-dessous. Les articles sont numérotés, par ordre chronologique, de [B1] jusqu'à [B20] et ces étiquettes seront utilisées pour faire référence à ces publications tout au long du document. Le nombre de citations dans la littérature de chaque article est également indiqué. Ces données de bibliométrie sont extraites de Google Scholar (en date du 01/09/2012) et les auto-citations ont bien entendu été décomptées.

Articles de journaux avec comité de lecture (16)

[B1] C. Bouveyron, S. Girard & C. Schmid, *Class-Specific Subspace Discriminant Analysis for High-Dimensional Data*, In Lecture Notes in Computer Science n.3940, pp. 139-150, Springer-Verlag, 2006. (4 citations)

[B2] C. Bouveyron, S. Girard & C. Schmid, *High-Dimensional Discriminant Analysis*, Communications in Statistics : Theory and Methods, vol. 36 (14), pp. 2607-2623, 2007. (26 citations)

[B3] C. Bouveyron, S. Girard & C. Schmid, *High-Dimensional Data Clustering*, Computational Statistics and Data Analysis, vol. 52 (1), pp. 502-519, 2007. (67 citations)

[B4] C. Bouveyron & S. Girard, *Classification supervisée et non supervisée des données de grande dimension*, La revue Modulad, vol. 40, pp. 81-102, 2009.

[B5] C. Bouveyron & S. Girard, *Robust supervised classification with mixture models : learning from data with uncertain labels*, Pattern Recognition, vol. 42 (11), pp. 2649-2658, 2009. (11 citations)

[B6] C. Bouveyron and J. Jacques, *Adaptive linear models for regression : improving prediction when population has changed*, Pattern Recognition Letters, vol. 31 (14), pp. 2237-2247, 2010. (4 citations)

[B7] C. Bouveyron, O. Devos, L. Duponchel, S. Girard, J. Jacques and C. Ruckebusch, *Gaussian mixture models for the classification of high-dimensional vibrational spectroscopy data*, Journal of Chemometrics, vol. 24 (11-12), pp. 719-727, 2010. (2 citations)

[B8] C. Bouveyron and J. Jacques, *Model-based Clustering of Time Series in Group-specific Functional Subspaces*, Advances in Data Analysis and Classification, vol. 5 (4), pp. 281-300, 2011. (3 citations)

[B9] C. Bouveyron, G. Celeux and S. Girard, *Intrinsic Dimension Estimation by Maximum Likelihood in Probabilistic PCA*, Pattern Recognition Letters, vol. 32 (14), pp. 1706-1713, 2011. (5 citations)

[B10] C. Bouveyron, P. Gaubert and J. Jacques, *Adaptive models in regression for modeling and understanding evolving populations*, Journal of Case Studies in Business, Industry and Government Statistics, vol. 4 (2), pp. 83-92, 2011.

[B11] C. Bouveyron and C. Brunet, *On the estimation of the latent discriminative subspace in the Fisher-EM algorithm*, Journal de la Société Française de Statistique, vol. 152 (3), pp. 98-115, 2011.

[B12] C. Bouveyron and C. Brunet, *Simultaneous model-based clustering and visualization in the Fisher discriminative subspace*, Statistics and Computing, vol. 22 (1), pp. 301-324, 2012. (3 citations)

[B13] C. Bouveyron and C. Brunet, *Probabilistic Fisher discriminant analysis : A robust and flexible alternative to Fisher discriminant analysis*, Neurocomputing, vol. 90 (1), pp. 12-22, 2012.

[B14] L. Bergé, C. Bouveyron and S. Girard, *HDclassif : an R Package for Model-Based Clustering and Discriminant Analysis of High-Dimensional Data*, Journal of Statistical Software, vol. 42 (6), pp. 1-29, 2012. (3 citations)

[B15] C. Bouveyron and C. Brunet, *Theoretical and practical considerations on the convergence properties of the Fisher-EM algorithm*, Journal of Multivariate Analysis, vol. 109, pp. 29-41, 2012.

[B16] C. Bouveyron, *Adaptive mixture discriminant analysis for supervised learning with unobserved classes*, Journal of Classification, à paraître, 2013.

Chapitres de livres (1)

[B17] F.Beninel, C.Biernacki, C.Bouveyron, J.Jacques and A.Lourme, *Parametric link models for knowledge transfer in statistical learning*, in Knowledge Transfer : Practices, Types and Challenges, Ed. Dragan Ilic, Nova Publishers, 2012.

Prépublications (3)

[B18] C. Bouveyron and J. Jacques, *Adaptive mixtures of regressions : Improving predictive inference when population has changed*, Preprint IRMA Lille Vol. 70-VIII, Preprint HAL n°00477597, 2010. (3 citations)

[B19] C. Bouveyron, S. Girard and M. Fauvel, *Kernel discriminant analysis and clustering with parsimonious Gaussian process models*, Preprint HAL n°00687304, Laboratoire SAMM, Université Paris 1 Panthéon-Sorbonne, 2012.

[B20] C. Bouveyron and C. Brunet, *Discriminative variable selection for clustering with the sparse Fisher-EM algorithm*, Preprint HAL n°00685183, Laboratoire SAMM, Université Paris 1 Panthéon-Sorbonne, 2012.

2

Apprentissage statistique en grande dimension

Cette première thématique de recherche porte sur la modélisation et la classification, supervisée et non supervisée, des données de grande dimension et fait directement suite aux travaux développés durant ma thèse de doctorat. Cet axe de recherche peut-être organisé en quatre sous-axes qui sont développés ci-dessous. Le premier traite des modèles parcimonieux pour la classification en grande dimension et a donné lieu à 3 articles méthodologiques [B1, B2, B3] et 1 article applicatif [B7]. Le second sous-axe, dédié à l'estimation de la dimension intrinsèque d'un jeu de données, répond à un problème soulevé par les travaux menés en classification en grande dimension et a donné lieu à 1 article méthodologique [B9]. Le troisième sous-axe s'intéresse à la classification non supervisée (appelée également clustering) discriminative qui permet à la fois de partitionner les données et de visualiser la partition obtenue. Ces travaux ont donné lieu à 3 articles méthodologiques [B11, B12, B15]. Enfin, le quatrième sous-axe est consacré à des travaux récents sur la sélection de variables discriminantes en classification non supervisée par pénalisation ℓ_1. Une prépublication [B20] est issue de ces travaux. Ces deux derniers sous-axes ont notamment été développés à l'occasion de l'encadrement de la thèse de Camille Brunet.

2.1 Modèles parcimonieux pour la classification en grande dimension

La modélisation et la classification des données de grande dimension est un problème important que l'on retrouve dans de nombreux domaines applicatifs. Parmi les méthodes probabilistes [16] de classification, les approches basées sur le modèle de mélange gaussien sont probablement les plus populaires [64]. Malheureusement, le comportement du modèle de mélange gaussien s'avère être décevant dans la pratique lorsque la taille de l'échantillon est faible au regard de la dimension. Ce phénomène bien connu est appelé *« curse of dimensionality »* [9] (fléau de la dimension). On pourra consulter [76, 77] pour une étude théorique de l'effet de la dimension en classification supervisée (ou analyse discriminante).

La classification supervisée vise à associer chacune des n observations $\{y_1, ..., y_n\} \in \mathcal{Y}$ à l'une des K classes connues a priori (au travers d'un classifieur appris sur un jeu dit d'apprentissage) tandis que la classification non supervisée a pour but de regrouper ces données en K groupes homogènes sans autre connaissance. Le lecteur pourra trouver de plus amples détails sur ces deux types de classification dans [62] et [63]. Dans ces deux situations, l'approche basée sur le modèle

de mélange gaussien suppose que les observations $\{y_1, ..., y_n\}$ sont des réalisations indépendantes d'un vecteur aléatoire Y à valeurs dans $\mathbb{R}^p$ de densité :

$$f(y, \theta) = \sum_{k=1}^{K} \pi_k \phi(u, \theta_k), \tag{2.1}$$

où ϕ est la densité de la loi normale multivariée de paramètres $\theta_k = \{\mu_k, \Sigma_k\}$ et π_k est la probabilité a priori de la $k$ème classe. Dans ce cadre, le fléau de la dimension se traduit notamment par la nécessité d'estimer des matrices de covariance pleines et cela implique que le nombre de paramètres à estimer croît avec le carré de la dimension p. Les solutions classiques pour pallier ce problème incluent la réduction de dimension [33, 50, 86], la régularisation [38, 44] et l'usage de modèles parcimonieux [10, 37]. Cependant, le phénomène de « l'espace vide » [88] nous permet de conjecturer que la classification est une tâche plus aisée à réaliser dans des espaces de grande dimension. Nous avons par conséquent proposé dans [B2] et [B3] une paramétrisation du modèle de mélange gaussien qui permet d'exploiter cette caractéristique des espaces de grande dimension. Notre idée est d'utiliser le fait que les données de grande dimension vivent dans des sous-espaces dont les dimensions intrinsèques sont faibles afin de limiter le nombre de paramètres du modèle et de régulariser l'estimation des matrices de covariance des classes.

2.1.1 Une famille de modèles gaussiens parcimonieux

Nous nous plaçons dans le cadre classique du modèle de mélange gaussien et nous supposons donc que les densités conditionnelles des K classes sont gaussiennes $\mathcal{N}_p(\mu_k, \Sigma_k)$ de moyennes μ_k et de matrices de covariance Σ_k, pour $k = 1, ..., K$. Soit Q_k la matrice orthogonale composée des vecteurs propres de Σ_k, alors la matrice de covariance Δ_k est définie dans l'espace propre de Σ_k de la manière suivante :

$$\Delta_k = Q_k^t \, \Sigma_k \, Q_k. \tag{2.2}$$

La matrice Δ_k est, par construction, une matrice diagonale contenant les valeurs propres de Σ_k. Nous supposons en outre que Δ_k n'a que $d_k + 1$ valeurs propres différentes et a donc la forme suivante :

$$\Delta_k = \begin{pmatrix} \boxed{\begin{matrix} a_{k1} & & 0 \\ & \ddots & \\ 0 & & a_{kd_k} \end{matrix}} & & \mathbf{0} \\ & \boxed{\begin{matrix} b_k & & 0 \\ & \ddots & \\ & & \ddots \\ 0 & & b_k \end{matrix}} & \\ \mathbf{0} & & \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} d_k \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} (p - d_k) \end{matrix} \tag{2.3}$$

avec $a_{kj} > b_k$, $j = 1, ..., d_k$, et où $d_k \in \{1, \ldots, p-1\}$ est inconnu. En introduisant la variable aléatoire $Z \in \{1, ..., K\}$ telle que z_i indique la classe d'origine de l'observation y_i, cette modélisation revient à supposer qu'il existe une variable latente X qui, conditionnellement à $Z = k$, vit dans un sous-espace E_k défini comme étant l'espace affine engendré par les d_k vecteurs propres associés aux valeurs propres a_{kj} et tel que $\mu_k \in E_k$. Ainsi, la dimension d_k du sous-espace E_k peut être considérée comme la dimension intrinsèque de la $k$ème classe. Dans l'orthogonal du sous-espace E_k, la variance est celle d'un bruit gaussien ε isotropique et donc modélisé par l'unique paramètre b_k. Par conséquent, les variables aléatoires X et Y sont liées, conditionnellement à $Z = k$, par la relation linéaire :

$$Y_{|Z=k} = Q_k X_{|Z=k} + \varepsilon_{|Z=k}.$$

Ce modèle gaussien parcimonieux sera noté $[a_{kj} b_k Q_k d_k]$ dans la suite. La figure 2.1 résume ces notations. En imposant certains paramètres à être communs entre les classes ou dans une même

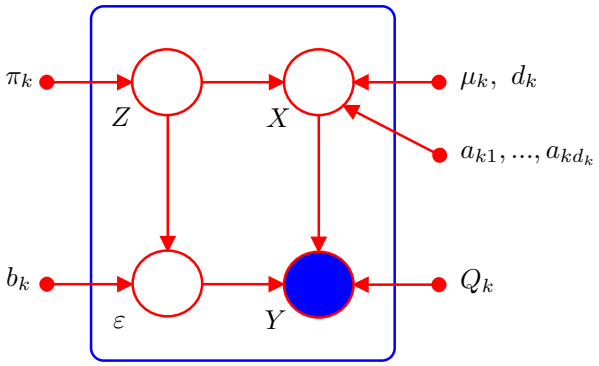


FIGURE 2.1: Représentation graphique du modèle gaussien parcimonieux $[a_{kj}b_kQ_kd_k]$.

classe, nous avons obtenu une famille de 28 modèles parcimonieux qui correspondent à différentes régularisations du modèle $[a_{kj}b_kQ_kd_k]$ (*cf.* [B1] pour une description détaillée).

2.1.2 Inférence et classification dans le cas supervisé : la méthode HDDA

L'utilisation dans le cadre de la classification supervisée des modèles gaussiens pour la grande dimension a donné naissance à la méthode *high-dimensional discriminant analysis* (HDDA) [B2]. Dans ce contexte, les données d'apprentissage étant complètes, *i.e.* un label z_i indiquant la classe d'appartenance est associé à chaque observation y_i, l'estimation des paramètres du modèle par maximum de vraisemblance est directe et conduit aux estimateurs suivants. Les proportions du mélange ainsi que les moyennes sont respectivement estimées par $\hat{\pi}_k = n_k/n$ et $\hat{\mu}_k = \frac{1}{n_k}\sum_{i/z_i=k} y_i$ où n_k est le nombre d'individus dans la $k$ème classe. L'estimation des paramètres spécifiques du modèle introduit précédemment est détaillée par la proposition suivante.

Proposition 1. *Les estimateurs du maximum de vraisemblance des paramètres du modèle* $[a_{kj}b_kQ_kd_k]$ *sont explicites et donnés par :*

- *les* d_k *premières colonnes de* Q_k *sont estimées par les vecteurs propres associés aux* d_k *plus grandes valeurs propres* λ_{kj} *de la matrice de covariance empirique* $S_k = \frac{1}{n_k}\sum_{i/z_i=k}(y_i - \hat{\mu}_k)(y_i - \hat{\mu}_k)^t$.
- *l'estimateur de* a_{kj} *est* $\hat{a}_{kj} = \lambda_{kj}$, $j = 1, ..., d_k$
- *l'estimateur de* b_k *est* $\hat{b}_k = \frac{1}{(p-d_k)}\left(\mathrm{trace}(S_k) - \sum_{j=1}^{d_k}\lambda_{kj}\right)$.

La démonstration de ce résultat est donnée dans [B2]. De façon classique, la classification d'une nouvelle observation $y \in \mathbb{R}^p$ se fait grâce à la règle du *maximum a posteriori* (MAP) qui affecte l'observation y à la classe la plus probable a posteriori. Ainsi, l'étape de classification consiste principalement à calculer $P(Z = k|Y = y)$ pour chaque classe $k = 1, ..., K$ et cette probabilité peut être calculée de la façon suivante dans le cas du modèle $[a_{kj}b_kQ_kd_k]$.

Proposition 2. *Dans le cas du modèle* $[a_{kj}b_kQ_kd_k]$*, la règle du MAP affecte une observation* y *à la classe de plus grande probabilité* $P(Z = k|Y = y) = 1/\sum_{\ell=1}^{K}\exp\left(\frac{1}{2}(D_k(y) - D_\ell(y))\right)$ *où la fonction de classification* $D_k(y) = -2\log(\pi_k\phi(y, \theta_k))$ *a la forme suivante :*

$$D_k(y) = \|\mu_k - P_k(y)\|^2_{\mathcal{A}_k} + \frac{1}{b_k}\|y - P_k(y)\|^2 + \sum_{j=1}^{d_k}\log(a_{kj}) + (p - d_k)\log(b_k) - 2\log(\pi_k) + \gamma,$$

avec $||y||^2_{\mathcal{A}_k} = y^t\mathcal{A}_k y$, $\mathcal{A}_k = \tilde{Q}_k^t\Delta_k^{-1}\tilde{Q}_k$, $\tilde{Q}_k$ *est la matrice composée des* d_k *premières colonnes de* Q_k *complétée par des zéros,* $P_k(y) = \tilde{Q}_k^t\tilde{Q}_k(y - \mu_k) + \mu_k$ *et* $\gamma = p\log(2\pi)$.

La démonstration de ce résultat est également donnée dans [B2]. Il est important de remarquer que la paramétrisation du modèle $[a_{kj}b_kQ_kd_k]$ permet d'obtenir une expression explicite de Σ_k^{-1} alors que les méthodes classiques doivent réaliser numériquement cette inversion et échouent généralement du fait de la singularité de la matrice. De plus, en observant l'expression de $D_k(y)$, on remarque que le calcul des probabilités a posteriori n'utilise pas la projection sur l'orthogonal de E_k et par conséquent ne nécessite pas le calcul des $(p-d_k)$ dernières colonnes de la matrice d'orientation Q_k. La méthode HDDA ne dépend donc pas de la détermination de ces axes associés aux plus petites valeurs propres dont l'estimation en grande dimension est généralement instable. Ainsi, la méthode de classification HDDA n'est pas perturbée par l'éventuel mauvais conditionnement ou la singularité des matrices de covariance empiriques des classes. En outre, le fait de n'avoir qu'à déterminer les d_k plus grandes valeurs propres ainsi que leur vecteur propre associé permet de traiter le cas $n < p$ en travaillant sur la matrice $\bar{X}_k\bar{X}_k^t$, de taille $n \times n$, au lieu de $S_k = \bar{X}_k^t\bar{X}_k$, de taille $p \times p$.

2.1.3 Inférence et classification dans le cas non supervisé : l'algorithme HDDC

L'utilisation dans le cadre de la classification non supervisée des modèles gaussiens pour la grande dimension a donné naissance à la méthode *high-dimensional data clustering* (HDDC) [B3]. Dans ce contexte, les données d'apprentissage n'étant pas complètes, *i.e.* le label z_i indiquant la classe d'appartenance est manquant pour chaque observation y_i, l'estimation des paramètres du modèle par maximum de vraisemblance n'est pas directe et nécessite l'utilisation d'un algorithme itératif : l'algorithme EM [29]. Le lecteur pourra consulter [63] pour plus de détails sur l'algorithme EM et ses extensions. En particulier, les modèles présentés dans cet article peuvent également être combinés avec les algorithmes *classification EM* (CEM) [23] et *stochastic EM* (SEM) [22]. Le résultat suivant présente l'algorithme EM dans le cas du modèle $[a_{kj}b_kQ_kd_k]$.

Proposition 3. *Avec les hypothèses et notations du modèle $[a_{kj}b_kQ_kd_k]$, l'algorithme EM prend la forme suivante à l'itération q :*

- *étape E : cette étape calcule, pour $k = 1, ..., K$ et $i = 1, ..., n$, la probabilité conditionnelle $t_{ik}^{(q)} = P(Z = k|Y = y_i)$ comme suit :*

$$t_{ik}^{(q)} = 1 \Bigg/ \sum_{\ell=1}^{K} \exp\left(\frac{1}{2}(D_k^{(q-1)}(y_i) - D_\ell^{(q-1)}(y_i))\right),$$

  avec $D_k^{(q-1)}(y) = -2\log(\pi_k^{(q-1)}\phi(y, \theta_k^{(q-1)}))$ et où $\pi_k^{(q-1)}$ et $\theta_k^{(q-1)}$ sont les paramètres du modèle estimés dans l'étape M à l'itération $(q-1)$.
- *étape M : cette étape maximise l'espérance de la vraisemblance complétée conditionnellement aux $t_{ik}^{(q)}$. Les estimateurs des proportions du mélange et des moyennes sont $\hat{\pi}_k^{(q)} = n_k^{(q)}/n$ et $\hat{\mu}_k^{(q)} = \frac{1}{n_k^{(q)}}\sum_{i=1}^n t_{ik}^{(q)} y_i$ où $n_k^{(q)} = \sum_{i=1}^n t_{ik}^{(q)}$. Les estimateurs de a_{kj}, b_k et Q_k à l'itération q sont ceux donnés à la proposition 1 avec $S_k^{(q)} = \frac{1}{n_k^{(q)}}\sum_{i=1}^n t_{ik}^{(q)}(y_i - \hat{\mu}_k^{(q)})(y_i - \hat{\mu}_k^{(q)})^t$.*

La démonstration de ces résultats est faite dans [B3].

2.1.4 Estimation des hyper-paramètres

L'estimation des paramètres des modèles parcimonieux présentés ci-dessus, que ce soit dans le cadre supervisé ou non supervisé, requiert la connaissance de la dimension intrinsèque d_k de chaque classe. L'estimation des dimensions intrinsèques est un problème difficile pour lequel il n'y a pas de solution universelle. L'approche que nous avons proposé est basée sur les valeurs propres de

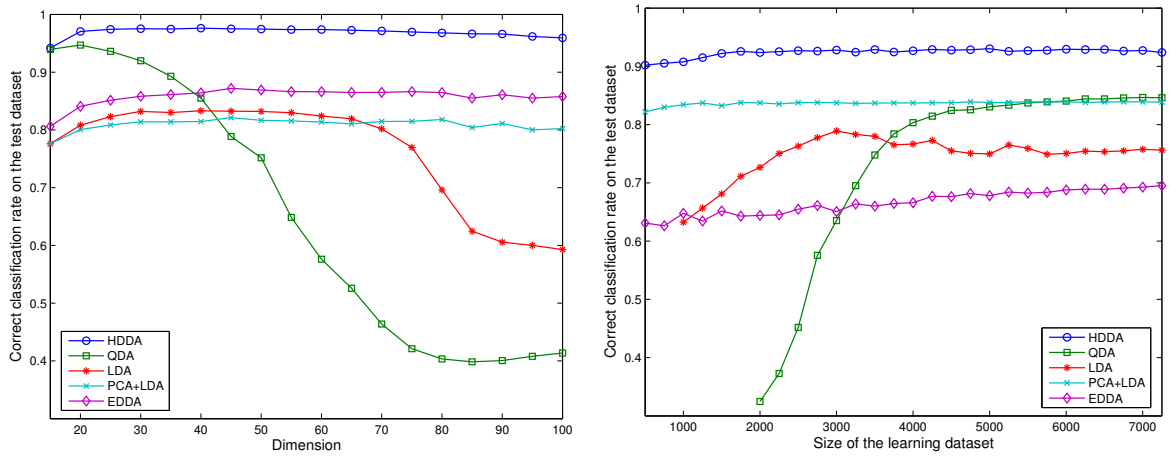


FIGURE 2.2: Taux de classification correcte en fonction de la dimension p de l'espace d'observation sur données simulées (à gauche) et en fonction de la taille du jeu d'apprentissage sur les données USPS (à droite) de HDDA, QDA, LDA, PCA+LDA et EDDA.

la matrice de covariance empirique S_k de chacune des classes. En effet, la $j$ème valeur propre de S_k correspond à la part de la variance totale portée par son $j$ème vecteur propre. Nous avons donc proposé d'estimer la dimension intrinsèque d_k, $k = 1, ..., K$ grâce au *scree-test* de Cattell [21] qui recherche un coude dans l'éboulis des valeurs propres de S_k. La dimension sélectionnée est la dimension pour laquelle les différences entre les valeurs propres sont plus petites qu'un seuil et nous recommandons en pratique de fixer ce seuil à 0.2 fois la valeur de la plus grande différence. Cette approche permet en outre de sélectionner, avec un même seuil, des dimensions d_k potentiellement différentes. Dans le cas non supervisé, il est également nécessaire de déterminer le nombre K de composantes du mélange et cela peut être fait grâce aux critères AIC [2], BIC [87] ou ICL [14].

2.1.5 Expérimentations numériques

Dans [B2], nous avons tout d'abord étudié l'influence de la dimension de l'espace d'observation et de la taille de l'échantillon d'apprentissage sur la performance de classification de HDDA. Pour cela, nous avons simulé des données selon le modèle $[a_k b_k Q_k d_k]$ avec 3 classes de dimension intrinsèque respective 2, 5 et 10. Les échantillons d'apprentissage et de validation étaient respectivement composés de 250 et 1000 observations. La figure 2.2 présente à gauche les taux de classification correcte de HDDA, QDA, LDA, PCA+LDA et EDDA [10], moyennés sur 50 réplications, en fonction de la dimension p de l'espace d'observation. Comme attendu, QDA et LDA s'avèrent être sensibles à la dimension. Les méthodes PCA+LDA et EDDA, qui utilisent respectivement la réduction de dimension et la parcimonie pour pallier le fléau de la dimension, ont des performances qui ne se détériorent pas avec l'augmentation de p. Cependant, HDDA présente des résultats de classification très supérieurs tout en n'étant également pas sensible à l'augmentation de la dimension. Nous avons étudié d'autre part l'influence de taille de l'échantillon d'apprentissage sur la performance de classification de HDDA et de ses concurrents sur un jeu de données réelles (les données USPS). Ce jeu de données est composé de 10 classes, correspondant aux chiffres 0, 1, ..., 9, et chaque observation est la vectorisation d'une image 16×16 en niveaux de gris d'un chiffre manuscrit. La base de données contient 7291 observations d'apprentissage et 2007 observations de validation décrites dans un espace à 256 dimensions. La figure 2.2 présente à droite les taux de classification correcte de HDDA, QDA, LDA, PCA+LDA et EDDA, moyennés sur 50 réplications, en fonction de la taille de l'échantillon d'apprentissage. A nouveau, QDA et LDA s'avèrent sensibles à la taille de l'échantillon d'apprentissage. Comme précédemment, PCA+LDA et EDDA

Méthode	CCR	Modèle utilisé	Temps de calcul
K-means	0.340	–	<0.1 sec.
Mclust	0.575	VEV, 5 var.	0.2 sec.
Mclust sur CP	0.605	EEE, 5 var.	0.2 sec.
Clustvarsel	0.925	EEV, 4 var.	5.1 sec.
Clustvarsel sur CP	0.935	EEV, 3 var.	3.3 sec.
HDDC	0.945	$[a_{kj}b_kQ_kd_k]$, $d = 1$	1.8 sec.

TABLE 2.1: Performance de clustering (CCR) et temps de calcul de k-means, Mclust, Clustvarsel et HDDC sur les données Crabes (CP indique que le clustering a été réalisé sur les composantes principales).

apparaissent relativement robustes tout comme HDDA à la taille de l'échantillon d'apprentissage mais HDDA les surpasse à nouveau en terme de performance. Dans [B7], nous avons en outre mené une comparaison avec la méthode discriminative SVM [85] sur des données de chimiométrie, comportant plusieurs milliers de variables, et ces résultats seront présentés au paragraphe 5.3.

Dans [B3], nous avons également étudié les caractéristiques de HDDC dans différentes situations pratiques. Nous avons notamment considéré le clustering du jeu de données Crabes qui est composé de 5 mesures morphologiques faites sur 200 individus équi-répartis dans 4 groupes correspondants à 4 espèces de crabes. Ce jeu de données présente l'intérêt que les dimensions intrinsèques des groupes sont estimées égales à 1 et il est alors possible de visualiser les sous-espaces estimés par HDDC. La figure 2.3 propose de visualiser, sur les deux premiers axes de l'ACP, à la fois la partition des données en 4 groupes et les sous-espaces des groupes estimés par HDDC à différentes étapes de l'algorithme. La partition initiale a été fournie par l'algorithme k-means. L'adéquation entre la partition obtenue après convergence d'HDDC et la partition connue pour ces données est ici égale à 94.5%. Nous avons ensuite comparé, sur ces mêmes données, la performance de clustering de HDDC avec les méthodes de référence k-means, Mclust [36] et Clustvarsel [79], les deux dernières utilisant respectivement la parcimonie et la sélection de variables. Le tableau 2.1 présente la performance de clustering (adéquation entre la partition obtenue et la partition connue) et le temps de calcul des quatre méthodes de clustering pour le jeux de données considéré. Il apparaît que HDDC se compare favorablement aux méthodes de référence, à la fois en terme de performance de clustering mais aussi au niveau du temps de calcul. Notons que Clustvarsel réalise en outre une sélection de variables qui peut s'avérer très utile du point de vue de l'interprétation. Nous discuterons à la fin de ce chapitre de la possibilité de réaliser une telle sélection de variables avec des méthodes de classification dans des sous-espaces, telles que HDDA et HDDC. Dans [B3], HDDC a également été appliqué au clustering de données hyper-spectrales de dimension 256 pour la catégorisation du sol de la planète Mars, application qui sera présentée au chapitre 5.

La famille de modèles gaussiens parcimonieux proposée dans [B2] et [B3] a donné naissance à deux méthodes de classification, l'une supervisée (HDDA), l'autre non supervisée (HDDC), qui ont montré beaucoup d'intérêts pratiques. En effet, HDDA et HDDC se sont avérées performantes à la fois dans les espaces de grande dimension mais également quand la taille du jeu d'apprentissage est petite. De plus, les résultats fournis par les deux méthodes peuvent aider à la compréhension du phénomène sous-jacent, comme l'a illustré la visualisation des sous-espaces des groupes sur les données Crabes. Les méthodes HDDA et HDDC ont en outre été appliquées à de nombreux domaines applicatifs comme la classification d'images hyper-spectrales [B3] et de données de chimiométrie [B7].

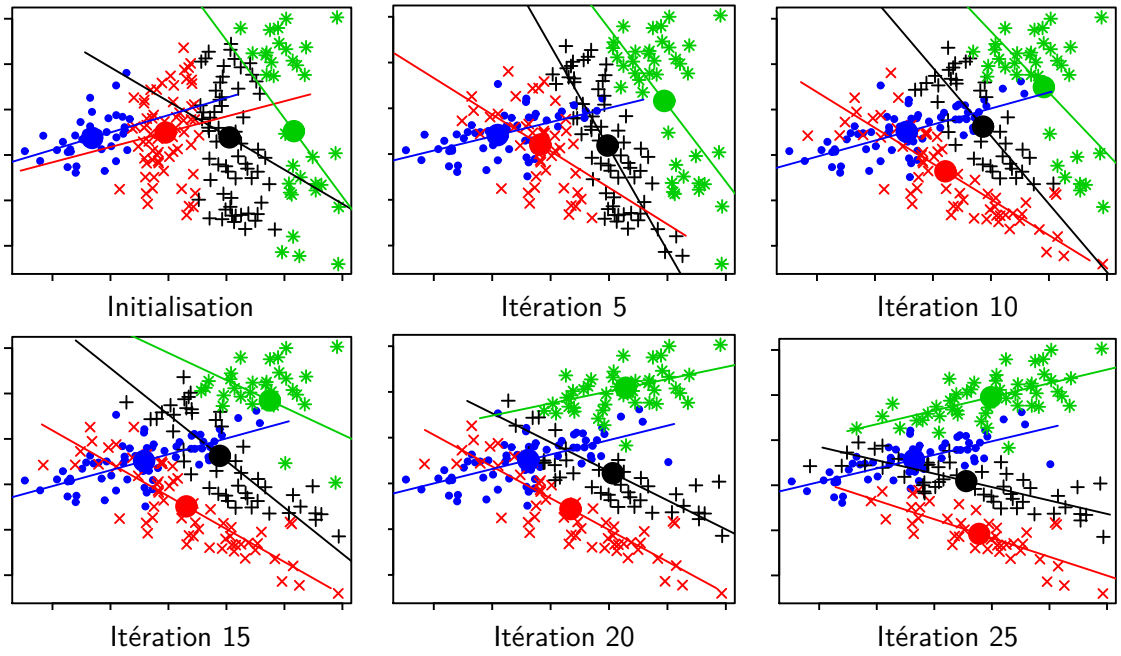


FIGURE 2.3: Partition des données Crabes en 4 groupes avec HDDC à différentes étapes de l'algorithme. Les segments de droite représentent les sous-espaces estimés des groupes. La visualisation est faite dans le plan principal.

2.2 Estimation de la dimension intrinsèque

Nous avons vu, dans le paragraphe précédent, que les dimensions intrinsèques d_k des classes jouent un rôle clé dans les méthodes HDDA et HDDC puisqu'elles contrôlent la complexité des modèles. Cela est également vrai pour les autres méthodes de classification dans des sous-espaces, telles que [65, 67, 71, 97], et l'on peut regretter que le problème du choix de la dimension intrinsèque des classes n'ait pas été traité dans certains de ces travaux. Nous considérons ici, pour des raisons de simplification, le choix de la dimension intrinsèque de données de grande dimension en dehors du contexte de classification (on peut voir cela comme la modélisation d'une unique classe). Les résultats obtenus seront toutefois facilement exploitables dans le contexte du paragraphe précédent en appliquant le modèle considéré ici à chacune des K classes.

L'estimation de la dimension intrinsèque, que l'on notera d^*, d'un jeu de données de grande dimension est un problème difficile qui a été considéré selon différents points de vue. Parmi les approches basées sur la vraisemblance, on peut citer les travaux de Minka [70], de Bickel & Levina [53] et de Fan *et al.* [32]. Minka a proposé, dans un cadre bayésien, une expression explicite de l'approximation de Laplace de la vraisemblance marginale. Nous rappelons que le critère BIC [87] en est une approximation asymptotique. Bickel et Levina ont, quant à eux, modélisé le nombre d'observations dans une sphère de petite dimension par un processus de Poisson ce qui leur permet de déduire un estimateur du maximum de vraisemblance de d^*. Enfin, Fan *et al.* se basent sur un modèle de régression polynomiale avec une hypothèse de distribution uniforme pour estimer la dimension intrinsèque d^*. Dans [B9], nous avons considéré l'estimation par maximum de vraisemblance de d^* dans le cas d'un sous-modèle contraint du modèle *probabilistic principal component analyzer* (PPCA) [98] et nous avons démontré le résultat, a priori surprenant, que l'estimateur du maximum de vraisemblance de la dimension intrinsèque est asymptotiquement optimal dans le cas de ce modèle.

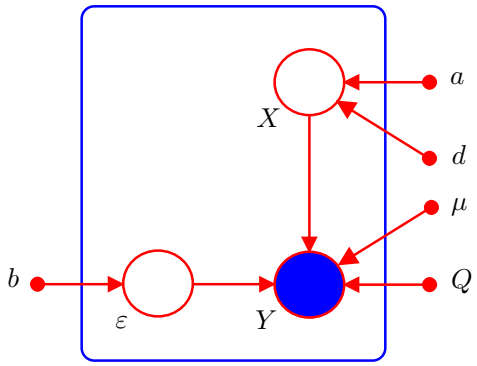


FIGURE 2.4: Représentation graphique du modèle PPCA isotropique.

2.2.1 Le modèle PPCA isotropique

Le modèle PPCA, proposé par Tipping & Bishop [98], est une version probabiliste de l'analyse en composantes principales (ACP) qui justifie, sous hypothèse gaussienne, les résultats classiques d'ACP. Ce modèle, qui est devenu populaire de part son usage en classification non supervisée [97], suppose l'existence d'une relation linéaire entre la variable observée $Y \in \mathbb{R}^p$ et la variable latente $X \in \mathbb{R}^d$:

$$Y = UX + \varepsilon,$$

où U est une matrice $p \times d$ de projection et ε un bruit gaussien. Le modèle PPCA suppose en outre que X est gaussien de sorte que la distribution marginale de Y soit $\mathcal{N}(U\mu, \Sigma)$ avec $\Sigma = UU^t + bI_p$. On montre que l'estimateur du maximum de vraisemblance de U est la matrice de d premiers vecteurs propres de la matrice de covariance empirique. Ce modèle suppose donc implicitement que la matrice de covariance Σ a seulement $d+1$ valeurs propres différentes. Le modèle PPCA isotropique, que nous avons proposé dans [B9], contraint un peu plus la modélisation en supposant que Σ a uniquement deux valeurs propres différentes, *i.e.* :

$$\Sigma = Q \begin{pmatrix} \begin{bmatrix} a & & 0 \\ & \ddots & \\ 0 & & a \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & \begin{bmatrix} b & & 0 \\ & \ddots & \\ 0 & & b \end{bmatrix} \end{pmatrix} Q^t \quad \begin{matrix} \} \, d \\ \} \, (p-d) \end{matrix}$$

avec $a > b$, $Q = [U, V]$ et où la matrice V, de taille $p \times (p-d)$, est telle que $Q^tQ = I_p$. Ainsi, le modèle PPCA isotropique est paramétrisé par μ, Q, a, b et d. Le modèle graphique associé à cette modélisation est présenté par la figure 2.4. On remarquera que le modèle $[a_k b_k Q_k d_k]$, présenté dans le paragraphe précédent, correspond à un mélange de modèles PPCA isotropiques. L'inférence d'un tel modèle par maximum de vraisemblance pour les paramètres μ, Q, a et b est donc similaire à celle du modèle $[a_k b_k Q_k d_k]$, présentée précédemment. Il est à noter que ce modèle, certes assez contraint, s'est avéré être en pratique très performant en classification supervisée (*cf.* [B2]).

2.2.2 Estimation de la dimension intrinsèque par maximum de vraisemblance

Nous considérons à présent l'estimation de la dimension intrinsèque d^* par maximum de vraisemblance. Dans ce cadre, nous avons obtenu le résultat suivant :

Theorème 1. *L'estimateur du maximum de vraisemblance de la dimension intrinsèque d^* est asymptotiquement unique et consistant dans le cas du modèle PPCA isotropique.*

Eléments de démonstration. Nous avons tout d'abord montré que la log-vraisemblance du modèle PPCA isotropique prend la forme suivante en $\hat{\theta} = (\hat{\mu}, \hat{a}, \hat{b}, \hat{Q})$:

$$-\frac{2}{n}\log(L(\hat{\theta}, d)) = d\log(\hat{a}) + (p-d)\log(\hat{b}) + p,$$

avec $\hat{a} = \frac{1}{d}\sum_{j=1}^{d}\lambda_j$, $\hat{b} = \frac{1}{(p-d)}\sum_{j=d+1}^{p}\lambda_j$ et où λ_j est la $j$ème plus grande valeur propre de la matrice de covariance empirique S des données. Ainsi, la maximisation de la vraisemblance par rapport à d est équivalent à la minimisation de la fonction $\phi_n(d) = d\log(\hat{a}) + (p-d)\log(\hat{b})$. Asymptotiquement, $S \overset{p.s.}{\to} \Sigma$ quand $n \to +\infty$ et, conséquence du lemme 2.1 de [99], $\lambda_j \overset{p.s.}{\to} a$ si $j \leq d^*$ et $\lambda_j \overset{p.s.}{\to} b$ si $j \geq d^*$. On considère alors deux cas. Si $d \leq d^*$, alors $\hat{a} \overset{p.s.}{\to} a$ et $\hat{b} \overset{p.s.}{\to} \frac{1}{p-d}\left[(d^*-d)a + (p-d^*)b\right]$ quand $n \to +\infty$. Ainsi, $\phi_n(d) \overset{p.s.}{\to} \phi(d)$ telle que :

$$\phi(d) = d\log(a) + (p-d)\log\left(\frac{(d^*-d)}{(p-d)}a + \frac{(p-d^*)}{(p-d)}b\right)$$

qui est une fonction strictement décroissante sur $[1, d^*]$. Par conséquent, le minimum de ϕ est unique et atteint pour $d = d^*$. Si $d \geq d^*$, alors $\hat{a} \overset{p.s.}{\to} \frac{1}{d}\left(d^*a + (d-d^*)b\right)$ et $\hat{b} \overset{p.s.}{\to} b$ quand $n \to +\infty$. Par conséquent, $\phi_n(d) \overset{p.s.}{\to} \phi(d)$ telle que :

$$\phi(d) = d\log\left(\frac{d^*}{d}a + \frac{d-d^*}{d}b\right) + (p-d)\log(b)$$

qui est une fonction strictement croissante sur $[d^*, p]$ et donc le minimum de ϕ est unique et atteint pour $d = d^*$. □

Ce résultat, qui peut paraître surprenant de prime abord, s'explique par la parfaite dualité qui existe entre le sous-espace de dimension d associé aux plus grandes valeurs propres de Σ et son sous-espace supplémentaire de dimension $(p-d)$. Du fait des rôles symétriques joués par les paramètres a et b dans le modèle PPCA isotropique, le nombre de paramètres à estimer est $\nu(d) = p + 2 + \min\{d(p-(d+1)/2), (p-d)(p-(p-d+1)/2)\}$. Ainsi, la complexité du modèle augmente jusqu'à $d = d^*$ et décroît au delà. Remarquons également que ce résultat implique que les critères de sélection de modèles du type $L(\theta) + pen(n)$, où pen est une pénalité telle que $pen(n)/n \to 0$ quand $n \to +\infty$, sont également asymptotiquement consistant pour estimer d^*. En particulier, les critères AIC [2] et BIC [87] sont de cette forme et profitent donc du résultat que nous avons établi.

2.2.3 Expérimentations numériques

Nous avons également étudié dans [B9] l'intérêt pratique de la méthode du maximum de vraisemblance pour l'estimation de la dimension intrinsèque d'un jeu de données de taille finie. Le résultat présenté dans le paragraphe précédent nous apporte des garanties sur le comportement de l'estimateur du maximum de vraisemblance (EMV) de d^* quand $n \to +\infty$ mais nous sommes souvent amenés à considérer des cas pratiques où n est petit et où $n/p \to 1$. De même, il est naturel de se demander comment l'EMV de d^* se comportera dans des situations où le rapport signal sur bruit est petit. Nous avons, pour ces raisons, étudié le comportement de l'EMV de d^* en fonction de deux paramètres $\alpha = n/p$ et $\beta = d^*a/[(p-d^*)b]$ qui caractérisent respectivement les conditions d'estimation et le rapport signal sur bruit dans le cadre du modèle PPCA isotropique.

Nous avons comparé sur simulations le comportement de la méthode du maximum de vraisemblance (ML sur les figures) aux principales techniques de l'état de l'art : BIC [87], AIC [2], le

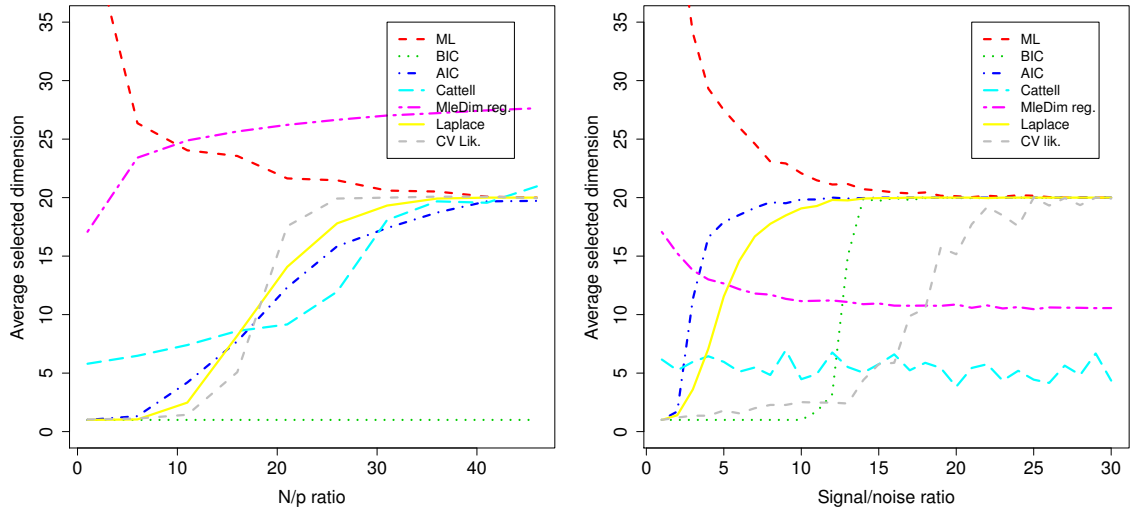


FIGURE 2.5: Influence du rapport $\alpha = n/p$ (à gauche, $\beta = 1$) et du rapport signal sur bruit β (à droite, $\alpha = 1$) sur l'estimation de d^* pour différentes méthodes.

scree-test de Cattell [21], MleDim [53], Laplace [70] et à la vraisemblance « cross-validée » (CvLik). Afin de ne pas se placer dans une situation qui aurait pu être trop favorable à notre méthode d'estimation de d^*, nous avons simulé des données en dimension $p = 50$ selon une loi uniforme avec d^* dimensions de variance $a/12$ et $(p - d)$ dimensions de variance $b/12$. La dimension intrinsèque d^* avait été fixé à $d^* = 20$.

La figure 2.5 présente l'estimation de d^* moyennée sur 50 réplications pour les différentes méthodes étudiées. Nous ne reportons ici que les résultats obtenus pour $\beta = 1$ avec α variant (rapport signal sur bruit petit, faible différence entre a et b) et $\alpha = 1$ avec β variant (rapport n/p petit, conditions d'estimation peu favorables). Il apparaît tout d'abord que la méthode MleDim échoue à déterminer la valeur correcte de d^* dans presque toutes les situations considérées ici. Pour le cas avec $\beta = 1$ (figure de gauche), on remarque tout d'abord que le critère BIC échoue pour toutes les valeurs de α dans l'estimation de d^*. Les méthodes AIC, Cattell, Laplace et CvLik ont quant à elles tendance à sous-estimer la vraie valeur de d^* alors que ML fournit une bonne estimation de d^* pour $\alpha \geq 10$ et surestime d^* pour $\alpha < 10$. Rappelons que dans le cadre de la réduction de dimension, il est préférable de surestimer d^* plutôt que de le sous-estimer. En effet, il vaut mieux conserver quelques variables non informatives en plus des variables pertinentes plutôt que de rejeter des variables informatives. Nous recommandons donc l'usage de l'estimateur du maximum de vraisemblance ML dans le cas $\beta = 1$ et pour toutes les valeurs de α. Pour le cas avec $\alpha = 1$ (figure de droite), il apparaît tout d'abord que BIC et CvLik se comportent significativement moins bien que AIC et Laplace. Ces derniers fournissent une bonne estimation de d^* pour $\beta \geq 5$ et sous-estiment d^* en deçà. L'estimateur du maximum de vraisemblance ML founit quant à lui une bonne estimation de d^* pour $\beta \geq 10$ et surestime d^* en deçà. Dans ce cas, il convient de recommander l'usage du critère AIC pour $\beta < 10$.

Ce travail sur l'estimation de la dimension intrinsèque d'un jeu de données a donc apporté, d'une part, des garanties théoriques sur le comportement de l'estimateur du maximum de vraisemblance de d^*. Ce résultat théorique peut de plus être étendu aux critères de vraisemblance pénalisée tels que AIC ou BIC. D'autre part, nous avons étudié le comportement de ces critères dans des situations pratiques, pour certaines très éloignées du cas asymptotique, et nous avons recommandé l'usage des estimateurs ML et AIC en fonction des situations (*cf.* figure 2.6).

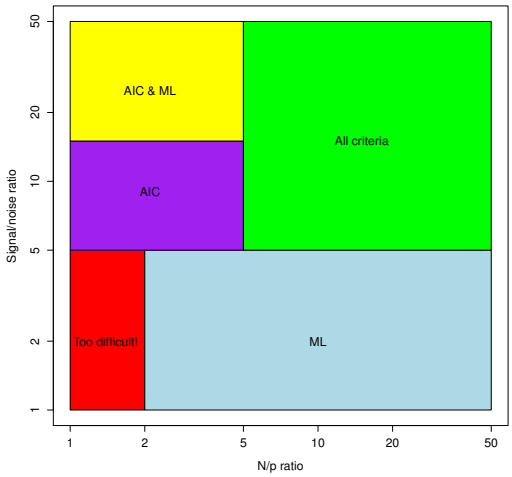


FIGURE 2.6: Critères recommandés pour l'estimation de la dimension intrinsèque d^* en fonction du rapport $\alpha = n/p$ et du rapport signal sur bruit β.

2.3 Clustering discriminatif : l'algorithme Fisher-EM

Comme nous l'avons vu précédemment, les méthodes de classification dans des sous-espaces sont des méthodes très performantes et qui ont été appliquées avec succès à un grand nombre de domaines applicatifs. Néanmoins, on peut regretter que ces techniques ne permettent pas une visualisation aisée de la classification produite. En particulier, en classification non supervisée dont le but est de réaliser une analyse exploratoire des données, la possibilité de visualiser la partition obtenue est un atout important dans la compréhension du phénomène étudié. Certes, il existe des modèles à sous-espace commun [5, 34] pour les méthodes de classification dans des sous-espaces mais le sous-espace commun est choisi dans ces travaux par rapport au critère de maximisation de la variance projetée. Malheureusement, on sait depuis les travaux pionniers de Fisher [33] que les axes maximisant la variance projetée ne sont pas forcément ceux qui permettent de discriminer au mieux les groupes. Les *discriminative latent models* (DLM) et leur algorithme d'inférence (Fisher-EM), que nous avons proposé dans [B12], visent à pallier ce défaut des méthodes existantes de classification dans des sous-espaces. Les propriétés de convergence de l'algorithme Fisher-EM ont également été considérées dans [B15].

2.3.1 Une famille de modèles latents discriminants

Considérons un ensemble de n observations $\{y_1, ..., y_n\} \in \mathbb{R}^p$ que l'on souhaite classer en K groupes homogènes, *i.e.* adjoindre une valeur $z_i = \{1, \ldots, K\}$ à l'observation y_i où $z_i = k$ indique que y_i appartient au groupe k. On suppose que $\{y_1, ..., y_n\}$ et $\{z_1, ..., z_n\}$ sont des réalisations indépendantes respectivement d'un vecteur aléatoire observé $Y \in \mathbb{R}^p$ et d'une variable aléatoire non observée $Z \in \{1, \ldots, K\}$. Par ailleurs, on définit $\mathbb{E} \subset \mathbb{R}^p$ un espace latent, supposé être l'espace le plus discriminant de dimension $d \leq K - 1$, tel que $0 \in \mathbb{E}$ et $K < p$. Dans cet espace latent, $\{x_1, ..., x_n\} \in \mathbb{E}$ représentent les « vraies » données qui sont supposées être des réalisations indépendantes d'un vecteur aléatoire $X \in \mathbb{E}$ non observé. On suppose en outre que les variables $Y \in \mathbb{R}^p$ et $X \in \mathbb{E}$ sont liées par la transformation linéaire :

$$Y = UX + \varepsilon,$$

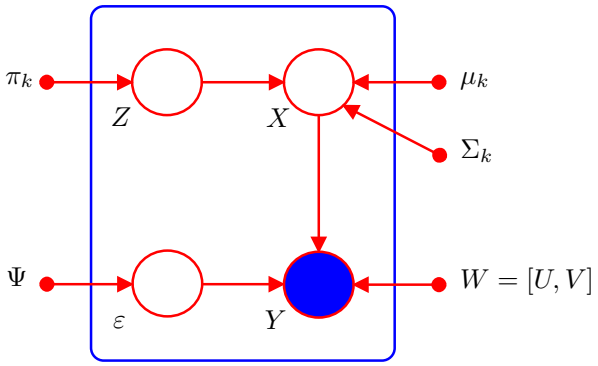


FIGURE 2.7: Représentation graphique du modèle $\mathrm{DLM}_{[\Sigma_k \beta]}$.

où U est une matrice de taille $p \times d$ vérifiant $U^t U = \mathbf{I}_d$ et ε est un terme de bruit. On suppose de plus que ε est un bruit gaussien centré en 0 et de matrice de covariance Ψ :

$$\varepsilon \sim \mathcal{N}(0, \Psi).$$

On suppose d'autre part que, conditionnellement à $Z = k$, X est également distribué selon une loi normale :

$$X|_{Z=k} \sim \mathcal{N}(\mu_k, \Sigma_k),$$

où $\mu_k \in \mathbb{E}$ et $\Sigma_k \in \mathbb{R}^{d \times d}$ représentent respectivement la moyenne et la matrice de covariance du groupe k dans l'espace latent $\mathbb{E}$. L'ensemble de ces hypothèses implique que la distribution conditionnelle de Y est $Y|_{X,Z=k} \sim \mathcal{N}(UX, \Psi)$ et que sa distribution marginale est un mélange de gaussiennes, *i.e.* $f(y) = \sum_{k=1}^{K} \pi_k \phi(y; m_k, S_k)$ où π_k, $m_k = U\mu_k$ et $S_k = U^t \Sigma_k U + \Psi$ représentent respectivement la proportion, la moyenne et la matrice de covariance de la classe k dans l'espace des observations et ϕ est la densité de probabilité de la loi normale multivariée. On introduit en outre la matrice $W = [U, V]$, de taille $p \times p$, telle que $W^t W = W W^t = \mathbf{I_p}$ où $V \in \mathbb{R}^{p \times (p-d)}$ est le complément orthogonal de U définie précédemment. Enfin, on suppose que la variance du bruit Ψ vérifie $V^t \Psi V = \beta \mathbf{I_p}$ et $U^t \Psi U = \mathbf{O_d}$ telle que $\Delta_k = W^t S_k W$ ait la forme suivante :

$$\Delta_k = \begin{pmatrix} \boxed{\Sigma_k} & \mathbf{0} \\ \mathbf{0} & \boxed{\begin{matrix} \beta & & 0 \\ & \ddots & \\ 0 & & \beta \end{matrix}} \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ b \end{matrix}}\right\} d \leq K-1 \\ \left.\vphantom{\begin{matrix} a \\ b \end{matrix}}\right\} (p-d) \end{matrix}$$

On fera, dans la suite, référence à ce modèle comme étant le modèle $\mathrm{DLM}_{[\Sigma_k \beta]}$ dont une représentation graphique est donnée par la figure 2.7. Par ailleurs, en relâchant la contrainte d'égalité entre les groupes portant sur β, il est possible d'obtenir le modèle $\mathrm{DLM}_{[\Sigma_k \beta_k]}$ qui est un modèle plus général. De même, en ajoutant des contraintes supplémentaires sur les paramètres Σ_k et β, il est possible de décliner 10 autres modèles DLM. Au total, nous avons créé ainsi une famille de 12 modèles DLM, plus ou moins contraints, permettant de modéliser différents types de données (*cf.* [B12] pour une description détaillée).

2.3.2 Inférence et classification : l'algorithme Fisher-EM

Dans le contexte de la classification non supervisée, la maximisation directe de la vraisemblance n'est pas faisable car les données ne sont pas complètes, *i.e.* Z n'est pas observée. Il est donc nécessaire d'utiliser un algorithme de type EM qui maximise la vraisemblance au travers de la maximisation de l'espérance de la log-vraisemblance complétée. En considérant l'ensemble de ces hypothèses, l'espérance de la log-vraisemblance complétée $Q\left(y_1,\ldots,y_n;U,\theta\right)$ du modèle $\text{DLM}_{[\Sigma_k\beta_k]}$ s'écrit :

$$\begin{aligned} Q\left(U,\theta\right) &= -\frac{1}{2}\sum_{k=1}^{K} n_k[-2\log\left(\pi_k\right)+\operatorname{trace}\left(\Sigma_k^{-1}U^tS_kU\right)+\log\left(|\Sigma_k|\right) \\ &\quad +(p-d)\log\left(\beta\right)+\frac{1}{\beta}(\operatorname{trace}\left(S_k\right)-\sum_{j=1}^{d}u_j^tS_ku_j)+\gamma]. \end{aligned} \tag{2.4}$$

où S_k est la matrice de covariance empirique de la classe k, u_j est le $j$ème vecteur colonne de la matrice U, $n_k=\sum_{i=1}^{n}t_{ik}$ où t_{ik} est la probabilité a posteriori qu'une observation y_i appartienne à la classe k et $\gamma=p\log\left(2\pi\right)$ est une constante. Du fait de la nature particulière de la matrice U qui définit le sous-espace discriminant $\mathbb{E}$, nous avons dû proposer dans [B12] une procédure itérative, baptisée Fisher-EM, qui est basée sur l'algorithme EM et qui insère entre les étapes E et M traditionnelles une étape F dédiée à l'estimation de U. Cette étape F vise à estimer la transformation linéaire $U\in\mathbb{R}^{p\times d}$, sous contrainte d'orthogonalité de ses colonnes, qui détermine le sous-espace latent de dimension $d=K-1$ dans lequel les K groupes sont le mieux séparés. Pour cela, nous avons adapté le critère traditionnel de Fisher $J(U)=\operatorname{trace}((U^tSU)^{-1}U^tS_BU)$, habituellement utilisé dans un contexte supervisé, à notre contexte (non supervisé) sous la contrainte d'orthogonalité et conditionnellement à la partition floue courante des données. La proposition suivante détaille les trois étapes de l'algorithme Fisher-EM.

Proposition 4. *Dans le cas du modèle $\text{DLM}_{[\Sigma_k\beta_k]}$, l'algorithme Fisher-EM prend la forme suivante à l'étape q :*

- *étape E : cette étape calcule, pour $k=1,...,K$ et $i=1,...,n$, la probabilité conditionnelle $t_{ik}^{(q)}=P(Z=k|Y=y_i)$ comme suit :*

$$t_{ik}^{(q)}=1\Big/\sum_{\ell=1}^{K}\exp\left(\frac{1}{2}(D_k^{(q-1)}(y_i)-D_\ell^{(q-1)}(y_i))\right),$$

avec

$$\begin{aligned} D_k^{(q-1)}(y_i) = \; & ||P(y_i-m_k^{(q-1)})||^2_{\mathcal{D}_k}+\frac{1}{\beta_k^{(q-1)}}||(y_i-m_k^{(q-1)})-P(y_i-m_k^{(q-1)})||^2 \\ & +\log\left(\left|\Sigma_k^{(q-1)}\right|\right)+(p-d)\log(\beta_k^{(q-1)})-2\log(\pi_k^{(q-1)})+\gamma, \end{aligned} \tag{2.5}$$

où $||y||^2_{\mathcal{D}_k}=y^t\mathcal{D}_ky$, $\mathcal{D}_k=\tilde{W}\Delta_k^{-1}\tilde{W}^t$, $\tilde{W}$ est la matrice composée des d colonnes de $U^{(q-1)}$ complétée par des zéros, $P(y)=U^{(q-1)}U^{(q-1)t}y$ et $\gamma=p\log(2\pi)$.

- *étape F : cette étape cherche, conditionnellement à la partition courante des données déterminée par les $t_{ik}^{(q)}$ estimés dans l'étape précédente, la solution du problème d'optimisation sous contrainte suivant :*

$$\begin{aligned} \hat{U}^{(q)} &= \max_{U} \quad \operatorname{trace}\left((U^tSU)^{-1}U^tS_B^{(q)}U\right), \\ &\quad \text{s.c.} \quad U^tU=\mathbf{I}_d, \end{aligned} \tag{2.6}$$

*où S représente la matrice de covariance et $S_B^{(q)} = \frac{1}{n}\sum_{k=1}^{K} n_k^{(q)}(m_k^{(q)} - \bar{y})(m_k^{(q)} - \bar{y})^t$ la matrice de covariance inter classe floue avec $n_k^{(q)} = \sum_{i=1}^{n} t_{ik}^{(q)}$, $m_k^{(q)} = 1/n_k^{(q)} \sum_{i=1}^{n} t_{ik}^{(q)} y_i$ et $\bar{y} = 1/n \sum_{i=1}^{n} y_i$. Ce problème d'optimisation peut être résolu en utilisant le concept de l'*orthonormal discriminant vector*, développé par [35], grâce à une procédure itérative de type Gram-Schmidt.*

– *étape M : cette étape maximise l'espérance de la vraisemblance complétée $Q(y_1, \ldots, y_n, U, \theta)$, définie par l'expression (2.4), conditionnellement aux $t_{ik}^{(q)}$ et à $\hat{U}^{(q)}$. Cette maximisation conduit aux formules de mise à jour des paramètres estimés suivantes :*

$$\hat{\pi}_k^{(q)} = \frac{n_k^{(q)}}{n}, \qquad \hat{\mu}_k^{(q)} = \frac{1}{n_k^{(q)}} \sum_{i=1}^{n} t_{ik}^{(q)} \hat{U}^{(q)t} y_i,$$

$$\hat{\Sigma}_k^{(q)} = \hat{U}^{(q)t} S_k \hat{U}^{(q)}, \qquad \hat{\beta}_k^{(q)} = \frac{\text{trace}(S_k) \text{-} \sum_{j=1}^{d} \hat{u}_j^{(q)t} S_k \hat{u}_j^{(q)}}{p - d},$$

où $n_k^{(q)} = \sum_{i=1}^{n} t_{ik}^{(q)}$.

La démonstration de ces résultats est donnée dans [B12]. L'algorithme Fisher-EM met à jour les paramètres et les probabilités conditionnelles $t_{ik}^{(q)}$ jusqu'à convergence de l'algorithme. Dans [B12], nous avons utilisé le critère de Aitken [55] qui estime à chaque itération le maximum asymptotique de la vraisemblance et permet de détecter aussi tôt que possible la convergence de l'algorithme.

2.3.3 Convergence de l'algorithme Fisher-EM

Bien que l'algorithme Fisher-EM repose sur un algorithme de type EM, il ne satisfait pas a priori à toutes les conditions relatives à la théorie de convergence de l'algorithme EM car l'étape F ne maximise pas directement l'espérance conditionnelle de la vraisemblance complétée. Sa convergence vers un maximum local de la vraisemblance n'est donc a priori pas garantie. Dans [B15], la convergence de l'algorithme Fisher-EM a été étudiée d'un point de vue théorique. Pour ce faire, nous avons distingué deux cas : le cas isotropique et le cas général.

Le cas isotropique : le modèle DLM$_{[\alpha\beta]}$

Nous considérons dans un premier temps le modèle DLM$_{[\alpha\beta]}$ qui suppose une matrice de covariance commune et sphérique pour chaque groupe, dans le sous-espace latent discriminant ($\forall k \in \{1, \ldots, K\}$, $\Sigma_k = \alpha \mathbf{I}_d$) et dans le sous-espace non discriminant ($\forall k \in \{1, \ldots, K\}$, $\beta_k = \beta$). Dans ce cas, nous avons le résultat suivant :

Theorème 2. *Dans le cas du modèle DLM$_{[\alpha\beta]}$, l'algorithme Fisher-EM est un algorithme EM et sa convergence vers un maximum local de la vraisemblance est donc garantie.*

Eléments de démonstration. Dans le but de montrer que l'algorithme Fisher-EM est un algorithme EM classique dans le cas du modèle DLM$_{[\alpha\beta]}$, il est nécessaire et suffisant de montrer que, à l'itération q, la maximisation du critère de Fisher (2.6) est équivalent à la maximisation de l'espérance conditionnelle de la log-vraisemblance complétée $Q(U, \theta)$. Nous supposons tout d'abord, et cela sans perte de généralité, que $cov(Y) = \mathbf{I_p}$. Ainsi, du fait de l'égalité $cov(Y) = S_W + S_B$, le problème (2.6) peut être réécrit de la manière suivante :

$$\min_U \text{trace}\left(U^t S_W U\right), \text{ s.c. } U^t U = \mathbf{I}_d,$$

Considérons d'autre part la quantité $-2Q(U,\theta)$:

$$-2Q(U,\theta) \quad = \quad \sum_{k=1}^{K}\left[\sum_{i=1}^{n} t_{ik}[\log|S_k| + (y_i - m_k)^t S_k^{-1}(y_i - m_k)]\right] + \gamma_1,$$

où γ_1 est un terme indépendant de U. Les hypothèses faites sur le modèle $\text{DLM}_{[\alpha\beta]}$ impliquent que $-2Q(U,\theta)$ peut être réécrit comme suit :

$$-2Q(U,\theta) = n\text{trace}\left(\Delta^{-1}W^t S_W W\right) + \gamma_2.$$

Finalement, en considérant les matrices $\tilde{W} = [U, 0_{p-d}]$ et $\bar{W} = [0_d, V]$ telles que $W = \tilde{W} + \bar{W}$, avec V le complément orthogonal de U, la relation $W^t S_W W = \tilde{W}^t S_W \tilde{W}^t + \bar{W}^t S_W \bar{W}$ peut être établie puisque $\tilde{W}^t S_W \bar{W}$ et $\bar{W}^t S_W \tilde{W}$ sont toutes deux des matrices nulles. Ainsi,$-2Q(U,\theta)$ peut être finalement réécrit de la façon suivante :

$$-2Q(U,\theta) = \frac{n}{\alpha}\text{trace}\left(U^t S_W U\right) + \gamma_3,$$

avec γ_3 indépendant de U. En conséquence, minimiser la quantité $\text{trace}(U^t S_W U)$ par rapport à U est équivalent à maximiser $Q(U,\theta)$ dans le cas du modèle $\text{DLM}_{[\alpha\beta]}$. □

Cas général : les autres modèles DLM

Nous considérons à présent le cas général qui concerne les 11 autres modèles DLM. Dans ce cas, nous avons le résultat suivant :

Theorème 3. *Si, à chaque itération q, la quantité :*

$$\delta^{(q)} = \sum_{k=1}^{K}\text{trace}\left[n_k^{(q)}\left(\hat{\Sigma}_k^{(q-1)^{-1}} - \frac{1}{\hat{\beta}_k^{(q-1)}}\mathbf{I}_d\right)\left(\hat{U}^{(q-1)t} S_k^{(q)} \hat{U}^{(q-1)} - \hat{U}^{(q)t} S_k^{(q)} \hat{U}^{(q)}\right)\right]$$

est positive, alors l'algorithme Fisher-EM est un algorithme EM généralisé (GEM) et sa convergence vers un maximum local de la vraisemblance est garantie.

Eléments de démonstration. Dans le but de montrer que l'algorithme Fisher-EM est un EM généralisé [29], il est nécessaire de montrer qu'à chaque itération q : $Q(\hat{U}^{(q+1)}, \hat{\theta}^{(q+1)}) \geq Q(\hat{U}^{(q)}, \hat{\theta}^{(q)})$. Soit $\hat{U}^{(q)}$ et $\hat{\theta}^{(q)} = \{\hat{\mu}^{(q)}, \hat{\mathbf{\Sigma}}^{(q)}, \hat{\beta}^{(q)}, \hat{\pi}^{(q)}\}$ les paramètres du modèle estimés à l'itération q et $t_{ik}^{(q+1)}$ pour $i = 1, ..., n$, $k = 1, ..., K$, les probabilités a posteriori calculées dans l'étape E à l'itération $q+1$.

D'une part, considérons la quantité : $\delta^{(q+1)} = Q(\hat{U}^{(q+1)}, \hat{\theta}^{(q)}) - Q(\hat{U}^{(q)}, \hat{\theta}^{(q)})$ qui, avec les hypothèses du modèle $\text{DLM}_{[\Sigma_k\beta_k]}$, peut être réécrite de la manière suivante :

$$\delta^{(q+1)} = \frac{1}{2}\left[\sum_{k=1}^{K}\text{trace}\left(B_k^{(q)}\left(A_k^{(q)} - A_k^{(q+1)}\right)\right)\right],$$

où $A_k^{(\ell)} = \hat{U}^{(\ell)t} n_k^{(q+1)} C_k^{(q+1)} \hat{U}^{(\ell)}$ et $B_k^{(q)} = \hat{\Sigma}_k^{(q)^{-1}} - \frac{1}{\hat{\beta}_k^{(q)}}\mathbf{I}_d$. Toutefois, même si le critère maximisé dans l'étape F garantit $\sum_{k=1}^{K}\text{trace}(A_k^{(q)} - A_k^{(q+1)}) \geq 0$ si $S = I_p$, cela n'implique pas que la condition $\text{trace}(A_k^{(q)} - A_k^{(q+1)}) \geq 0$ soit satisfaite pour tout $k = 1, ..., K$. Aussi, à chaque itération, la condition $\delta^{(q+1)} \geq 0$ n'est pas nécessairement garantie même si $B_k^{(q)}$ est une matrice semi-définie positive. Nous supposerons donc par la suite que $\text{H1} : \delta^{(q+1)} \geq 0$ est vérifiée.

Méthode	iris (p=4)	glass (p=7)	wine (p=13)	zoo (p=16)	chiro (p=17)	satimage (p=36)	USPS358 (p=256)
Fisher-EM	97.8±0.1	51.1±2.1	98.9±0.0	80.2±5.3	98.2±3.4	70.1±0.0	82.3±4.7
EM (Full-GMM)	79.0±5.7	38.3±2.1	60.9±7.7	-	44.8±4.1	35.9±3.1	-
EM (Com-GMM	57.6±18.3	38.3±3.1	61.0±14.9	59.9±10.3	51.9±10.9	26.1±1.5	38.2±1.1
EM (Diag-GMM)	93.5±1.3	39.1±2.4	94.6±2.8	70.9±12.3	92.1±4.2	60.8±5.2	45.9±9.1
EM (Sphe-GMM)	89.4±0.4	37.0±2.1	96.6±0.0	69.4±5.4	85.9±9.9	60.2±7.5	78.7±11.2
PCA+EM	66.9±9.9	39.0±1.7	64.4±5.7	61.9±6.2	66.1±4.0	56.2±4.2	67.6±11.2
k-means	88.7±4.0	41.3±2.8	95.9±4.0	68.0±7.4	92.9±6.0	66.6±4.1	74.9±13.9
Mixt-PPCA	89.1±4.2	37.0±2.3	63.1±7.9	50.9±6.5	56.3±4.5	40.6±4.7	53.1±9.6
MCFA ($q = 3$)	80.6±12.6	47.7±6.9	92.9±8.2	-	75.4±7.8	67.9±8.8	54.2±8.7
Mclust	96.7	41.6	97.1	65.3	97.9	58.7	55.5

TABLE 2.2: Comparaison de la performance de clustering entre Fisher-EM avec les différents modèles DLM et les meilleures méthodes de l'état de l'art sur 7 jeux de données UCI.

D'autre part, la théorie générale de l'algorithme EM nous assure que l'ensemble des paramètres $\hat{\theta}^{(q+1)} = \{\hat{\mu}^{(q+1)}, \hat{\mathbf{\Sigma}}^{(q+1)}, \hat{\beta}^{(q+1)}, \hat{\pi}^{(q+1)}\}$ estimés dans l'étape M à l'itération $q+1$ vérifie, pour tout θ, $Q(\hat{U}^{(q+1)}, \hat{\theta}^{(q+1)}) \geq Q(\hat{U}^{(q+1)}, \theta)$. Ainsi, puisque $Q(\hat{U}^{(q+1)}, \hat{\theta}^{(q+1)}) \geq Q(\hat{U}^{(q+1)}, \hat{\theta}^{(q)})$ et $Q(\hat{U}^{(q+1)}, \hat{\theta}^{(q)}) \geq Q(\hat{U}^{(q)}, \hat{\theta}^{(q)})$, alors l'algorithme Fisher-EM est un algorithme GEM si $\mathrm{H1}$ est vérifiée. □

L'hypothèse $\mathrm{H1}$ de convergence n'apparaît cependant pas être une condition forte dès lors que $B_k^{(q)}$ est une matrice semi-définie positive. En effet, le critère maximisé dans l'étape F implique que $\sum_{k=1}^{K} \mathrm{trace}(A_k^{(q)} - A_k^{(q+1)}) \geq 0$ et que $\sum_{k=1}^{K} \mathrm{trace}(B_k^{(q)}) \geq 0$ à chaque itération q. Il est donc très probable que l'hypothèse $\mathrm{H1}$ soit fréquemment satisfaite. En outre, la quantité $\mathrm{H1}$ étant calculable facilement à chaque itération de l'algorithme, il est aisé de vérifier en pratique la validité de l'hypothèse $\mathrm{H1}$ lors de l'application de l'algorithme à un jeu de données.

2.3.4 Expérimentations numériques

Nous présentons à présent les résultats de quelques expérimentations numériques de [B12] illustrant les propriétés pratiques de l'algorithme Fisher-EM. Nous présentons, tout d'abord, à titre d'illustration le résultat de l'application de Fisher-EM sur les données Iris de Fisher qui sont bien connues dans le contexte de la classification. La figure 2.8 présente la projection des données sur le sous-espace discriminant estimé par l'algorithme Fisher-EM ainsi que la partition des données en 3 groupes à différentes étapes de l'algorithme. Pour cet exemple, le modèle $\mathrm{DLM}_{[\Sigma_k \beta_k]}$ a été utilisé, le nombre K de groupes était fixé à 3 et l'initialisation a été obtenue par tirage aléatoire selon une loi multinomiale. L'algorithme Fisher-EM converge ici en 10 itérations et la partition des données coïncide à 96% avec la classification connue pour ces données. La visualisation fournie par Fisher-EM s'avère être très informative et est très proche de la visualisation obtenue dans le cas supervisé (avec connaissance des labels) par l'analyse discriminante de Fisher [33]. Nous rappelons que la visualisation fournie par Fisher-EM a été obtenue sans avoir connaissance des appartenances aux classes (labels).

Nous avons ensuite évalué et comparé dans [B12] la performance de clustering de Fisher-EM aux meilleures méthodes de l'état de l'art. L'évaluation de la performance d'une méthode de clustering est un problème complexe qui n'a, à ce jour, pas de solution claire (*cf.* [41] pour une discussion sur le sujet). Nous avons fait le choix d'évaluer la performance des méthodes étudiées en comparant l'adéquation entre la partition proposée par la méthode et la classification supervisée disponible pour les données. Les méthodes ont été comparées sur 7 jeux de données, provenant du site web UCI, avec des dimensions allant de 4 à 256. Notons que le jeu de données USPS358 est un sous-ensemble du jeu de données USPS dans lequel ont été conservé uniquement les observations

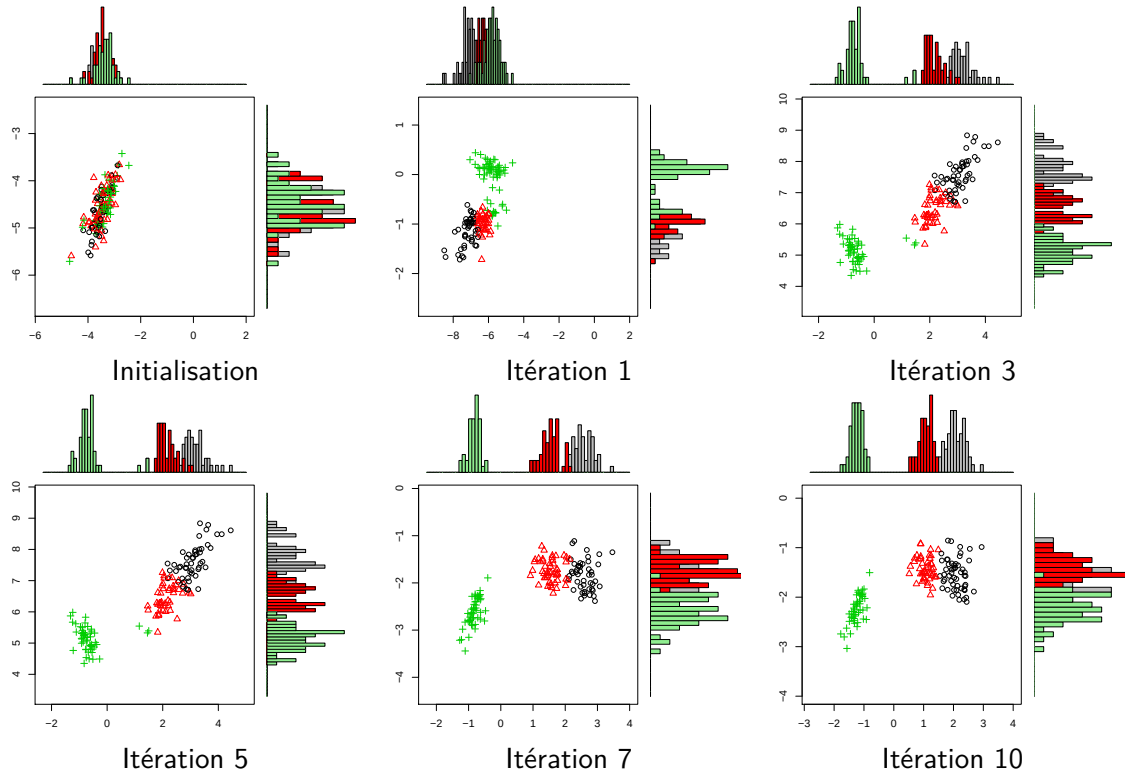


FIGURE 2.8: Représentation latente estimée et partition des données à différentes étapes de l'algorithme Fisher-EM sur les données Iris de Fisher.

des trois classes les plus difficiles à discriminer : les classes des chiffres 3, 5 et 8. La table 2.2 présente les résultats. Il apparaît clairement que Fisher-EM, outre ses qualités d'interprétation, est un algorithme de clustering très performant. En particulier, Fisher-EM fournit des résultats significativement meilleurs que ses concurrents en grande dimension (USPS358). Dans [B12], nous avons également vérifié sur simulations que Fisher-EM était robuste à l'augmentation de la dimension des données et que le critère BIC pouvait être utilisé efficacement pour la sélection du nombre de groupes et du modèle DLM adéquat pour les données considérées.

Dans [B15], nous avons en outre mené quelques expérimentations numériques afin d'étudier d'un point de vue pratique la convergence de l'algorithme Fisher-EM. Nous avons voulu en particulier voir si il était intéressant de baser le critère d'arrêt de l'algorithme sur le critère de Fisher plutôt que sur la vraisemblance, comme cela est fait traditionnellement. Pour ce faire, nous avons simulé des données dans un espace de 25 dimensions selon le modèle $\mathrm{DLM}_{[\alpha_k\beta]}$ avec $K = 3$. L'algorithme Fisher-EM a ensuite été lancé sur ces données avec, d'une part, une condition d'arrêt basé sur la vraisemblance et, d'autre part, une condition d'arrêt basé sur le critère de Fisher. Les deux critères d'arrêt étaient bien entendu basés sur des quantités standardisées pour être comparables et le seuil d'arrêt était fixé à 1×10^{-3}. La figure 2.9 présente les boxplots du nombre d'itérations de Fisher-EM et de la performance de clustering, tous deux évalués sur 25 réplications. Il apparaît nettement que le critère d'arrêt basé sur la vraisemblance stoppe l'algorithme plut tôt que celui basé sur le critère de Fisher. Cependant, les performances de clustering sont significativement meilleures avec ce dernier. On peut donc recommander l'usage du critère d'arrêt basé sur le critère de Fisher dans l'algorithme Fisher-EM quand on s'intéresse plus à la performance de clustering qu'à l'objectif de modélisation.

Enfin, nous avons comparé la vitesse de convergence algorithmique de Fisher-EM à celle de EM mais aussi de CEM [23] qui est réputé converger plus rapidement que EM. Pour ce faire, nous

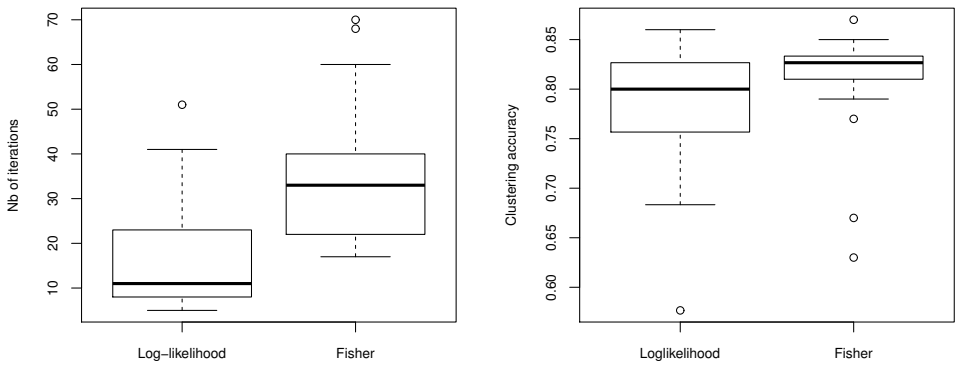


FIGURE 2.9: Influence des critères d'arrêt basés sur la vraisemblance et sur le critère de Fisher sur le nombre d'itérations et la performance de clustering (données simulées).

avons à nouveau simulé des données selon le modèle $\mathrm{DLM}_{[\alpha_k\beta]}$ mais cette fois-ci dans un espace de dimension 5 et avec un grand nombre d'observation de sorte que EM et CEM ne soient pas gênés par la dimension des données et puissent estimer correctement le modèle. La figure 2.10 présente les boxplots du nombre d'itérations, de la performance de clustering et de l'erreur d'estimation pour les algorithmes EM, CEM et Fisher-EM. Il apparaît que Fisher-EM converge plus vite qu'à la fois EM et CEM tout en fournissant des performances de clustering et d'estimation similaires, voire supérieures, à EM. Cela s'explique certainement par le fait que l'espace des paramètres dans lequel Fisher-EM recherche les estimateurs est très contraint du fait des hypothèses fortes sur le modèle probabiliste sous-jacent. La figure 2.11 montre le parcours emprunté dans l'espace des paramètres par les algorithmes EM, CEM et Fisher-EM pour l'estimation de la moyenne du 1er groupe. On voit clairement que Fisher-EM parcourt beaucoup plus efficacement l'espace des paramètres pour trouver l'estimateur du paramètre considéré.

En conclusion, l'algorithme Fisher-EM s'est avéré être une méthode de clustering très performante et qui facilite la visualisation, et de ce fait la compréhension, de la partition proposée des données. L'étude de la convergence de Fisher-EM a montré que Fisher-EM était plus rapide à converger que EM et CEM sans perte de performance en clustering et en estimation. Nous recommandons de plus l'usage du critère de Fisher comme critère d'arrêt de l'algorithme si l'utilisateur est principalement intéressé par la performance de classification.

2.4 Sélection de variables en clustering par pénalisation ℓ_1

Ces dernières années ont vu l'apparition d'un grand nombre de méthodes de sélection de variables, particulièrement en régression et en analyse discriminante, mais également en clustering. La sélection de variables pertinentes pour la tâche considérée (le clustering dans notre cas) présente l'avantage de pallier le problème de la grande dimension des données et facilite nettement l'interprétation du phénomène étudié. En effet, dans de nombreuses applications, les variables ont des interprétations physiques, économiques ou biologiques et la détermination des variables utiles à la discrimination des données peut s'avérer très informative.

Dans le contexte de la classification non supervisée, deux types d'approches ont été proposées ces dernières années. D'une part, certains auteurs tels que [59, 60, 79] considèrent la sélection de variables en clustering comme un problème de sélection de modèles dans un cadre bayésien. Dean

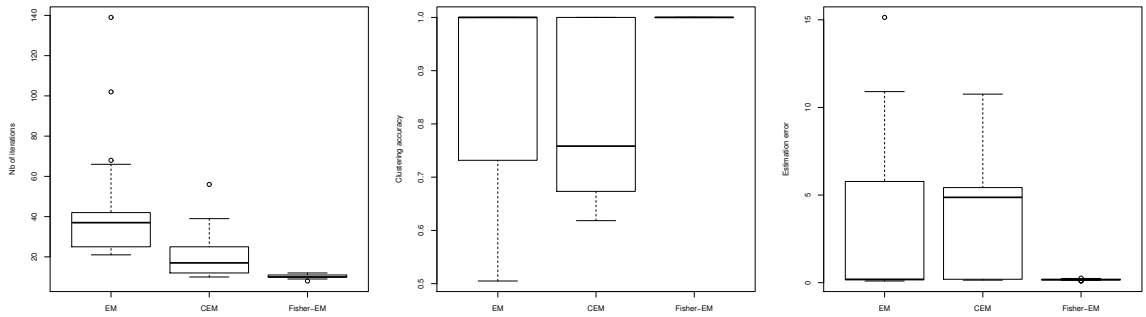


FIGURE 2.10: Nombre d'itérations, performance de clustering et erreur d'estimation pour les algorithmes EM, CEM et Fisher-EM (données simulées).

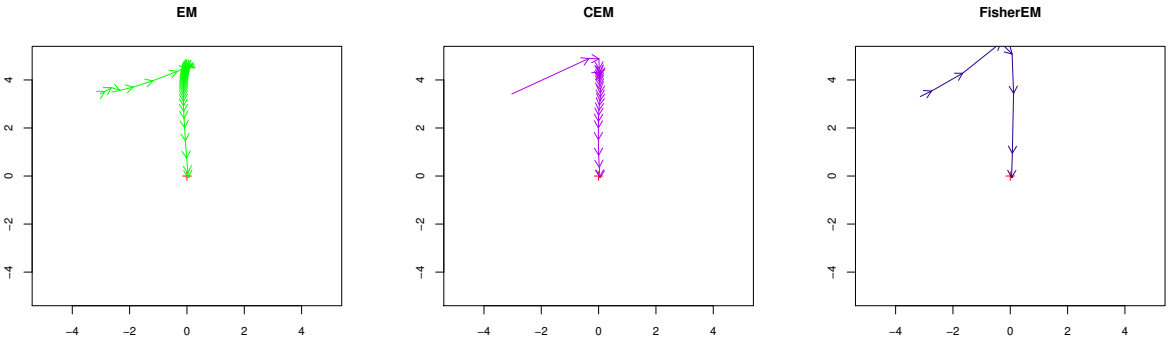


FIGURE 2.11: Parcours dans l'espace des paramètres des algorithmes EM, CEM et Fisher-EM pour l'estimation de la moyenne du 1er groupe (données simulées).

et Raftery [79] ont, en particulier, proposé une modélisation qui considère deux sous-ensembles de variables : les variables utiles pour le clustering et les variables non informatives pour cette tâche. Ce modèle a été étendu récemment par Maugis *et al.* [59, 60] qui ajoutent un sous-ensemble de variables utiles mais corrélées au premier sous-ensemble. Cette approche a été utilisée avec succès dans de nombreux contextes. On peut toutefois regretter le coût calculatoire élevé lié à la procédure d'exploration des combinaisons de variables. D'autre part, plusieurs travaux tels que [75, 100, 103] utilisent une pénalisation de type ℓ_1 pour effectuer la sélection de variables pertinentes pour le clustering. Dans le contexte du modèle de mélange, Pan & Shen [75] ont par exemple ajouté une pénalité ℓ_1 à la fonction de vraisemblance pour introduire de la parcimonie dans les matrices de covariance. De même, Witten & Tibshirani [100] ont proposé un critère pénalisé de classification permettant, en particulier, de faire de la sélection de variables avec les algorithmes k-means et CAH.

De notre côté, nous avons proposé dans un travail récent [B20] d'introduire de la parcimonie dans l'algorithme Fisher-EM afin de sélectionner les variables discriminantes pour le problème considéré. Pour cela, nous avons choisit d'introduire la parcimonie dans l'étape F grâce à une pénalisation ℓ_1 du critère de Fisher (2.6).

2.4.1 Trois versions sparses de l'algorithme Fisher-EM

Nous avons identifié dans [B20] trois manières différentes d'introduire de la parcimonie dans la matrice des loadings U, estimée dans l'étape F de l'algorithme Fisher-EM, ce qui a donné

naissance à trois versions « sparses » de Fisher-EM.

Approche par approximation pénalisée

Cette première approche procède en deux étapes : nous utilisons dans un premier temps l'étape F de l'algorithme Fisher-EM pour fournir une estimation $\hat{U}^{(q)}$ de la matrice d'orientation du sous-espace discriminant à l'étape q et, dans un second temps, nous cherchons par régression pénalisée la meilleure approximation parcimonieuse $\tilde{U}^{(q)}$ de $\hat{U}^{(q)}$. Dans ce cadre, nous avons obtenu le résultat suivant :

Proposition 5. *La meilleure approximation parcimonieuse et orthonormale au niveau λ de $\hat{U}^{(q)}$ est $\tilde{U}^{(q)} = u^{(q)}v^{(q)t}$ où $u^{(q)}$ et $v^{(q)}$ sont respectivement les d premiers vecteurs singuliers à gauche et à droite de la décomposition en valeurs singulières (SVD) de la solution du problème de régression pénalisée :*

$$\min_U \left\| X^{(q)t} - Y^t U \right\|_F^2 + \lambda \sum_{j=1}^{d} \|u_j\|_1 ,$$

où u_j est le $j$ème vecteur colonne de U et $X^{(q)} = \hat{U}^{(q)t}Y$.

La démonstration de ce résultat est donnée dans [B20]. D'un point de vue pratique, le problème de régression pénalisée peut tout d'abord être résolu en appliquant l'algorithme LARS [31] itérativement pour chacun des d axes discriminants à approcher, puis les d axes discriminants parcimonieux sont orthonormalisés au moyen d'une SVD. Il est également possible d'ajouter une régularisation de type ridge au problème de régression pénalisée, afin par exemple de traiter des problèmes où n est petit devant p, auquel cas l'algorithme ElasticNet [104] permettra d'en trouver la solution. Cependant, même si nous verrons que cette approche fournit des résultats satisfaisants, on peut regretter que l'étape d'estimation de $U^{(q)}$ et de parcimonie (calcul de $\tilde{U}^{(q)}$) soient faites indépendamment. Il serait plus naturel de chercher directement des axes qui soient à la fois discriminants et parcimonieux. Les deux approches suivantes proposent de telles solutions.

Approche par régression pénalisée

Cette seconde approche propose donc d'estimer directement des axes discriminants et parcimonieux et, pour ce faire, reformule le critère de Fisher (2.6) sous la forme d'un problème de régression qui sera ensuite pénalisé. Il est tout d'abord nécessaire de définir les matrices $H_W^{(q)}$ et $H_B^{(q)}$ conditionnellement aux probabilités a posteriori $t_{ik}^{(q)}$ calculée à l'étape E :

Définition 1. *Les matrices $H_W^{(q)} \in \mathbb{R}^{p\times n}$ et $H_B^{(q)} \in \mathbb{R}^{p\times K}$ sont définies, conditionnellement aux probabilités a posteriori $t_{ik}^{(q)}$ calculées à l'étape E de l'itération q, comme suit :*

$$H_W^{(q)} = \frac{1}{\sqrt{n}} \left[Y - \sum_{k=1}^{K} t_{1k}^{(q)} m_k^{(q)}, \ldots, Y - \sum_{k=1}^{K} t_{nk}^{(q)} m_k^{(q)} \right] \in \mathbb{R}^{p\times n} \tag{2.7}$$

$$H_B^{(q)} = \frac{1}{\sqrt{n}} \left[\sqrt{n_1^{(q)}}(m_1^{(q)} - \bar{y}), \ldots, \sqrt{n_K^{(q)}}(m_K^{(q)} - \bar{y}) \right] \in \mathbb{R}^{p\times K}, \tag{2.8}$$

où $n_k^{(q)} = \sum_{i=1}^{n} t_{ik}^{(q)}$ et $m_k^{(q)} = \frac{1}{n}\sum_{i=1}^{n} t_{ik}^{(q)} y_i$.

Ainsi, les matrices $H_W^{(q)}$ et $H_B^{(q)}$ sont telles que $H_W^{(q)}H_W^{(q)t} = S_W^{(q)}$ et $H_B^{(q)}H_B^{(q)t} = S_B^{(q)}$. Avec ces notations, nous avons obtenu le résultat suivant :

Proposition 6. *La meilleure approximation parcimonieuse et orthonormale au niveau λ de la solution de* (2.6) *est $\tilde{U}^{(q)} = u^{(q)}v^{(q)t}$ où $u^{(q)}$ et $v^{(q)}$ sont respectivement les d premiers vecteurs singuliers à gauche et à droite de la décomposition en valeurs singulières (SVD) de la solution $\hat{B}$ du problème de régression pénalisée suivant :*

$$\min_{A,B} \sum_{k=1}^{K} \left\| R_W^{(q)-t} H_{B,k}^{(q)} - AB^t H_{B,k}^{(q)} \right\|_F^2 + \rho \sum_{j=1}^{d} \beta_j^t S_W^{(q)} \beta_j + \lambda \sum_{j=1}^{d} \|\beta_j\|_1 \text{ tel que } A^tA = \mathbf{I}_d,$$

où $R_W^{(q)} \in \mathbb{R}^{p\times p}$ est une matrice triangulaire supérieure issue de la décomposition de Cholesky de $S_W^{(q)}$, *i.e.* $S_W^{(q)} = R_W^{(q)t} R_W^{(q)}$, $A = [\alpha_1, ..., \alpha_d]$, $B = [\beta_1, ..., \beta_d]$, $H_{B,k}^{(q)}$ *est la $k$ème colonne de* $H_B^{(q)}$ *et $\rho > 0$ est un paramètre de régularisation de type ridge.*

La démonstration de ce résultat est donnée dans [B20]. D'un point de vue pratique, ce problème de régression pénalisée peut être résolu grâce à l'algorithme proposé par Qiao *et al.* [78] dans le cadre supervisé en optimisant alternativement sur B avec A fixé puis sur A avec B fixé.

Approche par SVD pénalisée

La troisième et dernière approche vise également à estimer directement des axes discriminants et parcimonieux. Pour ce faire, nous nous ramenons à un problème d'optimisation dont la solution peut être obtenue par SVD. Ainsi, la version pénalisée de ce problème d'optimisation pourra être obtenue grâce à l'algorithme proposé par Witten & Tibshirani [101]. A cette fin, nous avons obtenu le résultat suivant :

Proposition 7. *La meilleure approximation parcimonieuse et orthonormale au niveau λ de la solution de* (2.6) *est $\tilde{U}^{(q)} = u^{(q)}v^{(q)t}$ où $u^{(q)}$ et $v^{(q)}$ sont respectivement les d premiers vecteurs singuliers à gauche et à droite de la décomposition en valeurs singulières (SVD) de la solution du problème de régression pénalisée suivant :*

$$\min_{U} \sum_{\ell=1}^{p} \left\| S_{B,\ell}^{(q)} - UU^t S_{B,\ell}^{(q)} \right\|^2 + \lambda \sum_{j=1}^{d} \|u_j\|_1 \text{ tel que } U^tU = \mathbf{I}_d,$$

où u_j *est le $j$ème vecteur colonne de U et* $S_{B,\ell}^{(q)}$ *est la ℓème colonne de* $S_B^{(q)}$.

La démonstration de ce résultat est donnée dans [B20]. D'un point de vue pratique, le problème de régression pénalisée considéré ci-dessus peut être résolu grâce à l'algorithme proposé par Witten & Tibshirani [101] qui réalise une « SVD sparse » de la matrice $S_B^{(q)}$ dans le cas présent.

2.4.2 Choix du paramètre de pénalisation et implantation de l'algorithme

Les trois approches présentées précédemment partagent tout d'abord le problème du choix du paramètre λ qui contrôle la parcimonie de la matrice des loadings U. Le choix de ce paramètre est un problème qui a été très étudié dans le contexte supervisé mais qui a malheureusement reçu très peu d'attention dans le contexte non supervisé. Une idée naturelle dans le contexte du clustering à base de modèle de mélange est d'utiliser un critère de vraisemblance pénalisée, tel que le critère BIC, pour choisir λ. Toutefois, le critère BIC requiert la connaissance du nombre de paramètres du modèle considéré et cela pose le problème du calcul de ce nombre en fonction de λ. C'est un problème a priori complexe car l'effet du paramètre λ sur la complexité du modèle associé est difficile à appréhender. Faisant suite aux conjectures de Efron *et al.* [31], Zou *et al.* [105] ont montré que le nombre de coefficients non nuls est un estimateur non biaisé et asymptotiquement

Méthode	iris (p=4)	glass (p=9)	wine (p=13)	zoo (p=16)	chiro (p=17)	satimage (p=36)	usps358 (p=256)
sparseFEM-1	96.5±0.3	50.2±1.9	97.8±0.2	71.4±8.5	84.2±11	69.6±0.1	84.7±3.2
	(2.0±0.0)	*(6.0±1.0)*	*(2.0±0.0)*	*(13±2.5)*	*(2.3±0.5)*	*(36±0.0)*	*(5.5±0.7)*
sparseFEM-2	89.9±0.4	48.4±3.0	98.3±0.0	70.1±12.2	84.8±12	67.5±1.6	82.8±9.1
	(4.0±0.0)	*(6.6±0.7)*	*(4.0±0.0)*	*(14±3.6)*	*(2.0±0.6)*	*(36±.0.0)*	*(15.5±16)*
sparseFEM-3	96.5±0.3	48.2±2.7	97.8±0.0	72.0±4.3	82.9±12	71.8±2.3	79.1±7.4
	(2.0±0.3)	*(7.0±0.0)*	*(2.0±0.0)*	*(10±2.8)*	*(2.0±0.0)*	*(36±0.0)*	*(6.0±1.3)*
sparseKmeans	90.7	52.3	94.9	79.2	95.3	71.4	74.7
	(4)	*(6)*	*(13)*	*(16)*	*(17)*	*(36)*	*(213)*
Clustvarsel	96.0	48.6	92.7	75.2	71.1	58.7	48.3
	(3)	*(3)*	*(5)*	*(3)*	*(6)*	*(19)*	*(6)*
Selvarclust	96.0	43.0	94.4	92.1	92.6	56.4	36.7
	(3)	*(6)*	*(5)*	*(5)*	*(8)*	*(22)*	*(5)*

TABLE 2.3: Performances de clustering et nombres de variables sélectionnées (entre parenthèses) pour 7 jeux de données UCI (modèle et λ choisis par BIC pour sparseFEM).

consistant du nombre de degrés de liberté du modèle. Ainsi, la complexité du modèle $\mathrm{DLM}_{[\Sigma_k \beta_k]}$ dans le cas sparse est donnée en fonction de λ par :

$$\gamma(\lambda) = (K-1) + Kd + \nu_U(\lambda) + Kd(d+1)/2 + K,$$

où $\nu_U(\lambda)$ est le nombre de coefficients non nuls dans la matrice des loadings U pour la valeur λ. Nous utilisons, dans les expériences présentées au paragraphe suivant, le critère BIC avec cette complexité pour choisir à la fois λ, K et le modèle DLM le plus adapté aux données considérées.

Concernant la mise en œuvre des trois versions sparses proposées pour l'algorithme Fisher-EM, plusieurs possibilités d'implantation sont envisageables. Il est tout d'abord possible d'utiliser directement une version sparse de Fisher-EM dès la première itération avec une initialisation quelconque. Nous craignions toutefois qu'une telle approche soit trop contraignante et que l'algorithme ne soit pas capable de trouver une solution satisfaisante. Nous avons donc proposé dans [B20] d'utiliser comme initialisation de la version sparse, les résultats (partition et estimateurs des paramètres) de l'algorithme Fisher-EM. Ainsi, nous recommandons d'utiliser Fisher-EM jusqu'à convergence puis d'appliquer une version sparse de Fisher-EM jusqu'à convergence également.

2.4.3 Expérimentations numériques

Afin d'illustrer les propriétés et les intérêts pratiques des versions sparses de Fisher-EM proposées, nous avons tout d'abord considéré dans [B20] les jeux de données réelles utilisés au paragraphe précédent et comparé sparseFEM aux meilleures méthodes de l'état de l'art. La table 2.3 présente les performances de clustering et les nombres de variables sélectionnées pour 7 jeux de données UCI par Fisher-EM, sparseFEM-1 (approche par approximation pénalisée), sparseFEM-2 (approche par régression pénalisée), sparseFEM-3 (approche par SVD pénalisée), sparseKmeans [100], Clustvarsel [79] et Selvarclust [59]. Les résultats de Clustvarsel et Selvarclust ont été fournis par Cathy Maugis que nous remercions vivement. Pour les trois algorithmes sparseFEM que nous proposons, le modèle DLM et le paramètre de parcimonie λ ont été choisis grâce au critère BIC. Les résultats présentés ont été moyennés sur 20 réplications de l'expérience avec des initialisations aléatoires. Nous avons également reporté les résultats de Fisher-EM à titre de comparaison. Il apparaît tout d'abord que les trois algorithmes sparseFEM se positionnent très favorablement par rapport aux autres approches et cela en particulier en grande dimension (USPS358). On note également que les performances de sparseFEM ne sont que légèrement inférieures à celles de Fisher-EM tout en effectuant une réduction importante du nombre de variables utilisées. Concernant la parcimonie

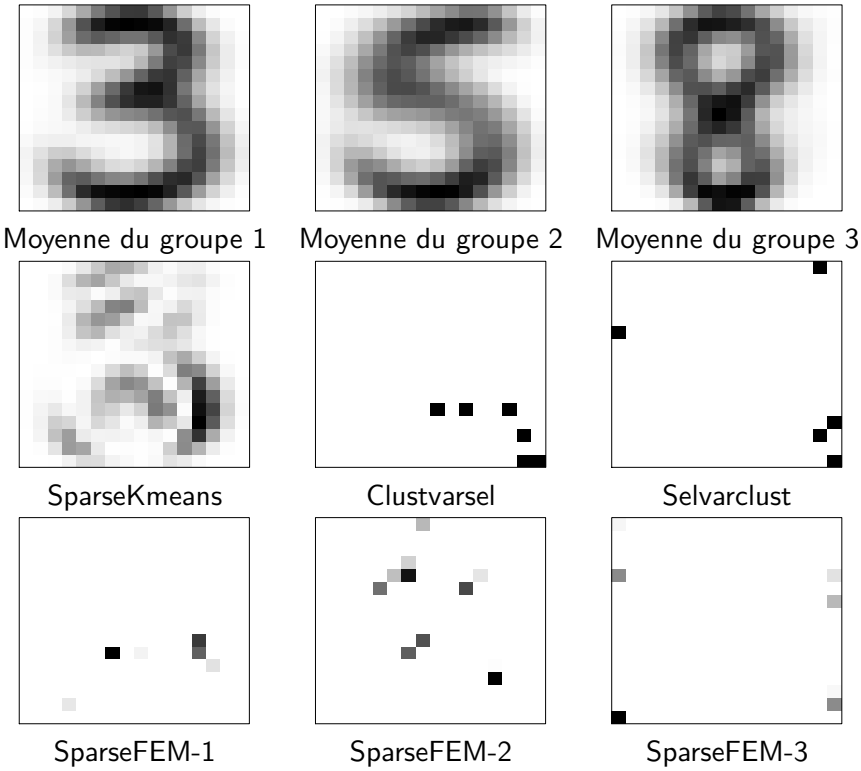


FIGURE 2.12: Moyennes estimées par sparseFEM-1 pour les trois groupes et sélection de variables obtenue avec sparseKmeans, Clustvarsel, Selvarclust et sparseFEM pour les données USPS358 (256 dimensions).

des différentes méthodes, sparseFEM semble réaliser un compromis entre sparseKmeans qui n'est pas vraiment parcimonieux et Clustvarsel / Selvarclust qui paraissent l'être au contraire trop.

Nous nous sommes ensuite concentrés sur le jeu de données USPS358 qui présente l'avantage d'être de grande dimension et pour lequel il est possible de visualiser et d'interpréter facilement la sélection de variables faite. En effet, chaque observation de dimension 256 est en fait la vectorisation d'une image 16×16 en niveaux de gris de chiffres manuscrits (3, 5 ou 8). Il est donc possible de visualiser sous forme d'images à la fois le vecteur moyen estimé pour chaque groupe et la sélection de variables faite par chacune des méthodes étudiées. La figure 2.12 présente les moyennes estimées par sparseFEM pour les trois groupes ainsi que la sélection de variables obtenue avec sparseKmeans, Clustvarsel, Selvarclust et sparseFEM pour les données USPS358. On remarque tout d'abord que sparseFEM réalise des estimations très satisfaisantes des trois groupes de chiffres manuscrits. Il apparaît également que Clustvarsel et Selvarclust sont trop parcimonieux et sélectionnent des variables peu discriminantes. Ce comportement est partagé par sparseFEM-3 qui semble en fait seuiller la matrice U estimée par Fisher-EM. A l'inverse, sparseKmeans s'avère être trop peu parcimonieux (213 variables selectionnées) puisqu'il exclut uniquement les variables où il n'y a pas de variance. En revanche, sparseFEM-1 et sparseFEM-2 réalisent une sélection d'un nombre raisonnable (respectivement 6 et 15) de variables qui s'avèrent de plus très pertinentes pour discriminer les trois groupes considérés. En effet, si l'on examine par exemple la matrice U sparse estimée par sparseFEM-1, le pixel le plus foncé au centre permet de discriminer le chiffre 8 des chiffres 3 et 5. De même, le groupe de pixels foncés en haut à gauche pour sparseFEM-2 permettent de discriminer le chiffre 3 des chiffres 5 et 8. La table 2.4 fournit les temps de calcul nécessaires à chacune des méthodes pour faire le clustering du jeu de données USPS358. Il apparaît que sparseFEM est très compétitif également en terme de temps de calcul par rapport aux autres

Méthode	Temps de calcul	Méthode	Temps de calcul
SparseFEM$_1$	729.1	SparseKmeans	1 567.7
SparseFEM$_2$	387.1	Clustvarsel	2 957.7
SparseFEM$_3$	409.6	Selvarclust	9 257.1

TABLE 2.4: Temps de calculs (en secondes) pour le clustering des données USPS358 par les méthodes sparseFEM, sparseKmeans, Clustvarsel et Selvarclust.

méthodes.

Pour conclure, les trois versions sparses proposées pour Fisher-EM se sont avérées très efficaces pour sélectionner les variables discriminantes et utiles au clustering. Comparées aux meilleures méthodes de l'état de l'art, les algorithmes sparseFEM semblent réaliser un compromis entre trop et trop peu de parcimonie tout en maintenant des performances de clustering très proches de l'algorithme Fisher-EM original. Les versions sparses de Fisher-EM devraient être particulièrement utiles pour l'interprétation des résultats dans les domaines applicatifs où les variables ont une signification propre. Nous avons d'ailleurs appliqué dans [B20] les algorithmes sparseFEM à la segmentation d'images hyperspectrales de la planète Mars dont les résultats seront présentés au chapitre 5.

3

Apprentissage statistique adaptatif

Cette seconde thématique de recherche traite du problème de l'apprentissage statistique dans un contexte évolutif. Dans ce cadre, nous avons développé trois sous-axes qui sont présentés ci-dessous. Le premier considère le problème de la régression quand la population a évolué entre la phase d'apprentissage et la phase de prédiction. Ces travaux ont donné lieu à ce jour à 2 articles méthodologiques [B6, B10], 1 chapitre de livre [B17] et 1 prépublication [B18]. Le second sous-axe a pour objet la classification supervisée avec labels incertains, situation fréquente en pratique mais non considérée jusqu'alors d'un point de vue théorique dans le cadre génératif. Deux articles méthodologiques [B5, B13] sont issus des travaux sur cette thématique. La classification supervisée avec classes non observées est l'objet du troisième sous-axe. Ce problème, également non considéré jusqu'ici d'un point de vue théorique, apparaît dans des contextes de censure ou de classes rares. Ces travaux ont donné lieu, à ce jour, à 1 article méthodologique [B16].

3.1 Modèles adaptatifs pour la régression

La régression linéaire et le mélange de régressions sont deux techniques très populaires pour modéliser le lien entre une variable quantitative Y et une ou plusieurs variables explicatives X. Une hypothèse classique, faite en apprentissage statistique et en particulier en régression, est de supposer que le phénomène étudié n'a pas évolué entre la phase d'apprentissage (construction du régresseur) et la phase de prédiction. Malheureusement, cette hypothèse s'avère souvent fausse en pratique du fait, par exemple, du délai entre les phases d'apprentissage et de prédiction ou de l'évolution entre la population d'apprentissage et celle pour laquelle on souhaite prédire Y. De plus, il peut paraître intéressant dans certaines situations d'utiliser la connaissance sur une population pour inférer un modèle de régression sur une population similaire mais pour laquelle l'accès aux données est difficile (coût d'acquisition, censure, ...). Par exemple, de nombreux travaux récents utilisent des modèles de régression pour inférer des réseaux de régulation entre gènes pour des données de séquençage ADN qui sont caractérisés par leur grand nombre de variables explicatives et leur petit nombre d'observations. Ainsi, si l'on s'intéresse par exemple au réseau de régulation des gènes d'une sous-espèce animale, il serait certainement intéressant de pouvoir utiliser les informations disponibles sur les autres sous-espèces de la même famille d'animaux. Pour cela, il est nécessaire de construire des modèles qui permettent de transférer la connaissance sur le modèle de régression d'une population de référence à celui d'une autre population, légèrement différente.

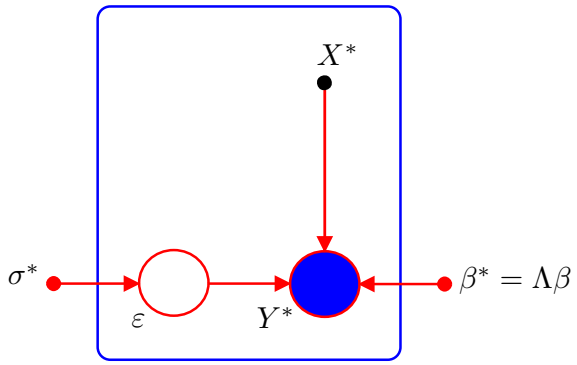


FIGURE 3.1: Représentation graphique du modèle adaptatif paramétrique pour la régression linéaire.

Un tel problème avait déjà été considéré dans le contexte de la classification supervisée. Biernacki *et al.* [13] ont en effet proposé un modèle de transformation paramétrique permettant de classer un jeu de données en adaptant la règle de classification apprise sur une population de référence. Ce travail a été étendu dans [48] à la classification adaptative de données binaires. Dans le contexte de la régression, peu de travaux ont été effectués dans le cadre des méthodes génératives. Les travaux les plus proches [91, 92, 94, 95], que l'on retrouve associés au mot-clé « covariate shift », considèrent le cas d'une évolution de la distribution des variables explicatives. Ces approches nécessitent donc un grand nombre d'observations de la nouvelle population pour estimer de façon fiable la densité des variables explicatives. Le cadre que nous avons considéré dans [B6, B10, B17, B18] est, de ce point de vue, plus général et pourra être appliqué à des situations pratiques où les données sont peu nombreuses.

3.1.1 Modèles adaptatifs paramétriques pour la régression linéaire

Considérons tout d'abord le problème de la régression linéaire sur une base de fonctions $\{\psi_0, ..., \psi_p\}$. Soit $S = \{(x_1, y_1), ..., (x_n, y_n)\}$ un échantillon de données provenant d'une population P pour laquelle on souhaite construire un modèle de régression. La plupart des approches paramétriques et non paramétriques considèrent que les observations de l'échantillon S sont des réalisations indépendantes d'un couple de variables (X, Y), liées par le modèle de régression suivant :

$$Y = f(X, \beta) + \varepsilon, \tag{3.1}$$

où β est le vecteur des régresseurs et le résidu ε est tel que $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Cette modélisation revient à supposer que la distribution conditionnelle de Y est :

$$Y|X \sim \mathcal{N}(f(X, \beta), \sigma^2).$$

Les approches paramétriques supposent généralement que la fonction de lien prend la forme suivante :

$$f(X, \beta) = \sum_{j=0}^{p} \beta_j \psi_j(X),$$

où $\{\psi_0, ..., \psi_p\}$ est une base de fonctions (polynomiales, splines, ondelettes, ...) telle que $\psi_0(x) = 1$. Remarquons que la régression linéaire classique correspond à un choix de ψ où $\psi_j(x) = x^{(j)}$, où $x^{(j)}$ est la $j$ème variable de l'observation x.

Modèle	$M0$	$M1$	$M2$	$M3$	$M4$	$M5$	$M6$
β_0^* est supposé être	β_0	$\lambda_0\beta_0$	β_0	$\lambda\beta_0$	$\lambda_0\beta_0$	β_0	$\lambda_0\beta_0$
β_i^* est supposé être	β_i	β_i	$\lambda\beta_i$	$\lambda\beta_i$	$\lambda\beta_i$	$\lambda_i\beta_i$	$\lambda_i\beta_i$
Nb. de paramètres	0	1	1	1	2	d	d+1

TABLE 3.1: La famille de modèles paramétriques de transformation pour la régression linéaire sur base de fonctions.

Modèles paramétriques de transformation

Considérons à présent un second échantillon de données $S^* = \{(x_1^*, y_1^*), ..., (x_{n^*}^*, y_{n^*}^*)\}$ provenant d'une population P^*, qui est supposée avoir un lien (évolution temporelle, géographique) avec la population P, et pour laquelle on souhaite construire le modèle de régression :

$$Y^* = f^*(X^*, \beta^*) + \varepsilon^*. \tag{3.2}$$

On suppose de plus que f^* est telle que $f^*(X^*) = \sum_{j=0}^{p} \beta_j^* \psi_j(X^*)$ et que le nombre d'observations de S^* est petit et en particulier $n^* \ll n$. Cela nous incite naturellement à vouloir exploiter la connaissance sur le modèle de régression (3.1) pour guider l'estimation du modèle (3.2).

Pour ce faire, nous avons proposé dans [B6] un modèle qui suppose que la transformation entre f et f^* s'exprime au travers du lien entre β et β^*, qui est de plus supposé être de la forme :

$$\beta^* = \Lambda\beta,$$

où Λ est une matrice $(p+1) \times (p+1)$. Ce modèle nécessitant malheureusement l'estimation d'un trop grand nombre de paramètres au regard du petit nombre n^* d'observations, il a été nécessaire de contraindre ce modèle en supposant que $\Lambda = \mathrm{diag}(\lambda_0, ..., \lambda_p)$. Une représentation graphique de ce modèle est présenté par la figure 3.1. Afin de proposer des modèles parcimonieux correspondant à des situations concrètes, nous avons autorisé que certains λ_j soient égaux à une même valeur λ. Cela a donné naissance à une famille de 7 modèles dont les contraintes sont décrites dans le tableau 3.1. On remarque en particulier que tous ces modèles sont très parcimonieux et qu'il a été choisi de dissocier le rôle de l'intercepte β_0 de celui des autres régresseurs. Le modèle $M5$ par exemple suppose que l'intercepte du nouveau modèle de régression est le même que celui du modèle de référence et que les autres régresseurs sont chacun proportionnels aux régresseurs du modèle de référence. Il est également possible de considérer d'autres modèles en utilisant par exemple des informations a priori sur le phénomène à modéliser pour contraindre la matrice de transformation Λ, ce qui revient à proposer un modèle de transformation propre.

Estimation des paramètres

Supposons tout d'abord que les paramètres du modèle (3.1) sont connus ou ont été estimés précédemment et l'on cherche donc à présent à estimer $\beta^* = \Lambda\beta$. Dans le contexte de la régression linéaire comme présentée ci-dessus, la méthode d'estimation du maximum de vraisemblance coïncide avec celle des moindres carrés. Par conséquent, β sera en pratique remplacé par son estimateur des moindres carrés, *i.e.* $\hat{\beta} = (\mathbf{X}^t\mathbf{X})^{-1}\mathbf{X}^t\mathbf{y}$ si $\psi_j(X) = x^{(j)}$ pour $j = 1, ..., p$ et $S = (\mathbf{X}, \mathbf{y})$ est l'échantillon d'observations issu de P.

Par souci de généralisation, nous présentons ci-dessous l'estimation de β^*, au travers de l'estimation de Λ, dans le cas d'une contrainte quelconque de Λ. Dans la suite, les indices γ_j seront associés aux régresseurs de la population P^* qui ont changés, *i.e.* $\beta_{\gamma_j}^* = \lambda_{\gamma_j}\beta_{\gamma_j}$ avec $j = 1, ..., q$ et $\gamma_j \in \{0, ..., p+1\}$. De façon similaire, les indices $\bar{\gamma}_j$ seront associés aux régresseurs de P^* qui

sont égaux à ceux de P , *i.e.* $\beta^*_{\bar{\gamma}_j} = \beta_{\bar{\gamma}_j}$ avec $j = 1, ..., p-q$ et $\bar{\gamma}_j \in 0, ..., p+1$. Le modèle de régression (3.2) peut alors être réécrit comme suit :

$$\mathbf{y}^* = \boldsymbol{Q}\boldsymbol{\Lambda}_q + \bar{\boldsymbol{Q}}\mathbf{1}_{p-q} + \varepsilon,$$

où :

- $\boldsymbol{\Lambda}_q = \left(\lambda_{\gamma_1}, \ldots, \lambda_{\gamma_q}\right)^t$,
- $\boldsymbol{Q} = \begin{pmatrix} \beta_{\gamma_1}\psi_{\gamma_1}(x_1^*) & \cdots & \beta_{\gamma_q}\psi_{\gamma_q}(x_1^*) \\ \vdots & & \vdots \\ \beta_{\gamma_1}\psi_{\gamma_1}(x_n^*) & \cdots & \beta_{\gamma_q}\psi_{\gamma_q}(x_n^*) \end{pmatrix}$,
- $\mathbf{1}_{p-q}$ est le vecteur unité de dimension $p-q$,
- $\bar{\boldsymbol{Q}} = \begin{pmatrix} \beta_{\bar{\gamma}_1}\psi_{\bar{\gamma}_1}(x_1^*) & \cdots & \beta_{\bar{\gamma}_q}\psi_{\bar{\gamma}_q}(x_1^*) \\ \vdots & & \vdots \\ \beta_{\bar{\gamma}_1}\psi_{\bar{\gamma}_1}(x_n^*) & \cdots & \beta_{\bar{\gamma}_q}\psi_{\bar{\gamma}_q}(x_n^*) \end{pmatrix}$.

Proposition 8. *Avec les notations précédentes, l'estimateur des moindres carrés de $\boldsymbol{\Lambda}_q$ est :*

$$\hat{\boldsymbol{\Lambda}}_q = \left(\boldsymbol{Q}^t\boldsymbol{Q}\right)^{-1}\boldsymbol{Q}^t\left(\mathbf{y}^* - \bar{\boldsymbol{Q}}\mathbf{1}_{p-q}\right).$$

La preuve de cette proposition est donnée dans [B6]. L'estimateur de β^* se déduit ensuite facilement de celui de $\boldsymbol{\Lambda}_q$ en utilisant la méthode du *plug-in*. Les estimateurs des paramètres des 7 modèles spécifiques se déduisent également de ce résultat et ceux-ci sont également donnés dans [B6]. Nous avons de plus proposé dans [B6] une estimation jointe des paramètres β et β^*, basée sur une approche itérative, de sorte que l'échantillon S^* participe également à améliorer l'estimation du modèle de régression (3.1).

Choix du modèle de transformation

Le choix du modèle de transformation le plus approprié pour les données considérées est un problème important. Nous avons proposé d'utiliser l'un des trois critères classiques suivant pour résoudre ce problème. Le premier critère est le critère PRESS [4] qui estime la somme des erreurs de prédiction au carré par validation croisée *leave-one-out* :

$$PRESS = \frac{1}{n^*}\sum_{j=1}^{n^*} ||y_j^* - \widehat{y_j^*}^{-j}||^2$$

où $\widehat{y_j^*}^{-j}$ est la prédiction de y_j^* par le modèle de régression estimé sans le $j$ème individu y_j^* de S^*. Le critère PRESS est certainement l'un des critères les plus utilisés pour la sélection de modèle en régression et nous encourageons son utilisation quand cela est faisable numériquement. Si le coût calculatoire de PRESS est trop élevé pour les données considérées, il est possible d'utiliser les critères de vraisemblance pénalisée AIC [2] ou BIC [87] qui se définissent par :

$$\mathrm{AIC} = -2\ln\ell + 2\nu, \qquad \mathrm{BIC} = -2\ln\ell + \nu\ln n^*,$$

où ℓ est la valeur de la vraisemblance en $\hat{\beta}^*$ et ν est le nombre de paramètres du modèle considéré (voir le tableau 3.1). Notons que ces trois critères pourront être également utilisés dans le cas du mélange de régressions.

3.1.2 Modèles adaptatifs paramétriques pour le mélange de régressions

Nous avons ensuite adapté dans [B18] les modèles adaptatifs paramétriques au cadre du mélange de régressions. Le modèle du mélange de régressions, connu également sous le nom de « *switching regression* », est un modèle très populaire par exemple en Économie où il est utilisé pour modéliser

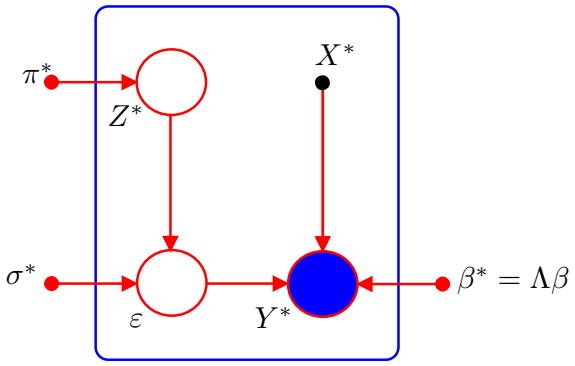


FIGURE 3.2: Représentation graphique du modèle adaptatif paramétrique pour le mélange de régressions.

des phénomènes à plusieurs états. Ce modèle suppose que la variable à prédire Y peut être liée aux variables explicatives $X = (1, x^{(1)}, ..., x^{(p)}) \in \mathbb{R}^{p+1}$ par un des K modèles de régressions suivants :

$$Y = X^t\beta_k + \sigma_k\varepsilon,\ k = 1, ..., K \tag{3.3}$$

de probabilités a priori $\pi_1, \ldots, \pi_K$ (avec $\sum_{i=1}^{K} \pi_k = 1$), où $\varepsilon \sim \mathcal{N}(0, 1)$, $\beta_k = (\beta_{k0}, ..., \beta_{kp}) \in \{\beta_1, \ldots, \beta_K\}$ et $\sigma_k^2 \in \{\sigma_1^2, \ldots, \sigma_K^2\}$. La distribution conditionnelle de Y est alors :

$$p(y|x) = \sum_{k=1}^{K} \pi_k \phi(y|x^t\beta_k, \sigma_k^2), \tag{3.4}$$

où $\phi(\cdot)$ est la densité de loi normale.

Modèles paramétriques de transformation

Afin de modéliser le lien entre le mélange de régression d'une population de référence P et celui d'une population P^*, nous avons fait les même hypothèses que dans le cas de la régression simple mais ce pour chacune des régressions du mélange. Nous avons de plus supposé que les modèles de régression des populations P et P^* ont le même nombre de composantes, *i.e.* $K = K^*$. Le modèle paramétrique de transformation pour la $k$ème composante du mélange de régression de P^* s'écrit :

$$\boldsymbol{\beta}_k^* = \Lambda_k\boldsymbol{\beta}_k, \text{ où } \Lambda_k = \text{diag}(\lambda_{k0}, \lambda_{k1}, \ldots, \lambda_{kd}) \quad \text{et} \quad \sigma_k^* \text{ est libre.}$$

Une représentation graphique de ce modèle est présenté par la figure 3.2. De même que précédemment, nous avons introduit des contraintes sur les matrices Λ_k et sur les paramètres σ_k et cela au sein de chacune des composantes du mélange mais également entre les composantes. Cela a donné naissance à 12 modèles de transformation qui sont présentés en détail dans [B18] et dont le tableau 3.2 donne le nombre de paramètres. Le modèle MM_1 suppose que $\Lambda_k = I_d$. Les modèles MM_2 supposent que le lien entre les modèles de régression est indépendant des variables et des composantes. Les modèles MM_3 supposent que le lien est uniquement indépendant des composantes alors que les modèles MM_4 supposent qu'il est uniquement indépendant des variables. Enfin, le modèle MM_5 suppose que les modèles de régression des deux populations sont indépendants. Des variantes de chacun des modèles (indicées de a à d) peuvent être obtenues en contraignant par exemple λ_{k0} à être égal à 1 ou σ_k^* à être égal à σ_k.

Modèle	MM_1	MM_{2a-c}	MM_{2d}	MM_{3a-c}	MM_{3d}	MM_{4a}	MM_{4b}	MM_5
Nb de paramètres	0	1	2	K	$2K$	$d+K$	$d+K+1$	$K(d+2)$

TABLE 3.2: Nombre de paramètres des modèles de transformation dans le cas du mélange de régressions.

Estimation des paramètres

Dans le contexte du mélange de régression, il n'est pas possible de maximiser directement la vraisemblance du modèle car les données $S^* = \{(x_1^*, y_1^*), ..., (x_{n^*}^*, y_{n^*}^*)\}$ sont incomplètes, *i.e.* l'appartenance aux composantes du mélange est inconnue. Dans une telle situation, il est nécessaire d'introduire une variable aléatoire $Z \in \{0,1\}^K$ non observée telle que $z_{ik} = 1$ indique que l'observation $((x_i^*, y_i^*))$ appartient à la $k$ème composante du mélange. Il est alors possible d'utiliser l'algorithme EM [29] pour maximiser itérativement la vraisemblance du modèle. L'algorithme EM prend alors la forme suivante dans le cas du modèle MM_{2b}.

Proposition 9. *Dans le cas du modèle MM_{2b}, l'algorithme EM prend la forme suivante à l'étape q :*

- *l'étape E calcule l'espérance conditionnelle de la log-vraisemblance complétée, ce qui revient au calcul des probabilités a posteriori t_{ik} :*

$$t_{ik}^{(q)} = P(z_{ik}^* = 1 | \boldsymbol{y}^*, \boldsymbol{x}^*) = \frac{\pi_k^{*(q)} \phi(y_i^* | x_i^{*t} \Lambda_k^{(q)} \beta_k, \sigma_k^{*2(q)})}{\sum_{l=1}^K \pi_l^{*(q)} \phi(y_i^* | x_i^{*t} \Lambda_l^{(q)} \beta_l, \sigma_l^{*2(q)})}.$$

- *l'étape M maximise l'espérance conditionnelle de la log-vraisemblance complétée pour fournir les mises à jour des paramètres π_k^*, Λ_k et σ_k^*. Dans le cas du modèle MM_{2b}, l'estimateur de λ est :*

$$\hat{\lambda} == \left(\sum_{i=1}^{n^*} \sum_{k=1}^{K} \frac{t_{ik}^{(q)}}{\sigma_k^2} {\beta_{k0}}^2 \right)^{-1} \sum_{i=1}^{n^*} \sum_{k=1}^{K} \frac{t_{ik}^{(q)}}{\sigma_k^2} (y_i^* - x_{i\sim 0}^{*t} \beta_{k\sim 0}) \beta_{k0},$$

où l'indice ~ 0 signifie que le vecteur est privé de son premier élément (celui indexé par 0).

La démonstration de ce résultat est donnée dans [B18]. Les formules de mise à jour des paramètres estimés dans le cas des autres modèles sont également données dans [B18].

3.1.3 Modèles adaptatifs bayésiens pour le mélange de régressions

Nous avons également proposé dans [B18] une modélisation bayésienne du lien entre les modèles de régression des populations P et P^*. En effet, puisque les deux populations sont supposées avoir un lien, il semble naturel d'encoder ce lien au travers d'une information a priori dans un cadre bayésien.

Modélisation bayésienne du mélange adaptatif de régressions

Nous avons donc proposé que la distribution a priori de β_k^*, $k = 1, ..., K$, soit :

$$\beta_k^* \sim \mathcal{N}(\beta_k, \sigma_k^{*2} A_k),$$

où A_k est une matrice $(p+1) \times (p+1)$ (A_k pourra en particulier être supposée égale à I_{p+1}) et que la distribution a priori de π^* soit une loi de Dirichelet :

$$\pi^* \sim \mathcal{D}(\pi_1, ..., \pi_K).$$

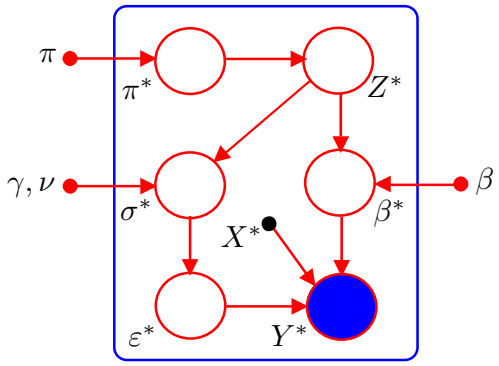


FIGURE 3.3: Représentation graphique du modèle adaptatif bayésien pour le mélange de régressions.

La loi a priori de σ_k^* est quant à elle supposée être une loi inverse-gamma $\mathcal{IG}(\gamma_k, \nu_k)$ pour $k = 1, ..., K$. Avec ces hypothèses, les paramètres du modèle de régression de la population P^* sont naturellement liés à ceux de la population P. Dans le contexte du mélange de régressions bayésiennes, il est en outre classique de supposer l'indépendance conditionnelle entre les proportions π^*et à la fois les régresseurs β^* et les variances σ^*. L'indépendance est également supposée entre les paramètres (β_k^*, σ_k^*) de deux composantes différentes du mélange. Une représentation graphique de cette modélisation est présentée à la figure 3.3. L'inférence d'un tel modèle est malheureusement infaisable, même pour des échantillons de petite taille, et l'utilisation d'une méthode MCMC est nécessaire.

Échantillonneur de Gibbs pour le mélange adaptatif de régressions

Nous avons proposé d'utiliser l'échantillonneur de Gibbs pour inférer le modèle de transformation ci-dessus. Pour cela, il a été à nouveau nécessaire d'adjoindre à la modélisation ci-dessus une variable cachée $Z \in \{0, 1\}^K$ représentant l'appartenance des observations aux K composantes du mélange. L'algorithme de Gibbs échantillonne alors, à l'itération q, des valeurs des paramètres selon les distributions a posteriori conditionnelles suivantes :

- la distribution a posteriori conditionnelle de Z^* est une loi multinomiale :

$$z_i^* | Y^*, \hat{\beta}, \hat{\pi}, \pi^*, \beta^*, \sigma^{*2} \sim \mathcal{M}(1, t_{i1}, ..., t_{iK}),$$

  où $t_{ik} = \pi_k^* \phi(y_i^* | x_i^{*t} \beta_k^*, \sigma_k^{*2}) / \sum_{\ell=1}^K \pi_\ell^* \phi(y_i^* | x_i^{*t} \beta_\ell^*, \sigma_\ell^{*2})$.
- la distribution a posteriori conditionnelle de π^* est une loi Dirichlet :

$$\pi^* | Y^*, \hat{\beta}, \hat{\pi}, Z^*, \beta^*, \sigma^{*2} \sim \mathcal{D}(\hat{\pi}_1 + n_1^*, ..., \hat{\pi}_K + n_K^*),$$

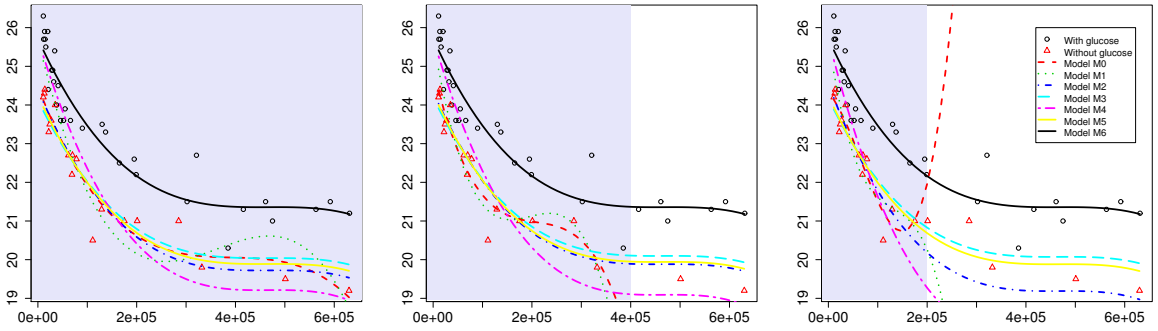
  avec $n_k^* = \sum_{i=1}^n z_{ik}^*$.
- une fois les appartenances aux composantes estimées, il est possible en appliquant la règle du maximum a posteriori de réunir les observations d'une même composante k dans les matrices x_k^* et Y_k^*, $k = 1, ..., K$. Avec ces notations, la distribution a posteriori conditionnelle de σ_k^{*2} est une loi inverse gamma :

$$\sigma_k^{*2} | Y^*, \hat{\beta}, \hat{\pi}, Z^*, \pi^*, \beta_k^* \sim \mathcal{IG}(\gamma_k + n_k/2, \nu_k + S_k/2),$$

  où $S_k = (Y_k^* - x_k^{*t} \beta_k^*)^t (Y_k^* - x_k^{*t} \beta_k^*) + (\hat{\beta}_k - \beta_k^*)^t (A_k + (x_k^{*t} x_k^*)^{-1})^{-1} (\hat{\beta}_k - \beta_k^*)$.
- enfin, la distribution a posteriori conditionnelle de β_k^* est une loi normale :

$$\beta_k^* | Y^*, \hat{\beta}, \hat{\pi}, Z^*, \pi^*, \sigma_k^{*2} \sim \mathcal{N}(m_k, \Delta_k),$$

  avec $m_k = (A_k^{-1} + x_k^{*t} x_k^*)^{-1} (x_k^{*t} Y_k^* + A_k^{-1} \hat{\beta}_k)$ et $\Delta_k = \sigma_k^{*2} (x_k^{*t} x_k^* + A_k^{-1})^{-1}$.



Toutes les données $X^* \leq 4 \times 10^5$ $X^* \leq 2 \times 10^5$

FIGURE 3.4: Effet de la censure sur P^* sur les modèles adaptatifs de régression pour les données hellung. La zone bleue correspond à la partie des observations de S^* utilisée pour l'estimation.

Des estimateurs consistants des paramètres π^*, β^* et σ^{*2} sont ensuite obtenus en moyennant les paramètres simulés sur les dernières $Q - q_0$ itérations, où q_0 est le nombre d'itérations de la phase dite de « burning ». Nous discutons également dans [B18] le problème du « label switching » dû à la possible multimodalité de la distribution a posteriori. Pour pallier ce problème, nous recommandons l'utilisation de la solution proposée par Celeux *et al.* [24].

3.1.4 Expérimentations numériques

Pour vérifier l'utilité des modèles adaptatifs proposés, nous avons tout d'abord considéré dans [B6] un modèle de censure s'appliquant à une nouvelle population P^*. Dans un tel contexte, la connaissance sur le modèle de régression d'une population P de référence doit permettre, grâce aux modèles adaptatifs proposés, de régulariser l'estimation du modèle de P^*. Pour vérifier cela, nous avons utilisé le jeu de données hellung, disponible dans le paquet ISwR pour R, qui reporte l'évolution de cellules *Tetrahymena* dans deux conditions de cultures : avec ou sans glucose. Dans les deux cas, le diamètre moyen et la concentration de la cellule ont été enregistrés au cours du temps. Pour l'expérience, nous avons considéré les cellules avec glucose ($n = 32$ observations) comme provenant de la population P de référence et celles sans glucose ($n^* = 19$ observations) comme provenant de la population P^*. Nous avons ensuite utilisé les modèles adaptatifs pour estimer le modèle de régression de P^* et ce d'une part avec toutes les données de S^* (pas de censure) et d'autre part en n'utilisant que la partie de données dont la concentration est plus petite qu'une certaine valeur (modèle de censure). La figure 3.4 présente les modèles de régression estimés dans le cas sans censure et dans deux cas avec censure. La base de fonction utilisée ici est composée de fonctions polynomiales allant jusqu'à l'ordre 3. Les modèles adaptatifs de régression sont ici comparés au modèle de régression classique directement estimé par moindres carrés sur S^*. Il apparaît très nettement que la censure sur S^* a un effet très marqué sur l'estimateur des moindres carrés (modèle M_0) alors que les modèles adaptatifs qui utilisent la connaissance sur P fournissent des estimateurs très stables et ce même avec une très forte censure.

Nous avons également étudié dans [B18] l'effet de la censure dans le contexte du mélange de régressions. Pour cela, nous avons simulé des données selon un modèle de mélange à deux composantes avec une base de fonction polynomiales d'ordres 1 et 2. La figure 3.5 présente en haut les données simulées issues de P (ronds gris) et les 20 observations issues de P^* (triangles rouges). Le modèle de transformation utilisé ici est le modèle MM_{2c} avec $\lambda = 3$. La figure présente également les modèles de régression construits par les méthodes MR (mélange de régressions), AMRp et AMRb (mélange adaptatif de régressions paramétrique et bayésien). La seconde rangée

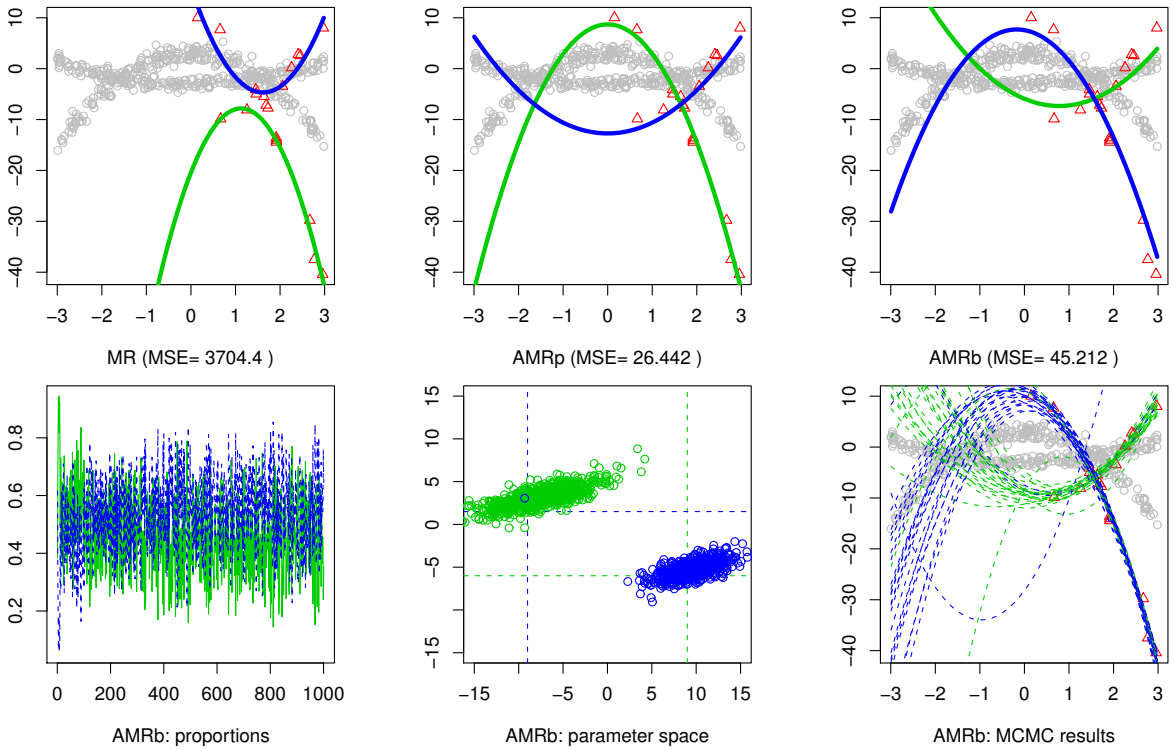


FIGURE 3.5: En haut : modèles de régression estimés pour un jeu de données simulées avec les méthodes MR, AMRp et AMRb. En bas et de gauche à droite : proportions estimées au cours des itérations MCMC, échantillonnage de β_k^* dans l'espace des paramètres et modèles de régression associés. Voir le texte pour plus de détails.

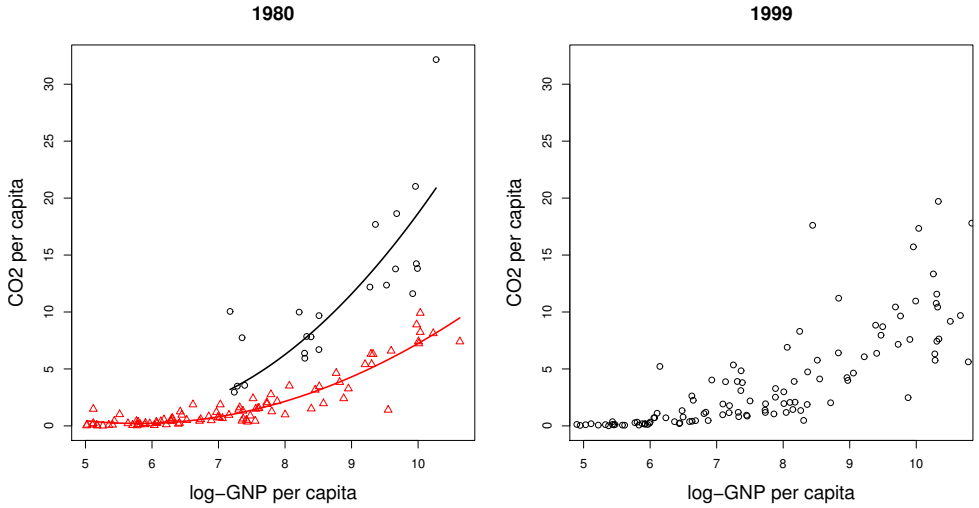


FIGURE 3.6: Emission de CO_2 en fonction du PIB de 111 pays en 1980 (à gauche) et 1999 (à droite).

Méthode	30% de S^*	50% de S^*	70% de S^*	100% de S^*
AMRp	3.86	3.44	3.53	3.47
AMRb	5.99	5.66	5.99	5.66
UR	7.66	7.21	7.10	6.99
MR	5.11	4.77	3.33	2.89

TABLE 3.3: Erreur quadratique moyenne (MSE) calculé sur un jeu de validation pour le modèle de régression estimé sur données de 1999 en fonction de la taille de S^* sur le jeu de données GNP-CO2. Les résultats reportés pour AMRp correspondent à ceux du modèle choisit par BIC.

d'images montre l'échantillonnage des paramètres π^* et β_k^* au cours des itérations MCMC ainsi que quelques modèles de régression associés. Il apparaît tout d'abord que, comme attendu, la méthode MR sur-apprend les données et produit un modèle de régression très éloigné de la vérité. Les méthodes AMRp et AMRb, qui utilisent toutes deux la connaissance sur P en plus de l'échantillon S^* pour estimer le modèle de régression de P^*, fournissent des estimations très satisfaisantes avec respectivement 26.4 et 45.3 d'erreur quadratique moyenne (MSE).

Nous avons également appliqué dans [B18] les modèles adaptatifs à la modélisation de données économiques et, de plus, les résultats ont été analysés d'un point de vue économique dans [B10], article co-écrit avec Patrice Gaubert, professeur d'Économie. Les données utilisées pour cette étude sont les émissions de CO_2 et le PIB de 111 pays mesurés en 1980 et en 1999. La figure 3.6 présente ces données pour chacune des deux années. Nous avons utilisé une base de fonctions polynomiales d'ordres 1 et 2 et comparé les modélisations faites par les méthodes UR (régression simple), MR, AMRp et AMRb. L'année 1980 a été utilisée comme année de référence (population P) et l'année 1999 comme nouvelle population (P^*). Le tableau 3.3 présente les erreurs quadratiques moyennes (MSE) en fonction de la taille de S^* pour les 4 méthodes étudiées. Les résultats reportés pour AMRp correspondent à ceux du modèle choisi par BIC. Il apparaît à nouveau que dans le cas où l'on dispose de peu de données, les modèles adaptatifs (AMRp en particulier) permettent de fournir une meilleure estimation du modèle de régression que les modèles travaillant uniquement sur l'échantillon S^*. Le modèle adaptatif choisi peut de plus apporter un éclairage dans l'analyse du phénomène étudié. Dans cette application, le modèle $MM2$ a été choisi ce qui indique que les deux sous-populations ont été modifiées de manière similaire au cours du temps.

Pour conclure, les modèles adaptatifs proposés dans [B6, B10, B17, B18] pour la régression linéaire et le mélange de régressions se sont avérés particulièrement utiles dans des situations où peu de données sont disponibles pour modéliser une nouvelle population P^*. De telles situations incluent les données censurées ou les données de coût d'acquisition élevé. Dans de tel cas, les modèles adaptatifs permettent de transférer efficacement la connaissance sur une population de référence pour améliorer la modélisation de P^*.

3.2 Classification supervisée avec labels incertains

Dans le contexte de l'apprentissage statistique adaptatif, nous nous sommes également intéressés au problème de la classification supervisée avec des labels incertains. En classification supervisée, la supervision par des experts humains est fréquemment requise pour associer aux données de l'échantillon d'apprentissage des étiquettes (labels) indiquant l'appartenance des observations aux classes. Sur la foi de ces labels, les méthodes de classification supervisée, génératives ou discriminatives, construisent un classifieur qui permettra de prédire la classe d'une nouvelle observation non étiquetée. Cependant, dans de nombreuses applications, la supervision par des

experts humains peut s'avérer imprécise, difficile ou coûteuse. Par exemple, en imagerie médicale, les médecins doivent étiqueter manuellement un grand nombre d'images d'apprentissage et il est évident que la qualité de la supervision baisse avec la fatigue de l'expert. Des erreurs sur les labels des données d'apprentissage peuvent malheureusement avoir des effets importants sur la performance du classifieur et cela plus encore si les données d'apprentissage sont peu nombreuses. Il nous a donc parut important de proposer des classifieurs supervisés qui prennent en compte une possible incertitude sur les labels.

Aussi important soit ce problème, il s'avère que le problème du bruit sur la variable à prédire a été très peu étudié en classification alors même que le problème du bruit sur les variables explicatives a reçu beaucoup d'attention dans la littérature. Les solutions existantes proposent soit de supprimer les données corrompues [27, 42, 46], soit d'utiliser des estimateurs robustes [7, 69, 82]. Ces approches n'ont toutefois, et de l'avis même de certains auteurs, pas permis de prendre en compte correctement le bruit sur les labels. Le travail de Lawrence et Shölkopf [52], étendu plus tard par [54], présente l'avantage de modéliser explicitement le bruit sur les labels. Cependant, ce travail a été mené uniquement dans le cadre de la classification discriminative binaire et cela s'avère relativement limitatif en terme d'usage. Nous avons donc proposé dans [B5] et [B13] des modèles probabilistes de classification robustes au bruit de labels.

3.2.1 Classification robuste par mélange de mélanges

Nous avons tout d'abord proposé dans [B5] une méthode, baptisée *robust mixture discriminant analysis* (RMDA), qui repose sur un mélange de mélanges. L'idée est de comparer la modélisation supervisée induite par les labels observés à une modélisation obtenue dans un contexte non supervisée. Les inconsistances entre les deux modélisations permettront de détecter les labels incertains et ceux-ci se verront attribuer un poids faible dans la construction du classifieur robuste.

Le mélange de mélanges

Considérons un modèle de mélange où deux structures de classes coexistent : une structure non supervisée de K groupes, représentée par la variable aléatoire Z, et une structure supervisée à C classes, induite par les labels observés et représentée par la variable aléatoire S. Nous supposons classiquement que les observations $\{x_1, ..., x_n\} \in \mathcal{X}$ sont des réalisations indépendantes d'un vecteur aléatoire X de densité :

$$p(x) = \sum_{k=1}^{K} P(Z = k) p(x|Z = k). \tag{3.5}$$

Puisque $\sum_{\ell=1}^{C} P(S = \ell|Z = k) = 1$, nous pouvons introduire l'information supervisée dans l'équation précédente, ce qui donne :

$$p(x) = \sum_{\ell=1}^{C} \sum_{k=1}^{K} r_{k\ell} \pi_k p(x|Z = k), \tag{3.6}$$

où $\pi_k = P(Z = k)$ et $r_{k\ell} = P(S = \ell|Z = k)$, qui peut d'ailleurs s'interpréter comme la probabilité que le $k$ème groupe appartienne à la ℓème classe. La figure 3.7 présente la représentation graphique de ce modèle qui peut de plus être utilisé dans différents contextes en adaptant les distributions conditionnelles $p(x|Z = k)$ au cas considéré. Dans de nombreuses situations, le modèle de mélange gaussien sera tout à fait adapté. Dans le cas de données de grande dimension, on pourra utiliser les modèles parcimonieux gaussiens présentés au chapitre 2. Un mélange de lois multinomiales pourra également être utilisé si les données considérées sont qualitatives. Nous avons notamment

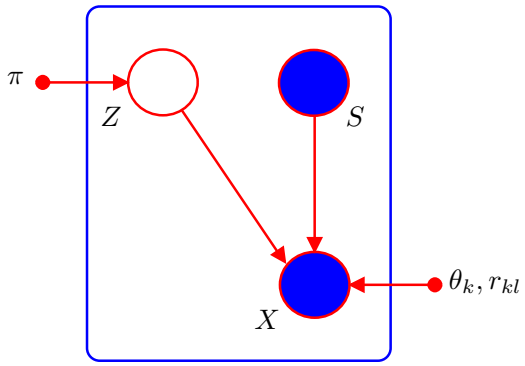


FIGURE 3.7: Représentation graphique du modèle de RMDA.

appliqué dans [20] un tel modèle pour la classification avec labels incertains de séquençages ADN. On notera d'autre part que le modèle proposé ci-dessus consiste en un mélange de mélanges et qu'il généralise le modèle de MDA [45] où chaque classe est modélisée par un mélange. Le modèle MDA peut en effet être retrouvé en forçant les $r_{k\ell}$ à être égaux à 0 ou 1.

La classification de nouvelles observations non étiquetées peut être faite en utilisant la règle du maximum a posteriori qui affecte l'observation x à la classe la plus probable a posteriori. Dans le cas du modèle proposé ci-dessus, la règle de classification prend la forme suivante :

$$\delta(x) = \underset{\ell=1,...,C}{\operatorname{argmax}} P(S = \ell | X = x) = \underset{\ell=1,...,C}{\operatorname{argmax}} \sum_{k=1}^{K} r_{k\ell} P(Z = k | X = x).$$

On remarque que la règle de classification dépend des probabilités $r_{k\ell}$. Ainsi, le classifieur associé à cette règle de classification reposera principalement sur les groupes qui sont les plus probables d'être composés d'observations de la même classe.

Procédure d'estimation

Du fait de la nature du modèle proposé ci-dessus, la procédure d'estimation comprend deux étapes : la première consiste en l'estimation du modèle non supervisé et la seconde en l'estimation des $r_{k\ell}$ dans un cadre supervisé. La première étape vise donc à estimer de façon non supervisée le modèle de mélange (3.5), ce qui consiste en l'estimation des proportions du mélange π_k ainsi que des paramètres des densité conditionnelles $p(x|Z = k)$, $k = 1, ..., K$. Comme nous sommes à nouveau en présence d'un problème à variable cachée, la vraisemblance de ce modèle n'est pas directement maximisable et il est nécessaire d'utiliser l'algorithme EM. Une fois la modélisation non supervisée réalisée, il reste à estimer la matrice $R = (r_{k\ell})_{k,\ell}$, de taille $C \times K$, en utilisant les labels observés. La log-vraisemblance du modèle supervisé (3.6) s'écrit :

$$\log(L(R)) = \sum_{\ell=1}^{C} \sum_{x \in \mathcal{C}_\ell} \log P(X = x, S = \ell) = \sum_{\ell=1}^{C} \sum_{x \in \mathcal{C}_\ell} \log \left(\sum_{k=1}^{K} r_{k\ell} P(Z = k | X = x) \right) + \xi,$$

où ξ est une quantité indépendante de R. Cette maximisation doit de plus être faite sous les contraintes que $r_{k\ell} \in [0, 1]$, $k = 1, ..., K$, $\ell = 1, ..., C$ et que $\sum_{\ell=1}^{C} r_{k\ell} = 1$, $k = 1, ..., K$. Un tel problème d'optimisation n'ayant pas de solution explicite, il nous a été nécessaire d'utiliser une méthode d'optimisation numérique. La fonction *fmincon* de Matlab permet en particulier de résoudre un tel problème.

Choix du nombre de sous-classes

La sélection du nombre de sous-classes est généralement un problème complexe, car les différentes classes peuvent avoir un nombre différent de composantes. Par exemple, dans le cas de MDA, il n'est pas possible d'essayer toutes les combinaisons de nombre de composantes car celui-ci sera le plus souvent bien trop élevé. Dans [45], les auteurs ont proposé de faire l'hypothèse supplémentaire que le nombre de composantes des classes est égal, mais cette hypothèse pourrait s'avérer trop restrictive dans de nombreux cas. En raison de la nature du modèle RMDA, le choix du nombre de sous-classes se réduit au choix du nombre total de groupes K. En effet, la sélection du nombre de sous-classes par classe se fera implicitement par l'intermédiaire du paramètre $r_{k\ell}$ qui quantifie la cohérence entre les sous-classes et les composantes du mélange. Le choix de K peut-être facilement résolu en utilisant par exemple la validation croisée ou le critère BIC.

3.2.2 Classification robuste par réduction de dimension

Dans [B13], nous avons proposé une approche différente qui utilise le modèle DLM, présenté au paragraphe 2.3, dans le cadre de la classification supervisée et semi-supervisée. Cela a donné naissance à une méthode de classification baptisée *probabilistic Fisher discriminant analysis* (PFDA) et celle-ci s'est avérée particulièrement robuste au bruit de labels.

Le modèle probabiliste et son lien avec celui de FDA

Nous avons dans [B13] repris le modèle latent discriminant (DLM), utilisé au paragraphe 2.3 pour la classification non supervisée de données de grande dimension, et nous l'avons appliqué à la classification supervisée puis semi-supervisée. Dans ce cadre, le modèle DLM peut être vu comme une version probabiliste du célèbre modèle de *Fisher discriminant analysis* (FDA) [33]. En effet, FDA cherche les $d = K - 1$ axes discriminants maximisant le critère :

$$J(U) = \mathrm{trace}((U^t S_W U)^{-1} U^t S_B U),$$

où S_W et S_B sont respectivement les matrices de covariance inter et intra-classes, puis construit un classifieur (généralement LDA) dans cet espace de faible dimension. FDA est une méthode très populaire et qui fournit le plus souvent de bons résultats en pratique. Toutefois, FDA possède deux limitations principales : l'optimalité des axes discriminants est prouvée uniquement dans le cas homoscédastique (égalité des matrices de covariances des classes) et FDA est connue pour être sensible au bruit de label et ce particulièrement quand les données d'apprentissage sont peu nombreuses. De ce point de vue, PFDA pallie ces deux limitations de FDA. En effet, le modèle DLM autorise différentes structures de covariances au travers de ses sous-modèles. En particulier, 9 des 12 modèles DLM (*cf.* table 1 de [B13]) supposent des matrices de covariance différentes entre les classes. D'autre part, au contraire de FDA qui conserve uniquement les d axes discriminants, PFDA prend en compte et modélise l'information supposée non discriminante grâce au terme de bruit ε. Ainsi, si les labels sont bruités, les axes qui auraient été écartés par FDA car supposés, à tort, non discriminants seront conservés par PFDA avec une pondération adaptée.

Inférence dans les cas supervisé et semi-supervisé

L'estimation des paramètres du modèle de PFDA est directe dans le cas supervisé. En effet, les données étant dans ce contexte complètes, la maximisation de la vraisemblance est possible et fournit des estimateurs explicites des paramètres du modèle. Seule l'estimation de la matrice de projection U reste itérative du fait de la contrainte d'orthogonalité imposée par la modèle DLM. Dans le cas semi-supervisé, *i.e.* n_ℓ observations du jeu d'apprentissage possèdent des labels et $n - n_\ell$

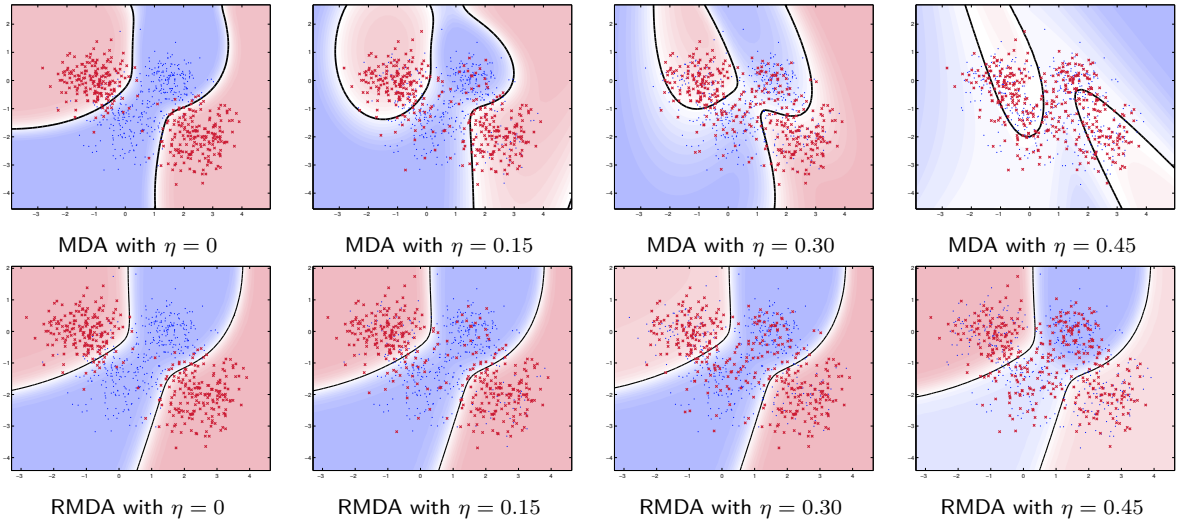


FIGURE 3.8: Règles de classification apprises par MDA et RMDA pour des taux de contamination η croissants sur un jeu de données simulé en dimension 2 (2 classes, 2 composantes par classe).

ne sont pas étiquetées, la vraisemblance n'est à nouveau pas maximisable directement et il est nécessaire d'utiliser une version légèrement modifiée de l'algorithme Fisher-EM (*cf.* paragraphe 2.3) pour l'inférence. La modification de Fisher-EM intervient au niveau de l'étape E qui, dans la version semi-supervisée, estime à chaque itération uniquement les probabilités a posteriori t_{ik} pour les $n - n_\ell$ observations non étiquetées (les autres restent fixes et dépendent des labels observés).

3.2.3 Expérimentations numériques

Nous avons tout d'abord voulu comparer dans [B5] le comportement de RMDA à celui de MDA face au bruit de labels. Pour cela nous avons simulé un jeu de données en 2 dimensions composé de 2 classes, chaque classe comportant 2 sous-classes. La simulation du bruit de labels a été faite par tirage aléatoire de η étiquettes pour lesquelles l'appartenance à la classe a été modifiée par tirage aléatoire selon une loi multinomiale de probabilités égales. La figure 3.8 présente les règles de classification apprises par RMDA et MDA pour différents degrés η de contamination. On remarque tout d'abord que si il n'y a pas de bruit de labels ($\eta = 0$), RMDA fournit une règle de classification très proche de celle de MDA. Quand le taux de contamination η augmente, MDA se trouve très vite perturbé et sa règle de classification s'éloigne rapidement de la règle optimale. A l'inverse, la règle de classification de RMDA s'avère être très robuste au bruit de labels et cela même dans des cas très perturbés ($\eta = 0.45$).

Nous avons ensuite comparé dans [B5] RMDA aux méthodes robustes de l'état de l'art et cela sur un jeu de données réel afin de ne favoriser aucune méthode. Cette comparaison a été reprise dans [B13] sur le même jeu de données et avec le même protocole expérimental. Nous présentons à la figure 3.9 la comparaison des taux de classification correcte de PFDA et RMDA avec ceux des meilleures méthodes robustes de l'état de l'art sur le jeu de données USPS24 (2 classes, 256 dimensions). La qualité de prédiction est évaluée par le taux de classification correcte sur un jeu de validation et les résultats ont été moyennés sur 25 réplications. Les barres verticales indiquent les écarts-types associés. Il apparaît tout d'abord que FDA et MDA, comme attendu, sont très sensibles au bruit de labels. La méthode robuste RLDA [52] de Lawrence et Shölkopf s'avère en effet robuste à des bruit de labels modérés mais RMDA et PFDA se montrent beaucoup plus robustes quand le taux de contamination est très élevé. Le comportement général de RMDA s'est

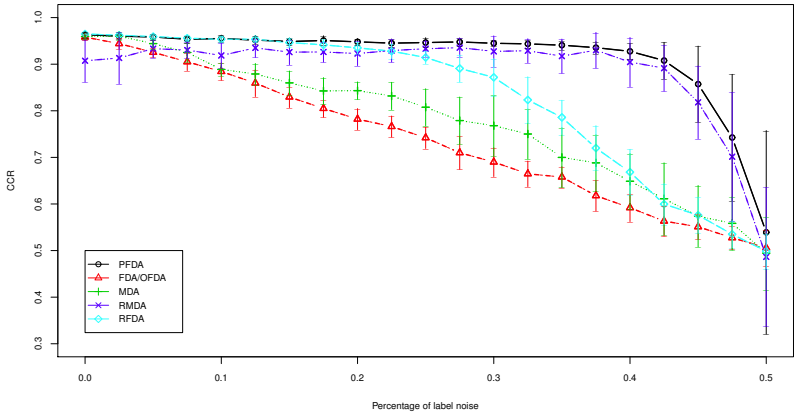


FIGURE 3.9: Comparaison des taux de classification correcte en validation de PFDA et RMDA avec ceux des meilleures méthodes robustes de l'état de l'art sur le jeu de données USPS24 (2 classes, 256 dimensions).

Taux de supervision	Méthode	Iris	Wine	Chiro	Ecoli
$\gamma = 0.2$	PFDA	94.5±4.6	78.8±9.0	77.8±15.9	97.7±2.1
	SELF	91.0±10.3	91.7±9.1	97.2±6.9	96.9±1.9
	FDA	51.7±24.4	80.6±27.4	72.2±12.6	75.1±32.1
$\gamma = 0.4$	PFDA	96.2±1.9	95.6±2.2	93.3±4.6	99.2±1.2
	SELF	88.8±8.9	95.2±3.6	98.4±1.8	96.9±1.7
	FDA	56.0±22.9	89.1±5.8	66.7±11.0	86.4±8.6
$\gamma = 0.8$	PFDA	97.5±1.1	97.0±1.5	97.8±1.5	99.4±0.6
	SELF	75.2±9.6	92.5±4.4	96.2±4.4	91.1±5.8
	FDA	54.4±16.8	78.7±10.6	70.8±4.8	81.6±3.5

TABLE 3.4: Taux de classification correcte (en pourcentage) et écarts-types pour différents taux de données d'apprentissage étiquetées pour PFDA, FDA et SELF dans le cas d'un bruit de label de $\tau = 40\%$.

avéré très similaire à celui de PFDA même si PFDA est apparu plus stable. En effet, RMDA étant basé sur un algorithme EM, il peut être sujet à des instabilités dues notamment à l'initialisation.

Nous avons enfin évalué dans [B13] la robustesse de PFDA au bruit de labels dans le contexte de la classification semi-supervisée. Pour cela, nous avons utilisé quatre jeux de données réelles provenant de la base UCI pour lesquels 50% des données ont été utilisé comme jeu d'apprentissage et le reste a servi à la validation. Dans chaque cas, une proportion γ du jeu d'apprentissage possédait des étiquettes et une proportion $1 - \gamma$ des données de ce jeu n'étaient pas étiquetées. Enfin, une proportion τ des données d'apprentissage labellisées ont vu leurs labels bruités selon le protocole des expériences précédentes. Dans ce cadre, nous avons comparé la performance de classification de PFDA à celle de FDA mais aussi de SELF [93] qui est une extension semi-supervisée de FDA. Le tableau 3.4 présente les taux de classification correcte de PFDA, FDA et SELF pour différents taux γ de données d'apprentissage étiquetées et dans le cas d'un bruit de label de $\tau = 40\%$. Il apparaît que PFDA est globalement plus performante que SELF et FDA aux différents niveaux de supervision considérés.

Pour conclure, nous avons proposé 2 méthodes génératives de classification supervisée robustes

au bruit de label. Les expériences menées ont montré que RMDA et PFDA ont un comportement similaire à MDA et FDA respectivement quand les labels sont certains et qu'elles sont robustes au bruit de label, et cela même pour des taux de contamination importants. Ces deux méthodes peuvent donc remplacer avantageusement MDA et FDA dans les applications où elles sont utilisées généralement. En outre, PFDA est une version probabiliste de FDA qui relâche notamment l'hypothèse d'homoscédasticité de FDA et autorise son utilisation dans le cadre semi-supervisé.

3.3 Classification supervisée avec classes non observées

Un autre problème qui n'est généralement pas pris en compte par les méthodes d'apprentissage statistique est la possibilité qu'une partie de la population ne soit pas représentée dans l'échantillon d'apprentissage. En particulier, il peut arriver qu'une ou plusieurs classes, présentes dans le jeu de test, n'étaient pas représentées dans le jeu d'apprentissage. Dans une telle situation, les méthodes classiques affecteront les nouvelles observations uniquement aux classes connues, sans vérifier si il est possible qu'elles appartiennent à des classes non observées.

Les travaux existants sur le sujet relèvent soit de la classification semi-supervisée, soit de la détection de nouveautés. Parmi les approches semi-supervisées, on peut citer les méthodes [61, 74] basées sur l'algorithme EM, les algorithmes de « co-training » [15] ou des techniques « graph-based » [51]. Cependant, toutes ces méthodes utilisent les observations non étiquetées pour améliorer l'inférence du modèle supposé et ne cherchent pas à détecter de nouveaux groupes homogènes d'observations. Les méthodes de détection de nouveautés (pour lesquelles un tour d'horizon exhaustif est proposé dans [57, 58]) visent quant à elles à détecter, parmi les observations à classer, celles qui semblent nouvelles vis-à-vis des classes connues. Les méthodes de ce type les plus récentes [89, 96] utilisent les *support vector machines* (SVM) pour différencier les nouveautés. Cependant, ces méthodes ne peuvent pas détecter des groupes homogènes de nouveautés et ne sont pas capables d'adapter le classifieur à la nouvelle situation. Pour pallier ce manque, nous avons proposé dans [B16] une méthode, baptisée *adaptive mixture discriminant analysis* (AMDA), qui combine une modélisation supervisée et non supervisée permettant ainsi de détecter des groupes de nouveautés et d'adapter le classifieur.

3.3.1 Le modèle de mélange utilisé

Nous avons considéré dans [B16] une modélisation du problème basée sur un modèle de mélange à K composantes où K est supérieur ou égal au nombre C de classes présentes dans les données d'apprentissage. On suppose donc que les données d'apprentissage et de test sont des réalisations indépendantes d'un vecteur aléatoire $X \in \mathcal{X}$ de densité :

$$f(x) = \sum_{k=1}^{K} \pi_k f_k(x; \theta_k),$$

où π_k et f_k sont respectivement la probabilité a priori et la densité conditionnelle de la $k$ème composante, $k = 1, ...K$. Le choix de la distribution conditionnelle dépendra bien entendu du contexte d'étude au travers des données. Les expérimentations numériques présentées plus loin utilisent la loi normale multivariée comme distribution conditionnelle du mélange. Il est bien entendu possible d'utiliser d'autres distributions conditionnelles en fonction du contexte et du type de données considérées. On adjoint à ce modèle la variable aléatoire $Z \in \{0, 1\}^K$ qui indique l'appartenance aux classes, *i.e.* $z_{ik} = 1$ si x_i appartient à la $k$ème classe et 0 sinon. Ainsi, les observations de l'échantillon d'apprentissage $S = \{(x_1, z_1), ...(x_n, z_n)\}$ sont telles que, $\forall i = 1, ...n$, $z_{ik} = 0$ pour tout $k > C$.

3.3.2 Approche transductive pour l'inférence

Nous avons tout d'abord proposé une approche transductive qui utilise à la fois les données d'apprentissage et les données de validation pour inférer le modèle. En effet, l'échantillon d'apprentissage S et l'échantillon de test $S^* = \{x_1^*, ..., x_{n^*}^*\}$ étant supposés provenir de la même population, il est naturel de vouloir utiliser les deux échantillons pour l'inférence si cela est possible. Ce contexte serait exactement celui de la classification semi-supervisée si $K = C$ mais nous considérons ici que $K \geq C$. La log-vraisemblance complétée de notre modèle s'écrit :

$$\ell(S, S^*; \Theta) = \sum_{i=1}^{n}\sum_{k=1}^{C} z_{ik} \log\left(\pi_k f_k(x_i; \theta_k)\right) + \sum_{i=1}^{n^*}\sum_{k=1}^{K} z_{ik}^* \log\left(\pi_k f_k(x_i^*; \theta_k)\right),$$

où Θ regroupe l'ensemble des paramètres du modèle. Les données de test n'étant pas complètes, *i.e.* les z_{ik}^* ne sont pas connus, il n'est pas possible de maximiser directement cette vraisemblance. Nous avons donc proposé un algorithme EM contraint qui réalise l'inférence du modèle avec les deux échantillons conjointement.

Proposition 10. *La procédure transductive d'estimation alterne les étapes E et M suivantes, à l'itération q :*

- *étape E (contrainte) : les probabilités a posteriori $t_{ik}^{(q)}$ sont contraintes à être égales à z_{ik} pour les observations d'apprentissage alors que les probabilités $t_{ik}^{*(q)}$ sont calculées pour les observations non étiquetées (de test) de la façon suivante :*

$$t_{ik}^{*(q)} = \frac{\hat{\pi}_k^{(q-1)} f_k(x_i^*; \hat{\theta}_k^{(q-1)})}{f(x; \hat{\Theta}^{(q-1)})},\ i = 1, ..., n^*,\ k = 1, ..., K.$$

- *étape M : les paramètres des K composantes du mélange sont estimées par maximisation de l'espérance conditionnelle de la log-vraisemblance complétée, ce qui conduit aux estimateurs suivants dans le cas gaussien :*

$$\hat{\pi}_k^{(q)} = \frac{n_k^{(q)} + n_k^{*(q)}}{n + n^*}, \qquad \hat{\mu}_k^{(q)} = \frac{1}{n_k^{(q)} + n_k^{*(q)}}\left(\sum_{i=1}^{n} z_{ik} x_i + \sum_{i=1}^{n^*} t_{ik}^{*(q)} x_i^*\right),$$
$$\hat{\Sigma}_k^{(q)} = \frac{1}{n_k^{(q)} + n_k^{*(q)}}\left(S_k^{(q)} + S_k^{*(q)}\right).$$

où $S_k^{(q)} = \sum_{i=1}^{n} z_{ik}(x_i - \hat{\mu}_k^{(q)})^t(x_i - \hat{\mu}_k^{(q)})$, $S_k^{*(q)} = \sum_{i=1}^{n^*} t_{ik}^{*(q)}(x_i^* - \hat{\mu}_k^{(q)})^t(x_i^* - \hat{\mu}_k^{(q)})$, $n_k^{(q)} = \sum_{i=1}^{n} z_{ik}$ et $n_k^{*(q)} = \sum_{i=1}^{n^*} t_{ik}^{*(q)}$.

La démonstration de ces résultats est donnée en annexe de [B16]. La classification des données de test non étiquetées peut être ensuite déduite des probabilités a posteriori t_{ik}^* en utilisant la règle du MAP.

3.3.3 Approche inductive pour l'inférence

Au contraire de l'approche transductive, qui suppose d'avoir les données d'apprentissage et de test simultanément, l'approche inductive ne requiert pas de conserver les données d'apprentissage pour classer de nouvelles observations. Cette approche paraît de ce fait plus défendable dans le cas de données de grandes tailles ou de la classification « online ». Dans un tel cas, il est nécessaire de procéder en deux phases : une phase d'apprentissage et une phase de découverte. La phase d'apprentissage est alors tout à fait classique et consiste en l'estimation des paramètres de C

classes représentées dans l'échantillon S (*cf.* [62] pour un point de vue complet à ce sujet). La phase de découverte estime les paramètres des $K-C$ classes non observées puis classe les données non étiquetées. Dans ce cas, la log-vraisemblance complétée a l'expression suivante :

$$\ell(\mathcal{X}^*;\Theta) = \sum_{i=1}^{n^*}\left(\sum_{k=1}^{C} z_{ik}^* \log\left(\pi_k f_k(x_i^*;\theta_k)\right) + \sum_{k=C+1}^{K} z_{ik}^* \log\left(\pi_k f_k(x_i^*;\theta_k)\right)\right),$$

où les paramètres θ_k pour $k = 1,...,C$ ont été estimés dans la phase d'apprentissage et les paramètres θ_k pour $k = C+1,...,K$ restent à estimer. Du fait de la contrainte $\sum_{k=1}^{K}\pi_k = 1$, les proportions des C classes connues devront toutefois être renormalisées en fonction de celles de $K-C$ nouvelles classes qui seront estimées sur l'échantillon S^*. L'échantillon S^* étant incomplet, *i.e.* les z_{ik}^* ne sont pas connus, il est nécessaire à nouveau d'utiliser un algorithme EM contraint pour maximiser la log-vraisemblance ci-dessus. La proposition suivante détaille cet algorithme.

Proposition 11. *La procédure inductive d'estimation alterne les étapes E et M suivantes, à l'itération q :*

- *étape E : les probabilités a posteriori $t_{ik}^{*(q)}$, pour $i = 1,...,n^*$ et $k = 1,...,K$, sont mises à jour classiquement comme suit :*

$$t_{ik}^{*(q)} = \frac{\hat{\pi}_k^{(q-1)} f_k(x_i^*;\hat{\theta}_k^{(q-1)})}{f(x;\hat{\Theta}^{(q-1)})},$$

  où $\hat{\pi}_k^{(q-1)}$ and $\hat{\theta}_k^{(q-1)}$ sont les paramètres estimés à l'étape M de l'itération $(q-1)$.
- *étape M (contrainte) : les paramètres θ_k des $K-C$ classes non observées sont estimées par maximisation de l'espérance conditionnelle de la log-vraisemblance complétée alors que les paramètres des C classes connues restent fixes, excepté pour les proportions de ces classes. Cela conduit aux estimateurs suivants dans le cas gaussien :*

$$\begin{cases} \text{pour } k = 1,...,C & \hat{\pi}_k^{(q)} = \left(1 - \sum_{\ell=C+1}^{K} \frac{n_\ell^{*(q)}}{n^*}\right)\frac{n_k}{n}, \\ \text{pour } k = C+1,...,K & \hat{\pi}_k^{(q)} = \frac{n_k^{*(q)}}{n^*} \end{cases}$$

  où $n_k^{*(q)} = \sum_{i=1}^{n^*} t_{ik}^{*(q)}$ et pour $k = C+1,...,K$:

$$\hat{\mu}_k^{(q)} = \frac{1}{n_k^{*(q)}}\sum_{i=1}^{n^*} t_{ik}^{*(q)} x_i^*, \quad \hat{\Sigma}_k^{(q)} = \frac{1}{n_k^{*(q)}}\sum_{i=1}^{n^*} t_{ik}^{*(q)}(x_i^* - \hat{\mu}_k^{(q)})(x_i^* - \hat{\mu}_k^{(q)})^t.$$

La démonstration de ces résultats est de même donnée en annexe de [B16]. Dans ce cas également, la classification des données de test non étiquetées peut être déduite des probabilités a posteriori t_{ik}^* en utilisant la règle du MAP.

3.3.4 Sélection du nombre de classes et classification

Au contraire de la classification supervisée classique, le nombre total K de classes est inconnu et il est nécessaire de le déterminer. En conséquence et du fait de nos objectifs, cette tâche est donc cruciale. Nous avons proposé dans [B16] de considérer ce problème comme un problème de choix de modèle et d'utiliser un critère de vraisemblance pénalisée pour le résoudre. Dans les expérimentations numériques présentées au paragraphe suivant, les avantages et les limites de l'usage des critères AIC [2], BIC [87] et ICL [14] dans ce contexte sont investigués.

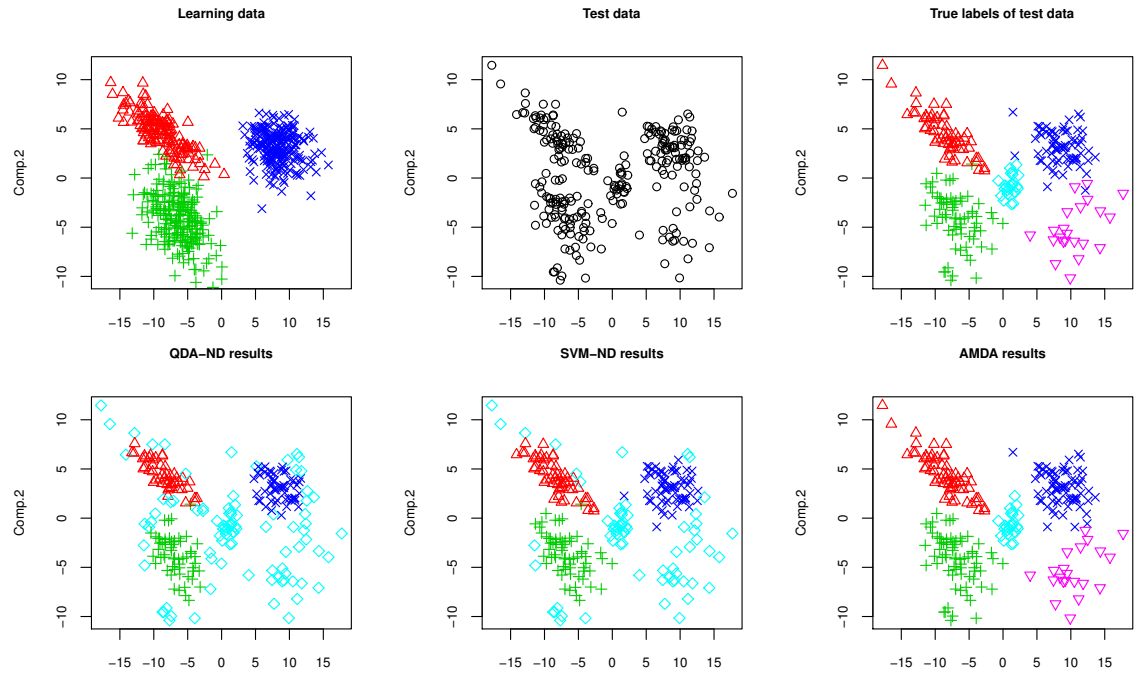


FIGURE 3.10: Détection de 2 classes non observées (losanges bleus clairs et triangles mauves) avec AMDA, QDA-ND et SVM-ND sur un jeu de données simulées.

QDA-ND						SVM-ND						AMDA					
	Truth						*Truth*						*Truth*				
Classif.	1	2	3	4	5	*Classif.*	1	2	3	4	5	*Classif.*	1	2	3	4	5
1	48					1	55					1	61	1			
2		56				2		64				2		71			
3			56			3			65			3			69		2
4	13	16	15	24	22	4	6	8	6	24	22	4			1	24	
5						5						5			1		20
Correct classif. rate = 0.74						*Correct classif. rate = 0.83*						*Correct classif. rate = 0.98*					

TABLE 3.5: Matrices de confusion pour QDA-ND, SVM-ND et AMDA sur les données de test pour le jeu de données simulées avec 2 classes non observées (classes 4 et 5).

Une fois l'inférence des modèles et le choix du nombre de classes faits, la classification de nouvelles observations est faite en utilisant la règle du MAP. Ainsi, la méthode de classification basée sur la modélisation proposée, baptisée AMDA, peut être qualifiée d'adaptative dans le sens où les nouvelles classes sont à l'issue de l'inférence inclues dans le modèle. En effet, le classifieur associé est capable dans le futur de reconnaître des observations provenant de ces classes sans avoir à chercher à nouveau la présence de ces classes.

3.3.5 Expérimentations numériques

Nous présentons à présent quelques résultats expérimentaux obtenus dans [B16]. Nous avons dans un premier temps comparé le comportement de AMDA (approche inductive) à celui des meilleures méthodes de l'état de l'art, en l'occurence QDA-ND [96] et SVM-ND [89], sur des données 2D simulées avec 3 classes observées et 2 classes non observées. La figure 3.10 présente, en haut, les données d'apprentissage, de test et les vrais labels des données de test et, en bas, les résultats fournis par les trois méthodes étudiées. Il apparaît que QDA-ND et SVM-ND parviennent

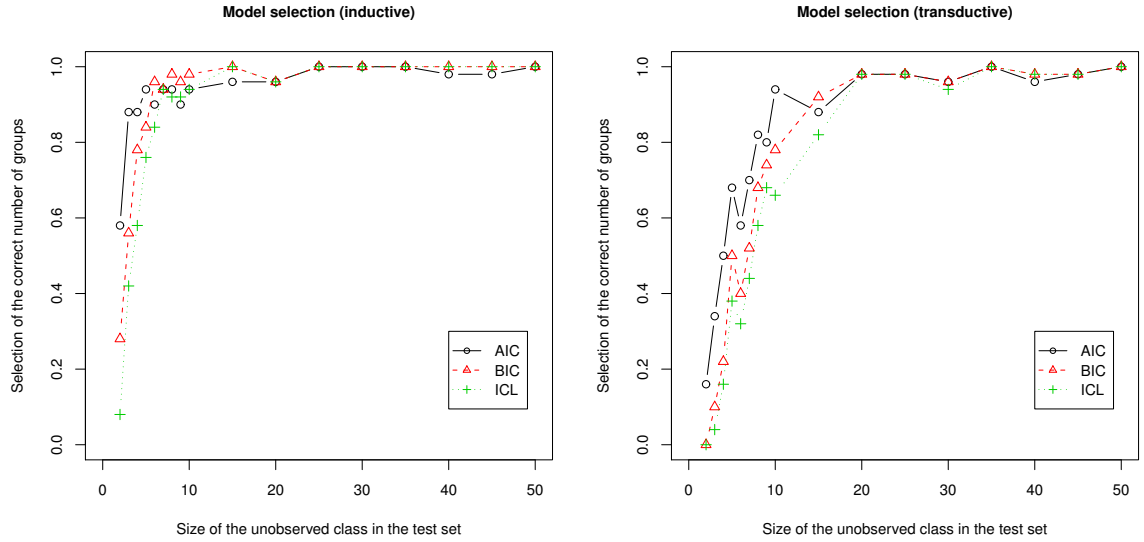


FIGURE 3.11: Taux de sélections correctes de K pour AMDA inductif (à gauche) et transductif (à droite) avec comme critère AIC, BIC ou ICL en fonction de la taille de la classe non observée.

à détecter les données provenant des 2 classes non observées, avec toutefois un nombre de fausses détections relativement important (*cf.* tableau 3.5). Cependant, QDA-ND et SVM-ND ne permettent pas d'identifier les deux groupes de nouveautés. En combinaison avec le critère AIC, AMDA parvient, quant à elle, à détecter les deux groupes de nouveautés tout en produisant un nombre de fausses détections relativement faible.

Nous avons ensuite étudié l'influence du choix du critère de sélection de modèle utilisé sur la détection du nombre correct de classes dans le jeu de test. Nous cherchions en particulier à déterminer le nombre minimum de nouveautés tel que AMDA soit capable de détecter la présence d'une nouvelle classe. Nous avons mené cette expérience à la fois dans le cas inductif et dans le cas transductif et nous avons mesuré le taux de sélections correctes de K sur 50 réplications. Les données simulées comportaient 3 classes observées et une classe non observée, *i.e.* $K = 4$. La figure 3.11 présente les taux moyens de sélections correctes de K en fonction du critère utilisé et ce dans les cas inductifs et transductifs. Dans les deux cas, les trois critères étudiés ont des comportements similaires même si AIC semble légèrement plus robuste que BIC et ICL. L'approche inductive s'avère cependant être globalement plus robuste que l'approche transductive à la taille de la classe non observée. Cette expérience nous permet également de conjecturer que la taille critique de la classe non observée pour sa détection par AMDA, avec l'approche inductive, se situe autour de $n_K = 5$. Ce résultat est particulièrement encourageant quant aux perspectives d'application d'AMDA à des problèmes réels.

Nous avons également appliqué AMDA avec succès à la détection de communautés non observées dans un réseau d'étudiants d'écoles américaines. Nous présenterons cependant ces résultats au paragraphe 4.1 qui est dévolu à l'analyse des réseaux. Pour résumer, nous avons donc proposé deux approches basées sur un modèle de mélange qui permettent d'une part la détection dans le jeu de test de classes non observées et d'autre part l'adaptation du classifieur à la nouvelle modélisation. La comparaison aux meilleures méthodes de l'état de l'art a mis en évidence que AMDA pallie l'incapacité de ces méthodes à reconnaître des groupes dans les nouveautés détectées. La comparaison des deux approches proposées nous pousse à recommander plutôt l'approche inductive qui s'est avérée capable de détecter des classes non observées de taille très limitée.

4

Apprentissage statistique sur données atypiques

Cette troisième thématique de recherche considère le problème de l'apprentissage statistique sur données non quantitatives. Cette thématique sera développée ci-dessous en trois sous-axes. Le premier est dédié à l'analyse statistique des réseaux qui sont des données de plus en plus fréquentes dans tous les domaines d'applications. Les travaux sur ce sujet ont donné lieu à 2 publications [18, 19] dans des conférences internationales. Le second sous-axe présente les travaux menés sur la problématique du clustering de données fonctionnelles. Un article méthodologique [B8] est issu de ces travaux. Le troisième sous-axe porte sur la problématique générale de la classification de données non quantitatives. Dans ce cadre, nous avons proposé récemment une approche générative basée sur des processus gaussiens parcimonieux et des fonctions noyaux. Ces travaux ont donné lieu à ce jour à une prépublication [B19].

4.1 Analyse statistique des réseaux

Depuis les travaux pionniers de Moreno [72], l'analyse des réseaux s'est principalement développée dans les domaines de la sociologie, d'une part, et de la théorie des graphes, d'autre part. La recherche dans ce domaine était principalement orientée dans l'analyse interprétative du phénomène sous-jacent ou sur les propriétés mathématiques du réseau en tant que graphe. L'augmentation récente des possibilités d'observation de réseaux a profondément modifié les besoins en analyse de réseaux. En effet, il est aujourd'hui possible d'observer des données de type réseau (appelées également données relationnelles) dans un très grand nombre de contextes, tels que les communications électroniques, les transports ou la biologie. En outre, les réseaux étudiés à l'heure actuelle sont souvent complexes et de grande taille, ce qui a incité les communautés statistique et informatique à contribuer à ce domaine de recherche.

Parmi les méthodes statistiques proposées dans ce contexte, la plupart se sont intéressées à la visualisation et au clustering des nœuds d'un réseau. La visualisation des réseaux est un enjeu principal car les réseaux sont souvent disponibles uniquement sous la forme d'une matrice des relations. Dans ce cadre, les méthodes statistiques [1, 43, 47] ont contribué à l'amélioration des visualisations en prenant notamment en compte une possible incertitude sur les relations observées. De même, la création de groupes homogènes (souvent appelés communautés en analyse de réseaux)

participe à la bonne compréhension du phénomène étudié au travers du réseau. Les travaux de [17], [56] et [73] ont notamment apporté des solutions intéressantes pour le clustering des nœuds d'un réseau. Plusieurs travaux ont également combiné visualisation et clustering avec des approches basées sur le modèle de mélange [43] ou sur une structure hiérarchique des groupes [81]. Cependant, dans le cadre des méthodes génératives, très peu de travaux ont porté sur la classification supervisée des nœuds d'un réseau. Dans [19], nous avons proposé une approche qui étend le modèle de Hoff, Raftery & Handcock [47] au cas supervisé et permet la classification de nouveaux nœuds non étiquetés.

4.1.1 Le modèle SLS

L'approche que nous avons proposé dans [19] se base donc sur le *latent space model* [47] qui suppose que la probabilité qu'il existe une relation entre les nœuds i et j dépend de la distance entre ces nœuds dans un espace latent de dimension p. Considérons une matrice des relations Y telle que $Y_{ij} = 1$ si les nœuds i et j sont connectés et $Y_{ij} = 0$ sinon, alors le *latent space model* suppose que :

$$\text{logit}P(Y_{ij} = 1|\theta) = \alpha - \|S_i - S_j\|\,,$$

où $\text{logit}(u) = \log(u/(1-u))$, $\theta = \{\alpha, S_1, ..., S_n\}$ est l'ensemble des paramètres du modèle, α détermine la probabilité a priori d'une connexion entre deux nœuds et $S_i \in \mathbb{R}^p$ est la position du nœud i dans l'espace latent. Ainsi, les nœuds i et j auront une probabilité d'être connectés d'autant plus forte que ces deux nœuds sont proches dans l'espace latent.

Dans [19], nous avons proposé d'introduire une information supervisée donnée par les étiquettes $\{z_1, ..., z_n\} \in \{1, ..., K\}$ des nœuds en ajoutant un terme βX_{ij} de la façon suivante :

$$\text{logit}P(Y_{ij} = 1|\theta) = \alpha - \beta X_{ij} - \|S_i - S_j\|\,,$$

où $X_{ij} = 1$ si les nœuds i et j sont dans la même classe (*i.e.* $z_i = z_j$), $X_{ij} = -1$ si i et j sont dans des classes différentes (*i.e.* $z_i \neq z_j$) et $X_{ij} = 0$ si la classe du nœud i ou j est inconnue. Remarquons que le terme βX_{ij} existait dans le modèle original [47] mais n'avait jamais été utilisé jusqu'alors ($\beta = 0$). Ainsi utilisé, le terme βX_{ij} force les nœuds d'une même classe à être proches les uns des autres dans l'espace latent et contraint les nœuds de classes différentes à être éloignés. De ce fait, l'espace latent construit prendra en compte à la fois les informations sur les liens entre les nœuds et sur les classes des nœuds. Ce modèle a été baptisé dans [19] le *supervised latent space model* (SLS). Notons toutefois que β n'est pas, à proprement parlé, considéré comme un paramètre du modèle car il contrôle l'influence de la supervision sur l'espace latent et devra être choisi sur un critère lié à la performance de classification (par validation croisée par exemple).

L'inférence du modèle SLS se résume donc à l'estimation des paramètres $\theta = \{\alpha, S_1, ..., S_n\}$, ce qui peut être fait par maximum de vraisemblance ou, comme proposé dans [43], par MCMC dans un cadre bayésien. La vraisemblance du modèle s'exprime de la façon suivante :

$$\log(L(\theta, Y)) = \sum_{i \neq j} \left[y_{ij}\eta_{ij} - \log\left(1 + \exp\left(\eta_{ij}\right)\right)\right],$$

où $\eta_{ij} = -\left(\alpha - \beta X_{ij} - \|S_i - S_j\|\right)$. Cette quantité n'étant clairement pas une fonction convexe, sa maximisation implique l'utilisation d'une procédure itérative d'optimisation (nous avons opté pour un algorithme de recuit simulé).

4.1.2 Inférence et classification : approche transductive

Dans le cas d'un réseau fixe, observé à un temps donné, contenant des nœuds étiquetés et d'autres non étiquetés, une approche transductive pour l'inférence et la classification de ce réseau

semble particulièrement appropriée. En effet, l'approche transductive utilise à la fois les nœuds étiquetés et non étiquetés pour apprendre le modèle. Cela est rendu possible ici par la nature du modèle SLS qui autorise que X_{ij} prenne la valeur 0 quand la classe du nœud i ou j est inconnue.

Phase d'apprentissage

La phase d'apprentissage a pour but, d'une part, de construire une représentation latente du réseau qui prenne en compte à la fois les relations entre les nœuds et la connaissance des classes des nœuds étiquetés et, d'autre part, d'apprendre un classifieur supervisé dans l'espace latent ainsi construit. La construction de l'espace latent se résume à l'estimation par maximisation de la vraisemblance des paramètres $\theta = \{\alpha, S_1, ..., S_n\}$, et ce pour une valeur donnée de β. L'influence des nœuds non étiquetés dans la construction de l'espace latent est bien entendu moins importante que celle des nœuds étiquetés. Toutefois, certains nœuds non étiquetés peuvent jouer un rôle important si par exemple ils ont un grand nombre de connections. Une fois les paramètres du modèle estimés, il est possible d'apprendre un classifieur supervisé dans l'espace latent à partir des nœuds étiquetés. Notre approche étant très générale, le classifieur peut être soit génératif (QDA, LDA par exemple), soit discriminitatif (SVM par exemple).

Phase de classification

Les positions latentes des nœuds non étiquetés ayant été estimées dans la phase d'apprentissage en même temps que celles des nœuds étiquetés, il suffit alors d'appliquer la règle de classification associée au classifieur supervisé pour prédire la classe des nœuds non étiquetés. Si le classifieur appris est un classifieur génératif, la règle du MAP permettra de déterminer la classe z_i du nœud associé à la position latente S_i.

4.1.3 Inférence et classification : approche inductive

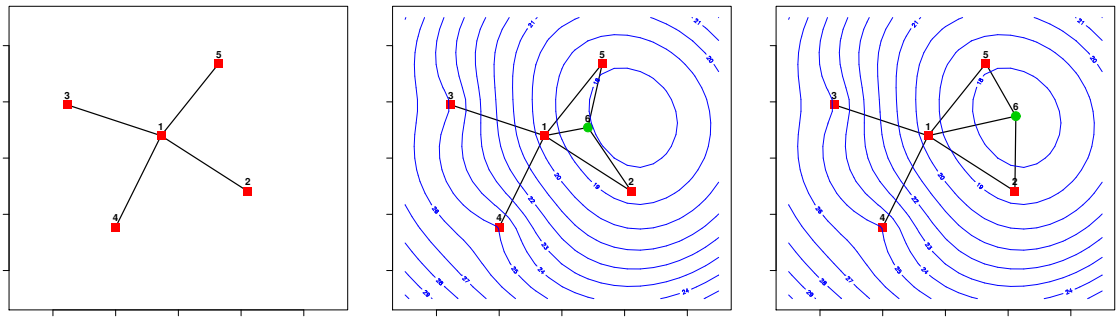
L'approche inductive suit, quant à elle, le schéma classique des méthodes de classification qui utilise les nœuds étiquetés pour apprendre le classifieur puis classe les nœuds non étiquetés a posteriori. Cette approche permet en particulier de traiter des réseaux dynamiques pour lesquels des nœuds non étiquetés viennent progressivement se joindre au réseau.

Phase d'apprentissage

Dans ce cadre, la phase d'apprentissage est tout à fait similaire à celle de l'approche transductive excepté le fait que seuls les liens et les classes des nœuds étiquetés sont utilisés pour construire l'espace latent associé au réseau. Il est à noter qu'il n'est pas recommandé d'utiliser cette approche sur des réseaux ne contenant que peu de nœuds étiquetés car la structure générale du réseau serait alors mal représentée. Une fois l'espace latent construit, le classifieur supervisé est appris de façon classique à partir des nœuds étiquetés.

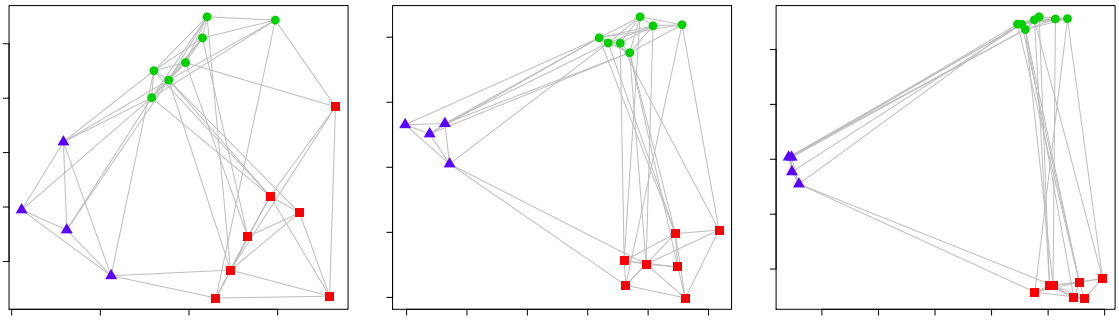
Phase de classification

Afin de pouvoir classer les nouveaux nœuds avec le classifieur supervisé appris dans la phase d'apprentissage, il est tout d'abord nécessaire de projeter les nouveaux nœuds dans l'espace latent construit précédemment. Cependant, l'espace latent associé au réseau considéré n'ayant pas de base et étant invariant aux rotations, il n'est pas possible d'obtenir directement la position latente d'un nouveau nœud à partir de la simple observation de ces liens avec les nœuds d'apprentissage. Nous avons donc proposé dans [19] de maximiser à nouveau la vraisemblance associée au réseau entier mais uniquement par rapport à la position latente du nœud à projeter, *i.e.* $\alpha, S_1, ...S_n$



Espace latent | Position latentes initiale | Position latente estimée

FIGURE 4.1: Projection d'un nouveau nœud dans l'espace latent. Les courbes de niveaux sont celles de la vraisemblance associé au réseau.



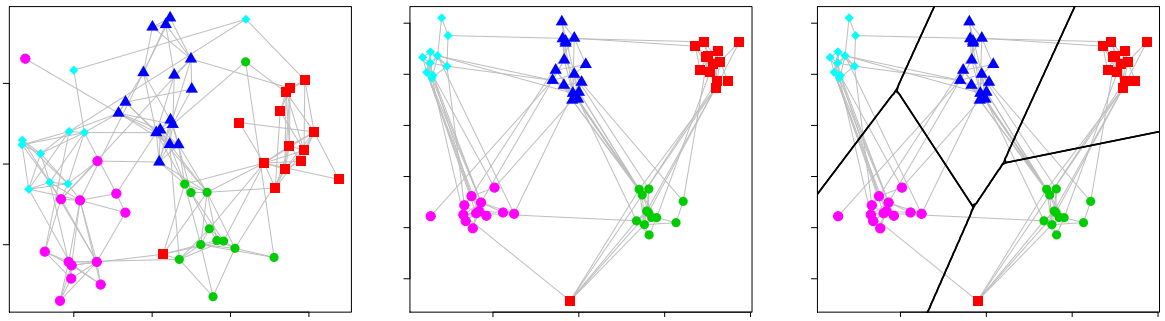
SLS avec $\beta = 0$ | SLS avec $\beta = 5$ | SLS avec $\beta = 10$

FIGURE 4.2: Représentation latente obtenue avec le modèle SLS pour le réseau des moines de Sampson.

restent fixés à leur valeur estimée dans la phase d'apprentissage et seule S_{n+1} change. La position latente S_{n+1} est initialisée à la position latente moyenne des nœuds auxquels le nouveau nœud est connecté et $X_{i,n+1}$ est fixé à 0 pour tout $i = 1, ..., n$. La figure 4.1 illustre ce processus sur un réseau jouet. Remarquons qu'une telle approche est capable de projeter simultanément plusieurs nouveaux nœuds. Une fois les positions latentes des nouveaux nœuds estimées, leur classe peut être prédite en appliquant la règle de classification associée au classifieur supervisé appris dans la phase d'apprentissage.

4.1.4 Expérimentations numériques

Nous avons tout d'abord appliqué l'approche transductive au réseau des moines de Sampson [83] afin d'observer l'effet de la supervision sur l'espace latent construit. Ce réseau social, très populaire dans le domaine de l'analyse de réseaux, a été collecté par Sampson dans un monastère de Nouvelle Angleterre (USA) et représente les liens d'amitiés entre 18 moines. En plus des liens entre les moines, Sampson a déduit de ses observations une classification en trois groupes : *loyal opposition* (7 moines), *young Turks* (7 moines) et *outcasts* (4 moines). Nous avons donc construit l'espace latent associé à ce réseau selon le modèle SLS avec $\beta = 0$ (ce qui correspond au modèle de [43]), puis $\beta = 5$ et 10. La figure 4.2 montre les représentations latentes bi-dimensionnelles obtenues. On remarque que, même si la représentation obtenue sans supervision est claire, la prise en compte de l'information supervisée facilite plus encore la visualisation et l'interprétation du réseau. Notons toutefois qu'une valeur trop élevée de β ($\beta = 10$ par exemple) a tendance à être contre-productif



Espace latent de [47] | Espace latent supervisé ($\beta = 3$) | Frontières de décision de LDA

FIGURE 4.3: Représentation latente et règle de décision du classifieur LDA pour le réseau Add-Health.

QDA-ND						SVM-ND						AMDA					
	Truth						*Truth*						*Truth*				
Classif.	1	2	3	4	5	*Classif.*	1	2	3	4	5	*Classif.*	1	2	3	4	5
1	2					1						1	2				
2		1				2						2		3			
3			2			3						3			2		
4		2		14	13	4	2	3	2	14	13	4				14	1
5						5						5					12
Correct classif. rate = 0.56						*Correct classif. rate = 0.41*						*Correct classif. rate = 0.97*					

TABLE 4.1: Matrices de confusion pour QDA-ND, SVM-ND et AMDA sur le jeu de validation pour le réseau Add-Health avec 2 classes non observées (classes 1 et 5).

puisque l'organisation au sein des groupes devient difficile à voir.

Nous avons ensuite appliqué cette approche à un réseau de taille plus réaliste qui contient 69 nœuds et 5 classes. Le réseau Add-Health est extrait d'une enquête nationale menée en 1994-95 aux USA sur la santé des adolescents. Nous avons considéré le réseau construit à partir des réponses de 69 élèves d'une même école qui devaient nommer leurs meilleurs amis au sein de l'école. Nous avons utilisé les niveaux scolaires (*grades* 7 à 11) comme information supervisée. La figure 4.3 montre les représentations latentes obtenues avec le *latent space model* de [47] et le modèle SLS, ainsi que les frontières de décision du classifieur LDA appris dans l'espace latent. Il apparaît ici clairement que la prise en compte de l'information supervisée pour la construction de l'espace latent facilite l'interprétation du réseau ainsi que sa classification. En outre, la visualisation obtenue avec $\beta = 3$ met en avant le caractère singulier d'un nœud qu'il était difficile de remarquer dans la visualisation de [47].

Nous avons également utilisé le modèle SLS dans [B16] pour illustrer l'intérêt de la méthode AMDA pour la détection de communautés non observées, qui est un problème important en analyse de réseaux. En effet, les réseaux étant par nature des objets évolutifs, il paraît essentiel de pouvoir détecter des évolutions majeures telles que l'apparition de nouveaux groupes au sein d'un réseau. Ce problème est particulièrement important dans les domaines de la sécurité (détection de groupes malveillants) et de la biologie (évolution d'un réseau de régulation de gènes). A titre d'illustration, nous avons à nouveau considéré le réseau Add-Health dans l'espace latent construit avec le modèle SLS pour $\beta = 1$. Nous avons ensuite divisé les données en un jeu d'apprentissage (35 nœuds) et un jeu de test (34 nœuds), ce dernier contenant tous les nœuds appartenant aux classes 1 et 5 (*cf.* figure 4.4), et utilisé les méthodes QDA-ND, SVM-ND et AMDA pour détecter de nouvelles classes. Le tableau 4.1 présente les matrices de confusion pour QDA-ND, SVM-ND et AMDA sur le jeu de validation. Il apparaît que SVM-ND parvient à détecter les nouvelles classes mais sans être

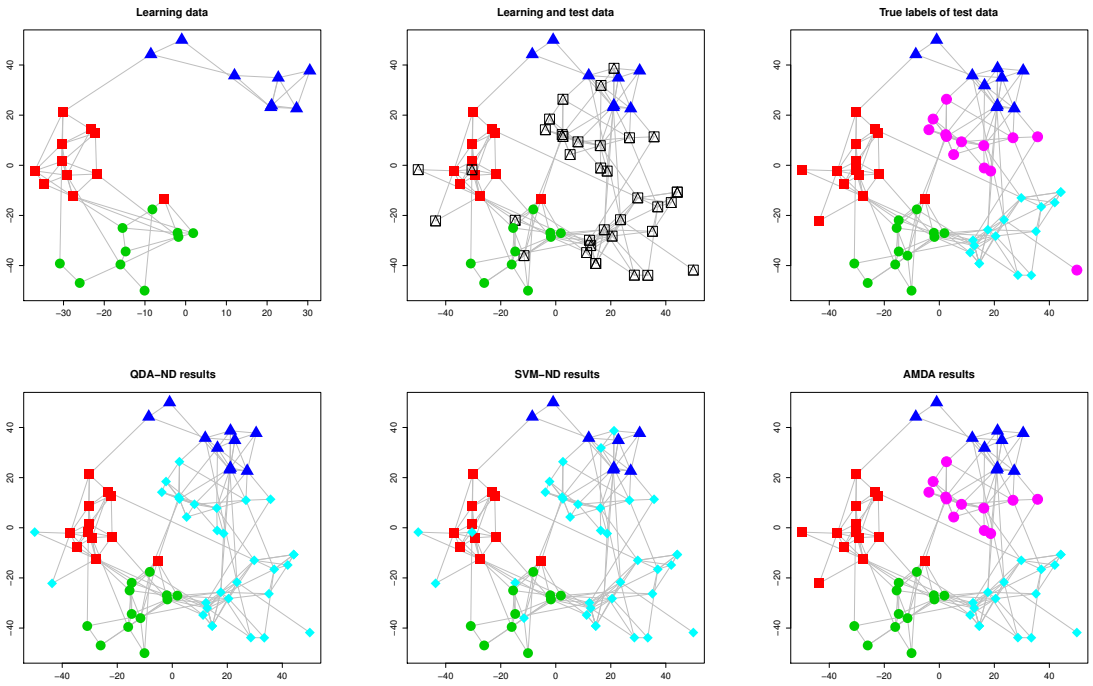


FIGURE 4.4: Détection de 2 classes non observées avec QDA-ND, SVM-ND et AMDA sur le réseau Add-Health.

capable de les discriminer et en classant à tort les autres nœuds non étiquetés comme nouveautés. QDA-ND produit de meilleurs résultats puisqu'elle détecte correctement les nouveautés et classe de façon satisfaisante les autres nœuds mais échoue également à détecter deux groupes au sein des nouveautés. Enfin, AMDA s'avère à nouveau être capable de pallier les défauts des méthodes concurrentes et détecte correctement les deux nouvelles communautés.

Le modèle SLS que nous avons proposé dans [19] est donc une extension du *latent space model* au cas supervisé. Il permet en particulier de construire une représentation latente d'un réseau qui prend en compte une information de classe sur les nœuds. Il est ensuite possible d'apprendre dans cet espace un classifieur supervisé qui sera ensuite utilisé pour la classification de nouveaux nœuds non étiquetés. Le modèle SLS a également été utilisé avec succès en combinaison avec la méthode AMDA pour la détection de communautés non observées.

4.2 Clustering de données fonctionnelles

Dans [B8], nous avons considéré le problème de la classification non supervisée de données fonctionnelles. Il s'agit d'un problème important car il est très fréquent d'observer des données fonctionnelles et, faute de méthodes adaptées, celles-ci sont souvent discrétisées et transformées en données quantitatives pour être classées. En particulier, les méthodes génératives de clustering ne peuvent pas être appliquées directement aux données fonctionnelles puisque la notion de densité de probabilité n'est généralement pas définie [28] pour ce type de variable.

Pour contourner ce problème, la plupart des méthodes existantes procèdent en deux étapes. La première étape consiste à transformer les fonctions observées en données quantitatives. Cela peut être fait en discrétisant par intervalle de temps, en décomposant sur une base de fonctions ou en considérant les scores d'une ACP fonctionnelle [80]. Une fois les données transformées, la seconde étape se résume en l'application d'une méthode classique de clustering. De telles

approches présentent toutefois l'inconvénient de sélectionner la méthode de discrétisation et la méthode de clustering de façon indépendante, ce qui peut s'avérer sous-optimal. Récemment, James et Sugar [49] ont proposé une méthode qui autorise des interactions entre les deux étapes en introduisant un modèle aléatoire pour les coefficients de la base de fonctions utilisée. Dans un même esprit, nous avons proposé dans [B8] de modéliser et de classer les fonctions observées dans des sous-espaces fonctionnels ce qui permet de prendre en compte la nature des données lors de la classification.

4.2.1 Le modèle fonctionnel latent

Considérons un échantillon de n fonctions $\{x_1, ..., x_n\}$, supposées être des réalisations indépendantes d'un processus aléatoire $X = \{X(t)\}_{t\in[0,T]} \in L_2[0,T]$, que l'on souhaite regrouper en K groupes homogènes. Les courbes $\{x_1, ..., x_n\}$ étant uniquement observées en un nombre fini d'instants, *i.e.* $x_{ij} = x_i(t_{ij})$ pour $\{t_{ij} : j = 1, \ldots, m_i\}$, il est dans un premier temps nécessaire de reconstruire la forme fonctionnelle des courbes observées. Une façon classique de faire cela est de décomposer les courbes observées sur une base de fonctions (splines par exemple, *cf.* [80]). Considérons donc une telle base $\{\psi_1, \ldots, \psi_p\}$ et supposons que X admet la décomposition suivante :

$$X(t) = \sum_{j=1}^{p} \gamma_j(X)\psi_j(t), \tag{4.1}$$

où $\gamma = (\gamma_1(X), ..., \gamma_p(X)) \in \mathbb{R}^p$ est un vecteur aléatoire et où le nombre p de fonctions de la base est supposé connu. Ainsi, les courbes observées $\{x_1, ..., x_n\}$ peuvent être représentées par leurs décompositions $\{\gamma_1, ..., \gamma_n\} \in \mathbb{R}^p$ sur la base $\{\psi_1, \ldots, \psi_p\}$ tel que $x_i(t) = \sum_{j=1}^{p} \gamma_{ij}\psi_j(t)$.

Considérons à présent uniquement les n_k courbes appartenant au $k$ème groupe, $k = 1, ..., K$, et décrites par leurs décompositions $\{\gamma_1, ..., \gamma_{n_k}\} \in \mathbb{R}^p$. Supposons tout d'abord que les $\{\gamma_1, ..., \gamma_{n_k}\}$ sont les réalisations indépendantes d'un vecteur aléatoire $\Gamma \in \mathbb{R}^p$. Supposons d'autre part que le processus stochastique associé au groupe k peut être décrit sans perte de généralité dans un sous-espace fonctionnel $\mathbb{E}_k[0,T] \in L_2[0,T]$ de dimension $d_k \leq p$ et que $\mathbb{E}_k[0,T]$ soit engendré par les d_k premiers éléments d'une base $\{\varphi_{kj}\}_{j=1,...,d_k}$ dans $L_2[0,T]$. Cette base, spécifique au $k$ème groupe, est liée à la base $\{\psi_j\}_{j=1,...,p}$ par une transformation linéaire $\varphi_{kj} = \sum_{\ell=1}^{p} q_{k,j\ell}\psi_\ell$ avec $Q_k = (q_{k,j\ell}) = [U_k, V_k]$ et où U_k, de taille $p \times d_k$, est telle que $U_k^t U_k = I_{d_k}$ et V_k, de taille $p \times (p - d_k)$, est telle que $V_k^t V_k = I_{p-d_k}$ et $U_k^t V_k = 0$. Soit$\{\lambda_1, ..., \lambda_{n_k}\}$ la décomposition latente des courbes du $k$ème groupe sur la base $\{\varphi_{kj}\}_{j=1,...,d_k}$, nous supposons en outre que $\{\lambda_1, ..., \lambda_{n_k}\}$ sont des réalisations indépendantes d'un vecteur aléatoire $\Lambda \in \mathbb{R}^{d_k}$. La relation entre les bases $\{\varphi_{kj}\}_{j=1,...,d_k}$ et $\{\psi_j\}_{j=1,...,p}$ impliquent que Γ et Λ sont liés par la relation suivante dans le cas du groupe k :

$$\Gamma = U_k\Lambda + \varepsilon$$

où $\varepsilon \sim \mathcal{N}(0, \Xi_k)$ est un bruit aléatoire indépendant. Nous supposons également que $\Lambda \sim \mathcal{N}(m_k, S_k)$ où $S_k = \mathrm{diag}(a_{k1}, ..., a_{kd_k})$. Avec ces hypothèses, la distribution de Γ pour le $k$ème groupe est :

$$\Gamma \sim \mathcal{N}(\mu_k, \Sigma_k),$$

où $\mu_k = U_k m_k$ et $\Sigma_k = U_k S_k U_k^t + \Xi_k$. Nous supposons finalement que Ξ_k est telle que $\Delta_k =$

$\text{cov}(Q_k^t\Gamma) = Q_k^t\Sigma_k Q_k$ s'écrit :

$$\Delta_k = \begin{pmatrix} \boxed{\begin{matrix} a_{k1} & & 0 \\ & \ddots & \\ 0 & & a_{kd_k} \end{matrix}} & & \mathbf{0} \\ & \boxed{\begin{matrix} b_k & & 0 \\ & \ddots & \\ & & \ddots \\ 0 & & b_k \end{matrix}} & \\ \mathbf{0} & & \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} d_k \\ \left.\vphantom{\begin{matrix} a \\ a \\ a \end{matrix}}\right\} (p-d_k) \end{matrix}$$

avec $a_{kj} > b_k$ pour $j = 1, ..., d_k$.

Supposons enfin qu'il existe une variable non observée $Z = (Z_1, \ldots, Z_K) \in \{0,1\}^K$telle que $z_{ik} = 1$ si la courbe x_i appartient au $k$ème groupe et 0 sinon. Le clustering des courbes observées $\{x_1, ..., x_n\}$ se résume alors à prédire la valeur $z_i = (z_{i1}, \ldots, z_{iK})$ de Z pour chaque observation x_i. Avec les notations et hypothèses précédentes, la distribution marginale de Γ est alors :

$$p(\gamma) = \sum_{k=1}^{K} \pi_k \phi(\gamma; \mu_k, \Sigma_k), \tag{4.2}$$

où ϕ est la densité de la loi normale, $\mu_k = U_k m_k$, $\Sigma_k = Q_k \Delta_k Q_k^t$ et $\pi_k = P(Z_k = 1)$.

Ce modèle de mélange sur une base de fonctions a été baptisé dans [B8] le *functional latent mixture* (FLM) *model*. Les hypothèses faites et la forme supposée des matrices de covariance Δ_k rappellent bien évidemment le modèle de mélange $[a_{kj}b_kQ_kd_k]$ présenté au paragraphe 2.1. La différence entre les deux modèles réside en la nature des bases dans lesquelles sont représentées les données modélisées : la base canonique de $\mathbb{R}^p$ pour l'un et la base de fonctions $\{\psi_1, \ldots, \psi_p\}$ pour l'autre.

4.2.2 Inférence et classification

Étant donné le fort lien existant entre les modèles FLM et $[a_{kj}b_kQ_kd_k]$, il a été possible d'adapter l'algorithme HDDC, permettant l'inférence et le clustering avec le modèle HD-GMM, au cas fonctionnel. L'algorithme résultant, baptisé funHDDC, est détaillé par la proposition suivante.

Proposition 12. *Dans le cas du modèle FLM, l'algorithme EM prend la forme suivante, à l'itération q :*

- *étape E : les probabilités a posteriori $t_{ik}^{(q)}$, pour $i = 1, ..., n$ et $k = 1, ..., K$, sont mises à jour classiquement comme suit :*

$$t_{ik}^{(q)} = \frac{\hat{\pi}_k^{(q-1)} f_k(x_i; \hat{\theta}_k^{(q-1)})}{f(x; \hat{\Theta}^{(q-1)})},$$

  où $\hat{\pi}_k^{(q-1)}$ and $\hat{\theta}_k^{(q-1)}$ sont les paramètres estimés à l'étape M de l'itération $(q-1)$.
- *étape M : les paramètres θ_k sont estimés par maximisation de l'espérance conditionnelle de la log-vraisemblance complétée, ce qui conduit aux estimateurs suivants :*
  - *les proportions et moyennes des groupes sont estimées respectivement par $\pi_k^{(q)} = \frac{n_k^{(q)}}{n}$ et $\mu_k^{(q)} = \frac{1}{n_k^{(q)}} \sum_{i=1}^n t_{ik}^{(q)} \gamma_i$ où $n_k^{(q)} = \sum_{i=1}^n t_{ik}^{(q)}$.*
  - *les d_k premières colonnes de Q_k sont estimées par les vecteurs propres associés aux d_k plus grandes valeurs propres de $W^{\frac{1}{2}} S_k^{(q)} W^{\frac{1}{2}}$ où $S_k^{(q)} = \frac{1}{n_k^{(q)}} \sum_{i=1}^n t_{ik}^{(q)} (\gamma_i - \mu_k^{(q)})^t (\gamma_i - \mu_k^{(q)})$ et $W = (w_{jk})_{1 \leq j,k \leq p} = \int_0^T \psi_j(t)\psi_k(t)dt$.*

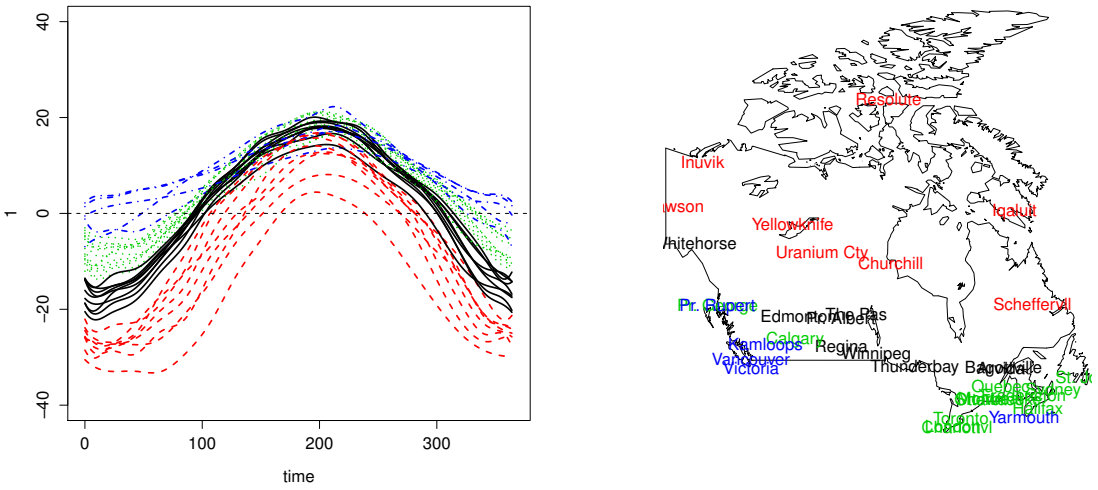


FIGURE 4.5: Clustering des 35 courbes de température avec funHDDC (modèle $[a_{kj}b_kQ_kd_k]$) et répartition géographique des stations de chaque groupe (les appartenances aux groupes sont indiquées par les couleurs).

- *les paramètres de variance* a_{kj}, $j = 1, ..., d_k$, *sont estimés par les* d_k *plus grandes valeurs propres de* $W^{\frac{1}{2}}S_k^{(q)}W^{\frac{1}{2}}$,
- *les paramètres de variance* b_k *sont estimés par* $b_k^{(q)} = \text{trace}(W^{\frac{1}{2}}S_k^{(q)}W^{\frac{1}{2}}) - \sum_{j=1}^{d_k}\hat{a}_{kj}^{(q)}$.

La démonstration de ces résultats est donnée dans [B8]. On remarque donc que la principale différence entre HDDC et funHDDC réside en le remplacement de la métrique usuelle de $\mathbb{R}^p$ par la métrique W induite par le choix de la base de fonctions. D'un point de vue pratique, HDDC pouvait être vu comme modélisant et classant les données de grande dimension par l'intermédiaire de leurs projections sur les axes d'une ACP par groupe. De façon similaire, funHDDC peut donc être vu comme modélisant et classant les courbes observées grâce à leurs projections sur les axes d'une ACP fonctionnelle par groupe. Le choix de K et des dimensions d_k est fait de la même façon qu'en HDDC.

4.2.3 Expérimentations numériques

Nous nous sommes tout d'abord intéressé dans [B8] aux intérêts pratiques de funHDDC et en particulier aux possibilités d'interprétation des résultats du clustering. Pour cela, nous avons considéré le jeu de données fonctionnelles des températures canadiennes (popularisées et décrites en détail par [80]) qui regroupe les observations quotidiennes de température de 35 stations météorologiques canadiennes. La figure 4.5 présente, à gauche, les 35 courbes de températures ainsi que le résultat de leur clustering par funHDDC en 4 groupes (choisit par BIC). Sur la même figure, à droite, on peut observer la répartition géographique des stations en fonction de la partition obtenue avec funHDDC. On remarque d'une part que funHDDC a produit des groupes homogènes de fonctions et que d'autre part ces groupes s'avèrent globalement en adéquation avec les positions géographiques des stations météorologiques (71% d'adéquation entre le clustering obtenu et la partition géographique connue pour ces données). Ainsi, le premier groupe (noir) regroupe principalement les stations continentales, le second (rouge) contient les stations arctiques, le troisième (vert) regroupe les stations atlantiques et le dernier (bleu) est formé des stations pacifiques. La figure 4.6 présente la courbe moyenne estimée des groupes 1 et 4 avec funHDDC ainsi

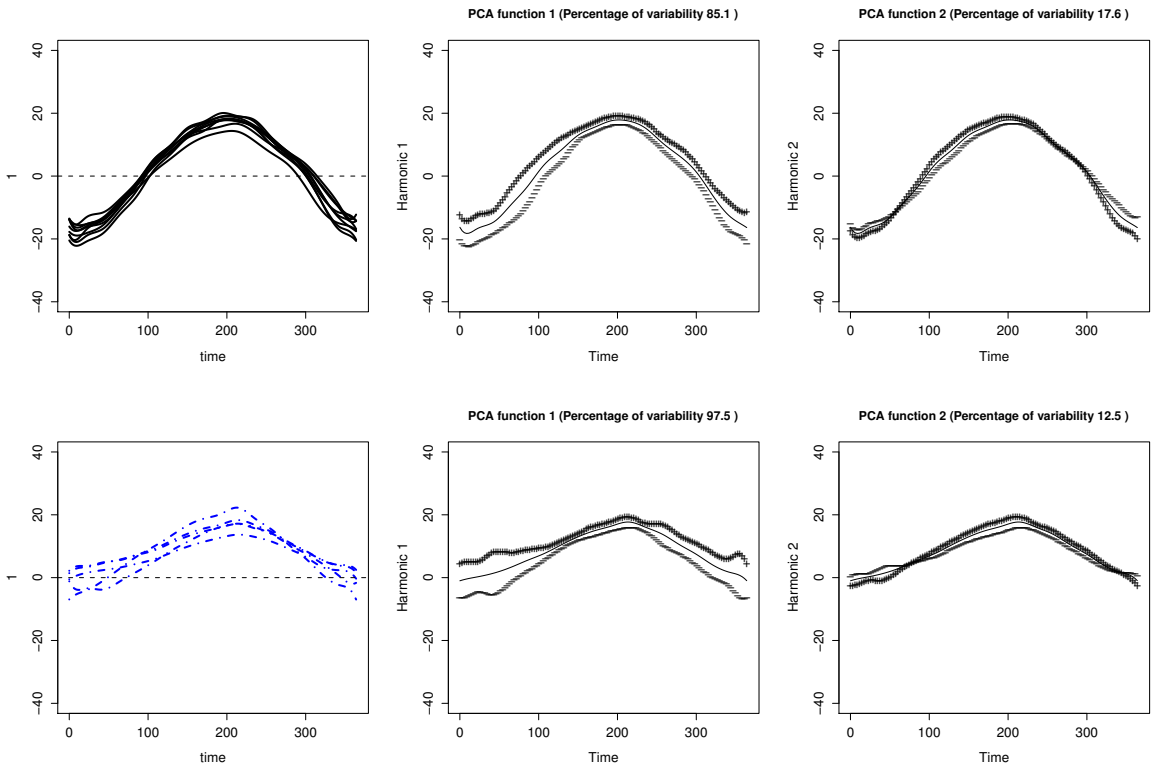


FIGURE 4.6: Courbes des groupes 1 (stations continentales, en haut) et 4 (stations pacifiques, en bas) formés par funHDDC sur les données de température (à gauche) et effet de l'ajout (+) et de la soustraction (−) de 2 écart-types sur les deux premières courbes principales (au centre et à droite).

que l'effet de l'ajout et de la soustraction de 2 écart-types sur chacune des courbes principales, ce qui autorise une interprétation fine du phénomène étudié (cf. [80] pour plus de détails). L'observation de la première fonction principale du groupe 4 révèle un phénomène spécifique qui intervient à l'entrée et à la sortie de l'hiver. En effet, on remarque une grande variance dans les données de ce groupe à ces périodes de l'année, ce qui peut être expliqué par la présence de stations de montagne non loin de l'océan. De même, la seconde fonction principale du groupe 1 présente un effet « time-shift » qui se traduit par un décalage dans le temps des courbes + et −. Cela suggère que certaines stations de ce groupe ont des saisons légèrement décalées dans le temps par rapport aux autres stations du groupe.

Nous avons également comparé les performances de clustering de funHDDC à celles des autres méthodes couramment utilisées dans ce contexte. En particulier, nous l'avons comparé à 5 méthodes en 2 étapes : HDDC, Mixt-PPCA, Mclust, k-means et hclust combinés à une discrétisation, à une décomposition sur une base de splines ou sur les fonctions principales de FPCA. Le tableau 4.2 fournit les performances de clustering de ces méthodes sur 4 jeux de données fonctionnelles provenant du *UCR Time Series Clustering website*. On peut tout d'abord remarquer qu'aucune des 6 méthodes en compétition s'est avérée significativement supérieure aux autres. En effet, presque toutes les méthodes ont fourni au moins une fois le meilleur résultat de clustering mais cela avec à chaque fois des approches de discrétisation différentes. FunHDDC s'est avéré être la meilleure méthode sur le jeu CBF et a fourni des résultats proches du meilleur résultat dans les autres cas. Nous avons remarqué en outre que funHDDC s'est avéré systématiquement meilleur que HDDC sur les scores de FPCA ce qui indique que la prise en compte de l'aspect fonctionnel des données

Données	funHDDC	Meilleur résultat de HDDC, Mixt-PPCA, Mclust, k-means et hclust		
		Discrétisation	Coefficients splines	Scores FPCA
Kneading	64.35	66.09 (HDDC)	64.35 (Mixt-PPCA)	62.61(Mixt-PPCA)
CBF	70.65	68.60 (HDDC)	62.79 (Mclust)	68.27 (Mixt-PPCA)
Face	60.71	59.82 (HDDC)	61.36 (Mixt-PPCA)	64.77 (Mixt-PPCA)
ECG	76.50	81.00 (Mclust)	80.50 (Mclust)	81.50 (Mclust)

TABLE 4.2: Performance de clustering (en pourcentage) pour funHDDC et 5 autres méthodes en 2 étapes sur 4 jeux de données fonctionnelles.

a permis d'améliorer les résultats de clustering.

Nous avons donc proposé dans [B8] une approche modélisant les données dans des sous-espaces fonctionnels et qui a donné naissance à la méthode de clustering funHDDC. Cette nouvelle méthode de clustering de données fonctionnelles s'est avérée compétitive avec les approches existantes tout en évitant le choix complexe de la discrétisation des données qui se pose dans le cas des méthodes classiques. En outre, l'analyse des sous-espaces fonctionnels estimés par funHDDC pour chaque groupe a permis des interprétations intéressantes du phénomène étudié.

4.3 Classification de données non quantitatives

Nous considérons, à présent, le problème de la classification des données non quantitatives d'un point de vue général. Comme nous l'avons vu précédemment, il est fréquent d'observer des données fonctionnelles, de type réseau mais aussi qualitatives, binaires ou même hétérogènes et, faute de méthodes de classification adaptées, ces données sont souvent transformées en données quantitatives avant d'être classées avec des méthodes standards. Dans un travail récent [B19], nous avons proposé une approche générative permettant de traiter tous les types de données pour lesquels il est possible de construire une fonction noyau.

Notre approche vise à combiner les avantages des méthodes génératives, en terme de modélisation et d'interprétation, avec la généricité et la performance des méthodes à noyau. En effet, les méthodes génératives sont très appréciées du fait de leur fondement probabiliste et des possibilités d'interprétation des résultats de classification. Cependant, ces méthodes ne s'appliquent guère qu'aux données quantitatives et qualitatives faute de distribution de probabilité adaptée aux autres types de données. Les méthodes à noyau quant à elles peuvent être appliquées à tous les types de données du moment qu'il est possible de construire une fonction noyau et sont généralement très performantes. Les résultats des méthodes à noyau sont en revanche souvent difficiles à interpréter et l'absence de fondements probabilistes peut-être préjudiciable dans certaines applications où la connaissance du risque d'erreur de classification est important.

4.3.1 Une famille de processus gaussiens parcimonieux

Considérons un échantillon d'apprentissage $\{(x_1, z_1), ..., (x_n, z_n)\}$ où $\{x_1, ..., x_n\} \in E$ sont des réalisations indépendantes d'un vecteur aléatoire X, possiblement non quantitatif et non gaussien, et où les étiquettes $\{z_1, ..., z_n\}$ sont des réalisations indépendantes d'une variable aléatoire $Z \in \{1, ..., K\}$ qui indiquent l'appartenance des observations x_i aux K classes, *i.e.* $z_i = k$ indique que x_i appartient à la $k$ème classe C_k. Supposons également qu'il existe une fonction non linéaire $\varphi : E \to F = L^2([0, 1])$ telle que $Y = \varphi(X)$ soit, conditionnellement à $Z = k$, un processus gaussien sur $[0, 1]$ de moyenne $\mu_k(t) = \mathbb{E}(Y(t)|Z = k)$ et d'opérateur de covariance

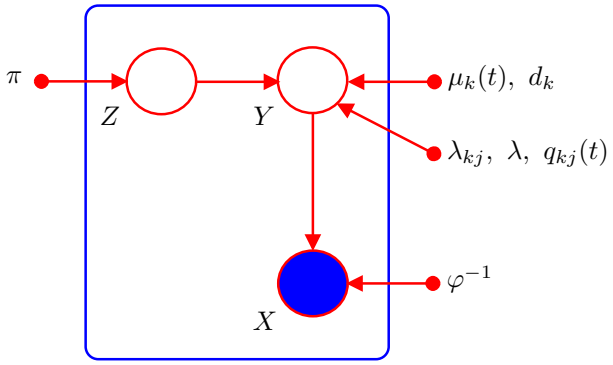


FIGURE 4.7: Représentation graphique du processus gaussien parcimonieux $\mathcal{M}_0$.

$\Sigma_k(s,t) = \mathbb{E}(Y(s)Y(t)|Z=k) - \mu_k(s)\mu_k(t)$, ce dernier admettant la décomposition spectrale :

$$\Sigma_k(s,t) = \sum_{j=1}^{\infty} \lambda_{kj} q_{kj}(s) q_{kj}(t),$$

où $\{\lambda_{kj}\}_{j\geq 1}$ et $\{q_{kj}(.)\}_{j\geq 1}$ sont respectivement les valeurs propres (ordonnées par ordre décroissant) et les vecteurs propres de Σ_k. Rappelons que les vecteurs propres $\{q_{kj}(.)\}_{j\geq 1}$ sont orthonormaux sur $F = L^2([0,1])$ pour le produit scalaire $< f, g >= \int_0^1 f(t)g(t)dt$.

Avec ces hypothèses, une idée naturelle pour classer de telles données serait d'utiliser la règle de classification du MAP qui se résume à l'estimation des fonctions de classification $D_k(x)$ pour $k = 1, ..., K$:

$$D_k(\varphi(x)) = \sum_{j=1}^{+\infty} \frac{1}{\lambda_{kj}} < \varphi(x) - \mu_k, q_{kj} >^2 + \sum_{j=1}^{+\infty} \log(\lambda_{kj}) - 2\log(\pi_k),$$

où π_k est la probabilité a priori de la $k$ème classe et affecte l'observation x à la classe associée au $D_k(\varphi(x))$ minimum. Malheureusement, les quantités $D_k(\varphi(x))$ sont évidemment difficilement estimables à partir d'un échantillon fini d'observations. Nous avons donc proposé dans [B19] de contraindre la décomposition spectrale du processus gaussien modélisant chaque classe de sorte que les fonctions de classification D_k puissent être estimées à partir d'un échantillon fini.

Définition 2. *Un processus gaussien parcimonieux est un processus gaussien Y pour lequel, conditionnellement à $Z = k$, la décomposition spectrale de son opérateur de covariance Σ_k est telle que :*

- *il existe une dimension $r < +\infty$ telle que $\lambda_{kj} = 0$ pour $j \geq r$ et pour tout $k = 1, ..., K$,*
- *et, il existe une dimension $d_k < \min\{r, n_k\}$ telle que $\lambda_{kj} = \lambda$ pour $d_k < j < r$ et pour tout $k = 1, ..., K$.*

Ces hypothèses supposent implicitement que, d'une part, les données de la classe k vivent en fait dans un sous-espace de $L^2([0,1])$ de dimension intrinsèque d_k et leur variance dans ce sous-espace est paramétré par $\lambda_{k1}, ..., \lambda_{kd_k}$, et que d'autre part la variance du bruit est modélisée par un unique paramètre λ qui est de plus commun entre les classes. Ce modèle sera appelé pgp$\mathcal{M}_0$ (ou alternativement $\mathcal{M}_0$) par la suite et la figure 4.7 en présente une représentation graphique. Il est également possible d'obtenir 8 sous-modèles ($\mathcal{M}_1$ à $\mathcal{M}_8$) en utilisant l'approche classique d'ajout de contraintes d'égalité des paramètres dans chaque classe et entre les classes. Le tableau 1 de [B19] détaille ces contraintes. Dans le cas du modèle $\mathcal{M}_0$, on a le résultat suivant.

Proposition 13. *Avec les hypothèses du modèle $\mathcal{M}_0$ et en notant $d_{\max} = \max\{d_1, ..., d_K\}$, la fonction de classification D_k s'écrit :*

$$\begin{aligned} D_k(\varphi(x)) &= \sum_{j=1}^{d_k} \left(\frac{1}{\lambda_{kj}} - \frac{1}{\lambda}\right) < \varphi(x) - \mu_k, q_{kj} >^2 + \frac{1}{\lambda}||\varphi(x) - \mu_k||^2 \\ &+ \sum_{j=1}^{d_k} \log(\lambda_{kj}) + (d_{\max} - d_k)\log(\lambda) - 2\log(\pi_k) + \gamma, \end{aligned} \tag{4.3}$$

où γ est un terme indépendant de l'indice de classe k.

La démonstration de ce résultat est donnée dans [B19]. Il est important de remarquer que l'estimation de D_k ne requiert plus, dans ce cas, que l'estimation des vecteurs propres associés aux d_k plus grandes valeurs propres de Σ_k et cela est rendu possible par l'inégalité $d_k < n_k$, $k = 1, ..., K$.

4.3.2 Inférence et classification grâce à une fonction noyau

Nous avons ensuite cherché à exploiter « l'astuce noyau » (*kernel trick*) afin d'exprimer l'estimation de la fonction de classification D_k uniquement en fonction des observations $\{(x_1, z_1), ..., (x_n, z_n)\}$ au travers d'une fonction noyau. Soit $\mathcal{K} : E \times E \to \mathbb{R}$ la fonction noyau associée à φ et définie classiquement par $\mathcal{K}(x, y) =< \varphi(x), \varphi(y) >$. En introduisant $\rho_k(x, y) =< \varphi(x) - \mu_k, \varphi(y) - \mu_k >= \mathcal{K}(x, y) - \frac{1}{n_k}\sum_{x_\ell \in C_k}(\mathcal{K}(x_\ell, y) + \mathcal{K}(x, x_\ell)) + \frac{1}{n_k^2}\sum_{x_\ell, x_{\ell'} \in C_k} \mathcal{K}(x_\ell, x_{\ell'})$, on obtient le résultat suivant.

Proposition 14. *L'estimation de D_k associée à l'échantillon $\{(x_1, z_1), ..., (x_n, z_n)\}$ peut-être obtenue, sans connaissance de φ, au travers de la fonction noyau $\mathcal{K}$ par :*

$$\begin{aligned} \hat{D}_k(\varphi(x)) = \frac{1}{n_k}\sum_{j=1}^{d_k} \frac{1}{\hat{\lambda}_{kj}} \left(\frac{1}{\hat{\lambda}_{ik}} - \frac{1}{\hat{\lambda}}\right) \left(\sum_{x_\ell \in C_k} \beta_{kj\ell}\rho_k(x, x_\ell)\right)^2 + \frac{1}{\hat{\lambda}}\rho_k(x, x) \\ + \sum_{j=1}^{d_k} \log(\hat{\lambda}_{ik}) + (d_{\max} - d_k)\log(\hat{\lambda}) - 2\log(\hat{\pi}_k), \end{aligned}$$

avec $\hat{\pi}_k = n_k/n$, $\hat{\lambda}_{kj}$ et $\beta_{kj} = (\beta_{kj\ell})_{\ell=1,...,n_k}$ sont respectivement les d_k plus grandes valeurs propres de M_k, où $(M_k)_{\ell,\ell'} = \rho_k(x_\ell, x_{\ell'})/n_k$, et leurs vecteurs propres associés, et l'estimateur de λ est $\hat{\lambda} = \sum_{i=1}^k \hat{\pi}_i(\text{trace}(M_i) - \sum_{j=1}^{d_i} \hat{\lambda}_{ij}) / \sum_{i=1}^k \hat{\pi}_i(r_i - d_i)$.

La démonstration de ce résultat est donnée dans [B19]. Il apparaît donc que la classification dans l'espace F, couramment appelé espace des caractéristiques dans la littérature des méthodes à noyaux, peut être faite sur la simple connaissance des matrices M_k qui sont définies par la fonction noyau $\mathcal{K}$, et ce sans nécessairement connaître la fonction φ.

Nous avons de plus proposé dans [B19] que l'estimation des dimensions intrinsèques d_k des classes soit basée sur la recherche d'un coude dans l'éboulis des valeurs propres de M_k, $k = 1, ..., K$, et le *scree-test* de Cattell [21] permet cela. La méthodologie proposée permet en outre de visualiser la projection des observations $\{x_1, ..., x_n\} \in E$ dans les sous-espaces de F associés à chacune des K classes. La projection $P_{kj}(x)$ d'une observation $x \in E$ sur le $j$ème axe principal de la $k$ème classe est donnée par :

$$P_{kj}(x) = \frac{1}{\sqrt{n_k \hat{\lambda}_{ik}}} \sum_{x_\ell \in C_k} \beta_{kj\ell}\rho_k(x, x_\ell).$$

Ainsi, même si les données à classer ne sont pas quantitatives, il est possible de visualiser leurs projections dans les sous-espaces de chaque classe de l'espace F des caractéristiques. La méthode de classification associée à la méthodologie présentée ci-dessus a été baptisée pgpDA.

4.3.3 Cas particuliers et extension au clustering

Nous avons également montré dans [B19] que si $E = \mathbb{R}^p$ et $\varphi(x) = x$ alors la méthode pgpDA, avec le modèle $\mathcal{M}_0$, est équivalente à la méthode HDDA [B1] avec le modèle $[a_{kj}bQ_kd]$. De même, si l'on considère des données fonctionnelles dans $E = L^2[0,1]$ et si φ est la décomposition de ces fonctions sur une base $(b_\ell)_{\ell=1,\dots,L}$de $L^2[0,1]$ telle que :

$$x(t) = \sum_{\ell=1}^{L} \varphi_\ell(x) b_\ell(t),$$

pour tout $t \in [0,1]$, alors le modèle $\mathcal{M}_0$ est équivalent au modèle FLM, proposé dans [B8] et présenté au paragraphe précédent, pour la classification non supervisée de données fonctionnelles. Ainsi, le modèle $\mathcal{M}_0$ peut être vu comme une généralisation des modèles de HDDA et funHDDC.

Nous avons en outre étendu la méthodologie présentée ci-dessus au cadre non supervisé. Dans le cadre génératif, outre l'objectif, les approches supervisées et non supervisées se différencient principalement par la procédure d'estimation des paramètres et de la règle de classification. Dans le cas non supervisé, l'échantillon $\{x_1, ..., x_n\} \in E$ à regrouper en K groupes homogènes étant incomplet (*i.e.* les étiquettes ne sont pas observées), il est nécessaire d'utiliser l'algorithme EM pour estimer les paramètres du modèles et déterminer les appartenances des observations aux groupes.

Proposition 15. *Dans le cas du modèle $\mathcal{M}_0$, l'algorithme EM prend la forme suivante, à l'étape q :*

- *étape E : les probabilités a posteriori $t_{ik}^{(q)} = \mathbb{E}(Z_i = k | x_i, \theta^{(q-1)})$ sont mises à jour selon, pour $i = 1, \dots, n$ et $k = 1, \dots, K$:*

$$t_{ik}^{(q)} = 1 / \sum_{\ell=1}^{K} \exp\left(D_k^{(q-1)}(\varphi(x_i)) - D_\ell^{(q-1)}(\varphi(x_i)) \right),$$

  où $\hat{D}_k^{(q-1)}(\varphi(x))$ est la fonction de classification estimée dans l'étape M de l'itération $q-1$.
- *étape M : la règle de classification $D_k(\varphi(x))$ est estimée conditionnellement aux probabilités a posteriori $t_{ik}^{(q)}$ comme à la proposition 14 avec $\left(M_k^{(q)}\right)_{\ell,\ell'} = \sqrt{t_{\ell k}^{(q)} t_{\ell' k}^{(q)}} \rho_k^{(q)}(x_\ell, x_{\ell'}) / n_k^{(q)}$et $\rho_k^{(q)}$ défini comme suit :*

$$\rho_i^{(q)}(x_\ell, x_{\ell'}) = \mathcal{K}(x_\ell, x_{\ell'}) - \frac{1}{n_k^{(q)}} \sum_{i=1}^{n} t_{ik}^{(q)} \left(\mathcal{K}(x_i, x_\ell) + \mathcal{K}(x_{\ell'}, x_i) \right) + \frac{1}{n_k^{(q)2}} \sum_{i,i'=1}^{n} t_{ik}^{(q)} t_{i'k}^{(q)} \mathcal{K}(x_i, x_{i'}).$$

La démonstration de ce résultat est donnée dans [B19]. Cette méthode de clustering a été baptisée pgpEM.

4.3.4 Expérimentations numériques

Nous avons dans [B19] utilisé les méthodes pgpDA et pgpEM pour classer des données de types variés et comparé leur performance à celles des meilleures méthodes de l'état de l'art. Nous avons tout d'abord vérifié que pgpDA est bien capable de classer des données quantitatives non linéaires. Pour cela, nous avons simulé des données « jouet » dans un espace de dimension 2 ayant deux classes en forme de croissants (cf. figure 4.8) et utilisé pgpDA avec un noyau gaussien (RBF). Pour cette expérience, le paramètre du noyau a été fixé à 0.5 et le seuil du *scree-test* de Cattell à 0.05. La figure 4.8 présente les frontières de décision générées par pgpDA pour chacun des 9 modèles $\mathcal{M}_0, ..., \mathcal{M}_8$. Il s'avère que 7 des modèles de pgpDA parviennent à discriminer

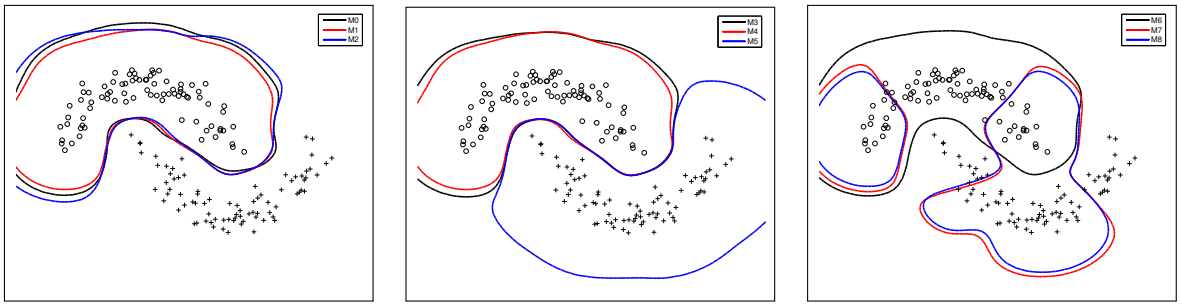
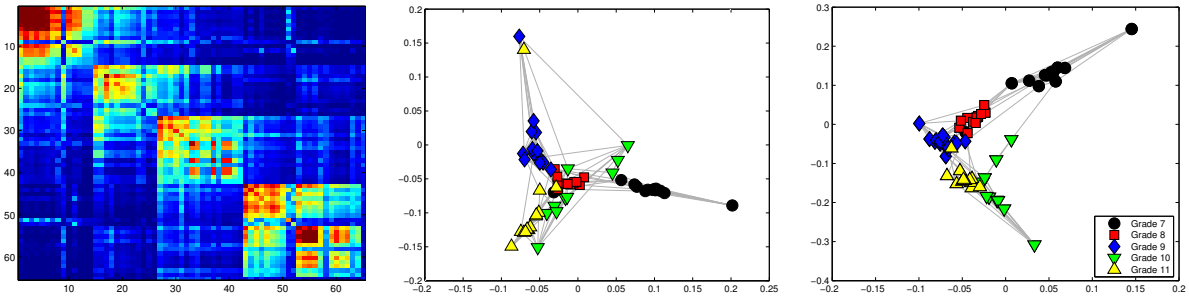


FIGURE 4.8: Frontières de décision générées par pgpDA pour chacun des 9 modèles $\mathcal{M}_0, ..., \mathcal{M}_8$ sur un jeu de données simulées.



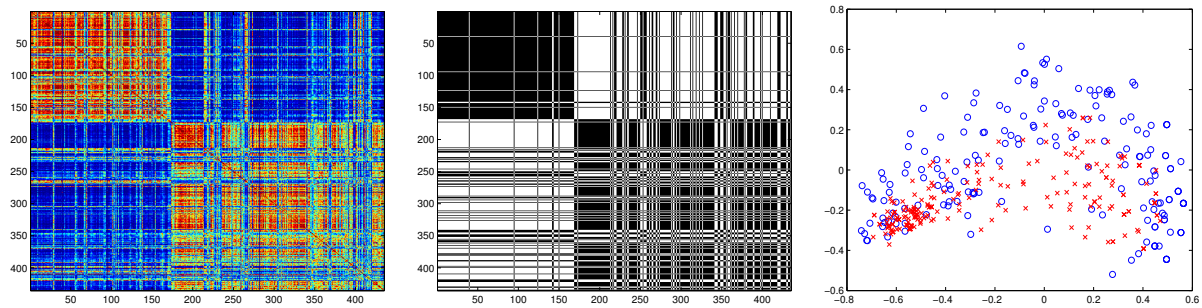
(a) Noyau (a) Sous-espace de la classe 2 (c) Sous-espace de la classe 4

FIGURE 4.9: Noyau Laplacien régularisé (à gauche) et visualisation du réseau Add-Health dans les sous-espaces des classes 2 et 4 (au centre et à droite) avec pgpDA.

parfaitement les deux classes. Seuls les modèles $\mathcal{M}_7$ et $\mathcal{M}_8$ semblent trop contraints pour pouvoir modéliser correctement les données dans l'espace des caractéristiques.

Nous avons ensuite utilisé pgpDA sur un jeu de données réelles pour la classification des nœuds d'un réseau. Le réseau considéré est le réseau Add-Health, présenté en détails au paragraphe 4.1, et nous avons utilisé le noyau du Laplacien régularisé [90]. Le paramètre du noyau a été sélectionné par validation croisée sur un jeu d'apprentissage et la valeur sélectionnée pour le paramètre a donné en moyenne 96.92% de classification correcte sur le jeu de validation. La figure 4.9 présente le noyau associé au réseau Add-Health (à gauche) ainsi que la visualisation du réseau dans les sous-espaces associés aux classes 2 et 4 (au centre et à droite). Ces deux visualisations s'avèrent très intéressantes pour analyser le réseau et interpréter sa classification. Il est à noter que la visualisation obtenue rappelle celle produite par le *latent space model* de [43]. La méthode pgpDA s'est donc avérée capable de classer les nœuds d'un réseau tout en produisant des visualisations informatives de ce réseau.

Nous avons également considéré le problème de la classification non supervisée de données qualitatives avec l'algorithme pgpEM. Les données considérées (US-House) proviennent du site UCI et contiennent les votes (*yea*, *nay* ou abstention) des membres de la chambre des représentants du congrès des USA à 16 questions majeures en 1984. A cette période, la chambre des représentants était contrôlée par les démocrates (168 républicains et 267 démocrates). Pour classer ces données, nous avons utilisé un noyau, proposé par [25] et basé sur la distance de Hamming, qui mesure le nombre minimum de substitutions pour transformer une observation en une autre. La figure 4.10 présente à gauche le noyau associé aux données. Nous avons ensuite appliqué l'algorithme pgpEM avec le modèle $\mathcal{M}_0$ et pour un nombre de groupes $K = 2$. La figure 4.10 présente, au centre, la partition des données obtenue sous la forme d'une image binaire (un pixel noir indique que les individus sont classés dans la même classe) et, à droite, la projection des données dans le



(a) Noyau (b) Partition obtenue (c) Sous-espace du groupe 2

FIGURE 4.10: Noyau basé sur la distance de Hamming (à gauche), partition obtenue avec pgpEM (au centre) et visualisation des données dans le sous-espace du groupe 2 (démocrates, croix rouges). Les résultats du clustering sont présentés sous la forme d'une matrice binaire où un pixel noir indique que les 2 individus sont dans la même classe.

Method	Iris	Glass	Wine	Ionosphere	Sonar	USPS 358	En moyenne
pgpDA	95.9±2.1	65.3±6.4	97.2±1.8	93.7±1.6	81.8±4.9	96.60±0.4	88.4±13.2
KFD	93.4±3.7	47.3±10.1	95.9±2.3	94.1±1.7	82.9±3.1	93.6±0.5	84.5±19.4
KGMM	96.6±2.3	64.5±6.3	96.6±1.7	88.1±2.3	68.9±18.1	64.7±37.5	79.9±23.3
SVM	95.7±2.0	69.1±5.5	96.8±1.4	92.8±1.8	84.8±4.0	77.6±5.4	86.1±11.7

TABLE 4.3: Taux moyen de classification correcte et écarts-type sur des jeux de validation des méthodes pgpDA, KFD, KGMM et SVM pour 6 jeux de données réelles.

sous-espace des démocrates. L'algorithme pgpEM s'est donc avéré performant pour retrouver la partition des données en 2 groupes (84.37% d'adéquation entre la partition obtenue et celle connue) tout en fournissant une visualisation de ce jeu de données qualitatives.

Nous avons enfin comparé la performance de pgpDA à celle des meilleures méthodes de l'état de l'art. Nous avons donc comparé pgpDA aux méthodes KFD [68], KGMM [30] et SVM [84] sur 6 jeux de données réelles provenant du site UCI. Pour toutes les méthodes, le noyau gaussien (RBF) a été utilisé et son paramètre a été à chaque fois sélectionné par double validation croisée sur des jeux d'apprentissage. Le tableau 4.3 présente les taux moyens de classification correcte sur les jeux de validation pour les différents jeux de données. Il apparaît tout d'abord que chacune des méthodes surpasse les autres sur au moins un jeu de données ce qui prouve le haut niveau de compétition entre les méthodes. Les méthodes pgpDA et SVM remportent chacune deux compétitions et pgpDA s'avère être en moyenne la méthode la plus performante sur les 6 jeux de données. On remarque en particulier que pgpDA surpasse SVM en grande dimension.

La méthodologie proposée dans [B19], qui a donné naissance aux méthodes pgpDA et pgpEM, s'est donc avérée performante pour classer des données de types variés tout en fournissant des représentations informatives de ces données. Il a en particulier été possible de traiter des données quantitatives non linéaires, qualitatives, de type réseau mais également des données hétérogènes (mélange de variables quantitatives et qualitatives) en combinant plusieurs noyaux.

5
Applications et logiciels

Ce chapitre présente quelques applications des méthodologies statistiques développées dans les chapitres précédents. Le premier paragraphe s'intéresse à l'application de la méthode RMDA au problème de la reconnaissance d'objets dans des images. Cette application a été publiée dans [B5]. Deux applications à l'analyse de données bio-médicales sont présentées dans le second paragraphe de ce chapitre. Ces applications ont été développées à l'occasion de l'encadrement de la thèse de Camille Brunet et ont été publiées dans [B12]. Le troisième paragraphe est dédié à l'application de la méthode HDDA à la classification de données de chimiométrie. Ces travaux ont donné lieu à une publication [B7]. La quatrième application traite de la classification non supervisée de données hyper-spectrales pour la catégorisation des sols de la planète Mars. Les résultats de cette étude ont été notamment publiés dans [B3] et [B20]. La dernière application que nous présentons est liée au problème de la détection de nouveautés en clustering online. Il s'agit d'un travail commencé récemment dans le cadre de la thèse CIFRE d'Anastasios Bellas que je co-encadre avec Marie Cottrell. Enfin, le dernier paragraphe présente brièvement les logiciels produits qui ont donné lieu en outre à une publication [B14].

5.1 Application à la reconnaissance d'objets dans images

La classification d'objets dans des images est un des problèmes les plus difficiles à l'heure actuelle en vision par ordinateur mais est également au cœur d'un grand nombre de progrès technologiques parmi lesquels on peut citer la télé-surveillance, la sécurité et le pilotage autonome de véhicules. Une approche devenue maintenant classique représente les images grâce à des descripteurs locaux qui représentent des parties importantes des images. La classification d'objets dans des images se ramène alors au problème de la classification de descripteurs locaux. Cependant, le nombre potentiellement infini d'objets à reconnaître n'autorise pas d'utiliser le schéma classique apprentissage / prédiction. En effet, la supervision de données d'apprentissage requièrerait la segmentation manuelle d'un grand nombre d'images et cela pour tous les types d'objets que l'on souhaite reconnaître. Il est donc nécessaire d'utiliser un mode de supervision, appelé faiblement supervisé, qui assigne tous les descripteurs locaux d'une image contenant un objet donné à la classe de cet objet. Ainsi, de nombreux descripteurs locaux sont mal annotés mais cela permet de réduire drastiquement le coût de la supervision. En regard de nos travaux sur le bruit de labels, ce mode de supervision peut-être vu comme une supervision générant du bruit de labels et l'on

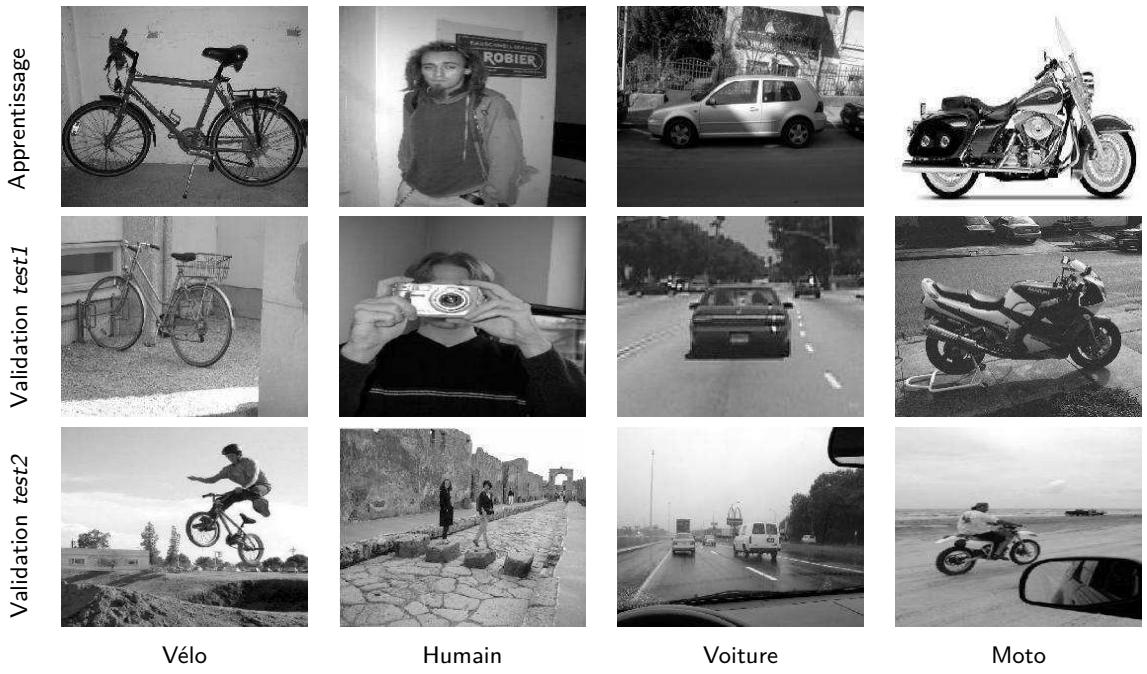


FIGURE 5.1: Echantillon d'images d'apprentissage et de validation de la base d'images Pascal .

peut donc espérer que la méthode RMDA [B5] sera capable de reconnaître correctement les objets contenus dans des images.

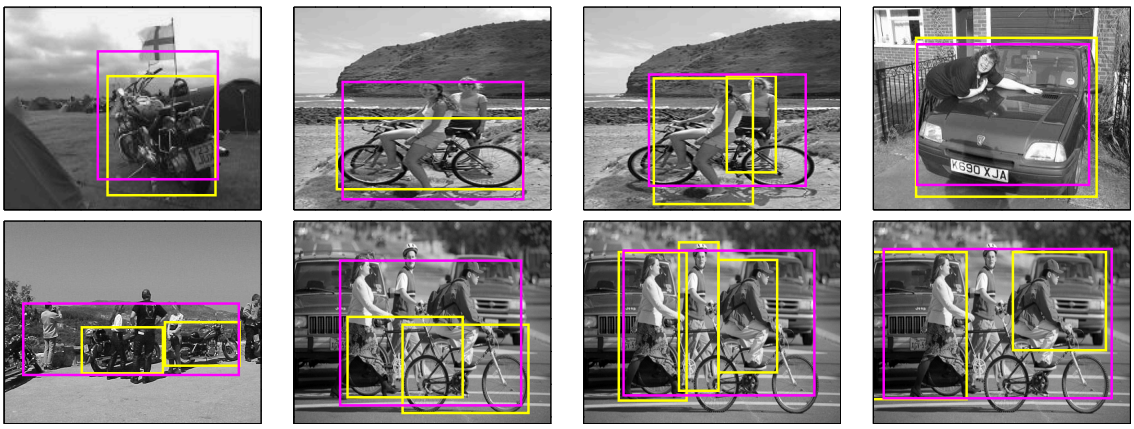
Pour vérifier cela, nous avons utilisé dans [B5] une grande base de données de classification d'objets pour laquelle les résultats de nombreuses méthodes de l'état de l'art sont disponibles. La base d'images Pascal [26] comporte quatre catégories d'objets : « moto », « vélo », « humain » et « voiture ». Elle est composée de 684 images d'apprentissage et de deux jeux de validation : le jeu *test1* qui comporte 689 images et le jeu *test2* qui en comporte 956. Les images du jeu *test1* sont du même type que les images d'apprentissage, *i.e.* les objets sont de même taille et dans des poses similaires. Par conséquent, ce jeu de validation est considéré comme un jeu « facile ». En revanche, les images du jeu de validation *test2* sont issues du moteur de recherche « Google Image » et sont par conséquent très différentes des images utilisées par l'apprentissage. La figure 5.1 met en évidence la différence de nature des images entre le jeu d'apprentissage et le jeu de validation *test2*. Nous avons utilisé sur ces différents jeux de données la méthode RMDA en combinaison avec les modèles gaussiens parcimonieux, présentés au paragraphe 2.1, car les descripteurs locaux utilisés sont en grande dimension. Les résultats de classification sont présentés au tableau 5.1 en utilisant la mesure de classification utilisée dans [26]. Il apparaît que RMDA améliore significativement les résultats de classification par rapport à la meilleure méthode connue sur ces jeux de données. Il est particulièrement intéressant de remarquer que les résultats de localisation supervisée et faiblement supervisée avec RMDA sont relativement proches. Cela est très encourageant pour le développement de la supervision faible. La figure 5.2 présente des résultats de localisation supervisée sur des images du jeu de validation *test2*.

5.2 Application au domaine bio-médical

A l'occasion de l'encadrement de la thèse de Camille Brunet, nous avons été amené à considérer deux applications médicales : la détection du cancer du col de l'utérus à partir d'images cytologiques et la détection du cancer colorectal par spectrométrie de masse. Pour des raisons de confidentialité

Jeu de validation	*Pascal test1*		*Pascal test2*	
Supervision	supervisé	faibl. sup.	supervisé	faibl. sup.
RMDA $[a_{ij}bQ_id_i]$	0.318	0.287	0.181	0.147
RMDA $[a_ib_iQ_id_i]$	0.313	0.285	0.183	0.142
RMDA $[a_ibQ_id_i]$	0.318	0.283	0.176	0.148
RMDA $[a_ib_iQ_id]$	0.314	0.287	0.179	0.130
K-means	0.261	0.204	0.160	0.099
Meilleure méthode de [26]	0.279	/	0.112	/

TABLE 5.1: Localisation supervisée et faiblement supervisée sur la base *Pascal* : les résultats présentés ici sont les moyennes sur les 4 catégories des mesures AP (voir le texte pour plus de détails).



Moto Vélo Humain Voiture

FIGURE 5.2: Localisation supervisée avec RMDA sur le jeu *test2* de la base *Pascal* : les *bounding boxes* prédites sont tracées en rouge et les *bounding boxes* réelles sont en jaune.

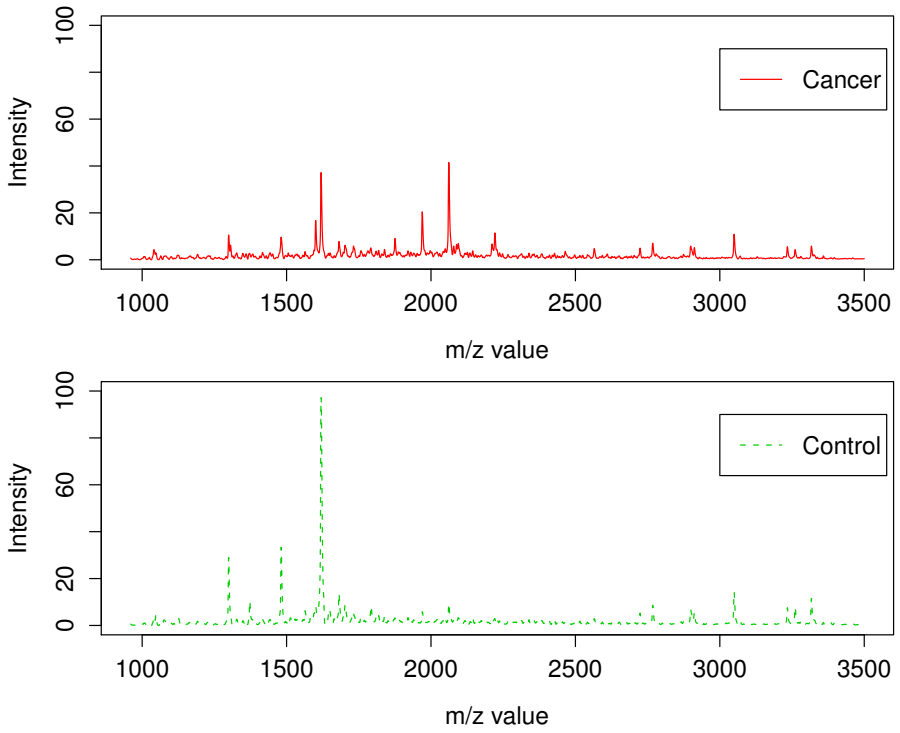


FIGURE 5.3: Spectres moyens des groupes cancer (en haut) et contrôle (en bas) sur la plage de valeurs m/z 900–3500 Da.

imposées par la société Novacyt avec laquelle nous avons travaillé et liées à l'extraction de caractéristiques discriminantes, nous ne pouvons malheureusement pas présenter les résultats de l'usage de l'algorithme Fisher-EM pour la détection précoce du cancer du col de l'utérus. Nous avons en revanche pu publier dans [B12] l'application de Fisher-EM à l'analyse de données de spectrométrie de masse pour la détection du cancer colorectal. Les données nous avaient été fournies par Théodore Alexandrov de l'université de Brème, Allemagne. Le jeu de données MALDI [3] qui a été mis à notre disposition contient 112 spectres de longueur 16 331 parmi lesquels 64 spectres sont ceux de patients atteints par un cancer colorectal. En suivant les résultats de [3], Fisher-EM a été appliqué sur les 6 168 dimensions correspondant aux rapports m/z entre 960 et 3 500 Da. La figure 5.3 présente les spectres moyens estimés des groupes cancer et contrôle. L'algorithme Fisher-EM a été utilisé pour le clustering de ces données ainsi que deux autres méthodes concurrentes (PCA-EM et Mixt-PPCA). Le tableau 5.2 présente les tables de confusion pour les trois méthodes de clustering considérées. Nous avons en outre utilisé la matrice U des *loadings* fournis par Fisher-EM pour sélectionner les variables (valeurs m/z) les plus discriminantes. La figure 5.4 présente la différence entre les spectres moyens des groupes cancer et contrôle ainsi que la sélection des variables les plus discriminantes. Comme attendu, certaines variables sélectionnées sont des variables où la différence entre les spectres des groupes cancer et contrôle sont très importantes. Cependant, Fisher-EM sélectionne également quelques variables (valeurs m/z 2800 et 3050) pour lesquelles les différences entre les spectres cancer et contrôle sont très petites. Il est intéressant de noter que cette sélection de variables a une intersection importante avec celle proposée sur des avis d'experts dans [3].

PCA-EM		
	Groupes	
Classes	Cancer	Control
Cancer	48	16
Control	1	47
Erreur de classif. = 0.15		

Mixt-PPCA		
	Groupes	
Classes	Cancer	Control
Cancer	62	2
Control	10	38
Erreur de classif. = 0.11		

Fisher-EM		
	Groupes	
Classes	Cancer	Control
Cancer	57	7
Control	3	45
Erreur de classif. = 0.09		

TABLE 5.2: Tables de confusion pour PCA-EM, mélange de PPCA et Fisher-EM pour le clustering des données de spectrométrie de masse.

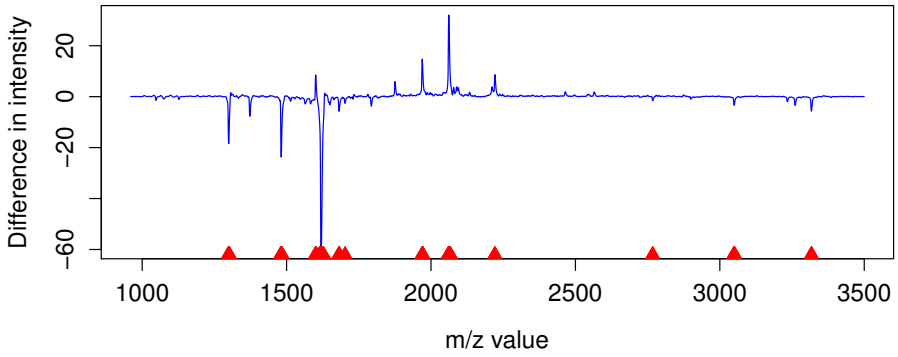


FIGURE 5.4: Différence entre les spectres moyens des groupes cancer et contrôle ainsi qu'une sélection des variables les plus discriminantes (indiquées par les triangles rouges).

5.3 Application à la chimiométrie

Dans [B7], nous avons utilisé la méthode HDDA pour la classification de données de chimiométrie. Ce travail a été mené en collaboration avec les chimiomètres du laboratoire LASIR de l'université Lille 1 qui sont fréquemment confrontés au problème de la classification de données de grande dimension. Les données qu'ils considèrent sont généralement des spectres échantillonnés à plusieurs milliers de longueurs d'ondes (2800 pour les jeux de données que l'on a considéré) pour seulement quelques centaines de spectres mesurés. Du fait de ce rapport défavorable entre nombre d'observations et nombre de variables, les données sont généralement classées avec les méthodes SVM (avec un noyau gaussien), SIMCA [102] ou PLS-DA [6].

Les données que nous avons considéré sont issues d'une étude où il s'agissait de mesurer l'effet d'une propriété physique (qu'il n'est pas possible de dévoiler pour des raisons de confidentialité) sur trois types de textiles. Le jeu de données contenait 202 spectres mesurés en proche infra-rouge sur 2800 longueur d'ondes. Les spectres moyens des trois textiles sont présentés par la figure 5.5. Le tableau 5.3 présente les résultats de classification obtenus et reporte les résultats obtenus pour la méthode SIMCA par d'autres auteurs sur le même jeu de données. Pour cette expérience, les paramètres de chaque méthode ont été réglé par validation croisée (5 *folds*). Il apparaît donc que HDDA surpasse ses concurrents à la fois en performance de classification mais également en temps de calcul pour l'apprentissage du classifieur. En outre, l'étude a posteriori des différents paramètres estimés par HDDA (tels que les dimensions intrinsèques et les matrices des loadings Q_k) a permis aux collègues chimiomètres de réaliser des interprétations fines des résultats.

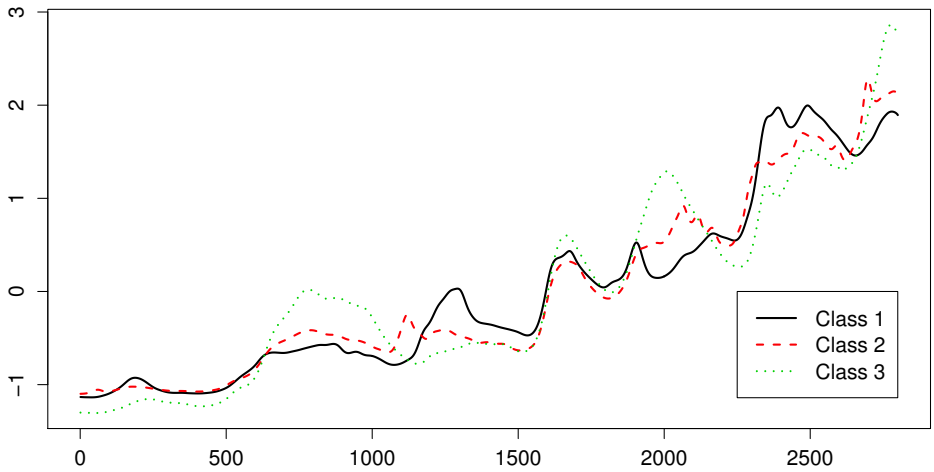


FIGURE 5.5: Spectre moyen de chacune des trois classes pour le jeu de données de chimiométrie.

Méthodes	CCR en CV	CCR en test	Temps d'apprentissage
HDDA	92.3	96.7	3 sec.
SVM	88.5	91.2	182 sec.
PLS-DA	87.7	84.7	59 sec.
SIMCA	–	82.4	–

TABLE 5.3: Taux de classification correcte (en pourcentage) en validation croisée (CV) lors de l'apprentissage et sur les données de test pour HDDA, SVM (noyau gaussien), PLS-DA et SIMCA sur les données de chimiométrie. Le temps d'apprentissage des classifieurs est également reporté.

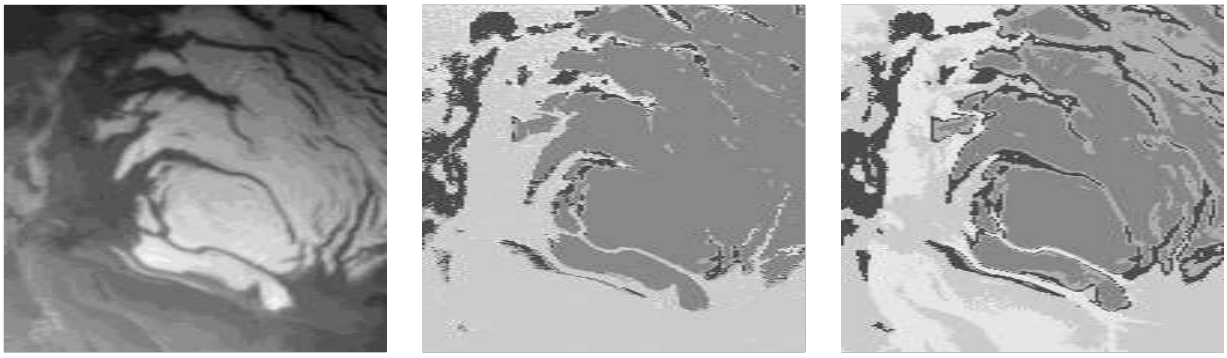


Image de la zone étudiée | Segmentation expert | Segmentation SparseFEM

FIGURE 5.6: Catégorisation du pôle sud de la planète Mars avec la méthode sparseFEM.

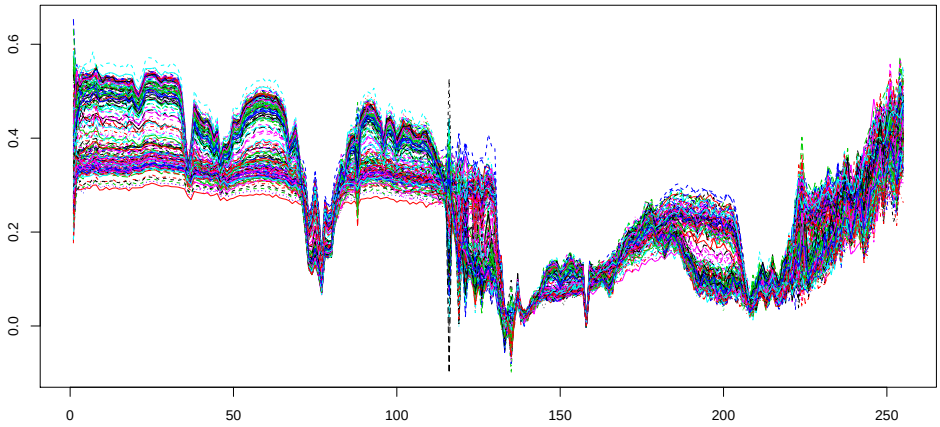


FIGURE 5.7: Quelques-uns des 38 400 spectres mesurés au pôle sud de la planète Mars et décris sur 256 longueur d'ondes.

5.4 Application à l'analyse d'images hyper-spectrales

Nous nous sommes également intéressés à plusieurs reprises, dans le cadre d'une collaboration avec le laboratoire de Planétologie de Grenoble, au problème de la classification non supervisée de données hyper-spectrales pour la catégorisation automatique des sols de la planète Mars. Les données, qui arrivent sous forme d'images hyper-spectrales, sont à la fois de grande dimension et en très grand nombre. L'imagerie hyper-spectrale visible et infrarouge est une technique de télédétection clef pour l'étude et le suivi des planètes du système solaire. Les spectromètres imageurs intégrés dans un nombre croissant de satellites génèrent des images hyper-spectrales à trois composantes (deux composantes spatiales et une spectrale). Les données mises à notre disposition par le laboratoire de Planétologie de Grenoble ont été acquises par l'imageur OMEGA [12]. Cet imageur a observé le sol de la planète Mars avec une résolution spatiale variant entre 300 et 3000 mètres en fonction de l'altitude du satellite. Il a acquis pour chaque pixel observé les spectres dont les longueurs d'ondes vont de 0.36 à 5.2 µm et stocké ces informations dans un vecteur de 256 dimensions [11]. Le but de notre étude était de caractériser la composition de la surface du sol martien en regroupant les spectres observés en 5 groupes minéralogiques.

Pour cette expérimentation, nous avons considéré une image de taille 300×128 pixels de la surface de la planète Mars dont chacun des 38 400 pixels est décrit par 256 variables. L'image de gauche de la figure 5.6 représente la zone étudiée et la figure 5.7 montre quelques-uns des

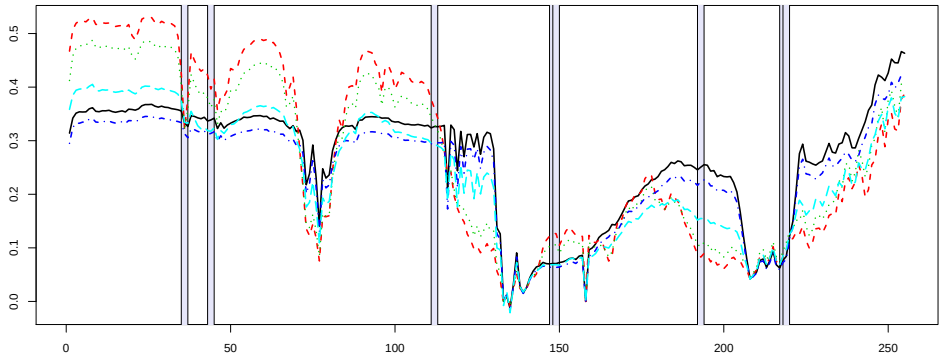


FIGURE 5.8: Spectre moyen de chacun des 5 groupes formés par sparseFEM et sélection des longueurs d'ondes discriminantes (indiquées par des bandes grises).

38 400 spectres à classer. Nous avons traité ces données à plusieurs reprises, notamment dans [B3] et [B20], où nous les avons analysé respectivement avec les méthodes HDDC et sparseFEM. L'image de droite de la figure 5.6 présente la segmentation obtenue avec la méthode sparseFEM alors que l'image du centre montre la segmentation fournie par le modèle physique utilisé au laboratoire de Planétologie de Grenoble. On peut tout d'abord observer que la segmentation fournie par sparseFEM concorde avec la partition expert sur une grande partie de l'image (60.3% d'accord entre les deux partitions). Les experts du laboratoire de Planétologie de Grenoble ont particulièrement apprécié que notre méthode soit capable de détecter le mélange de glace et de carbonate (liseré noir) présent autour des zones de glace (zones claires de l'image de gauche). La figure 5.8 présente les moyennes spectrales des 5 classes ainsi que les variables (longueur d'ondes) identifiées par sparseFEM comme étant discriminantes. A partir de ces informations, les experts peuvent, d'une part, déterminer à partir des spectres moyens la composition minéralogique de chacune des classes et, d'autre part, utiliser la sélection de variables discriminantes pour optimiser le nombre de longueurs d'ondes à mesurer lors des futures campagnes d'acquisition.

5.5 Application au *health monitoring* en aéronautique

Depuis le début de l'année 2011, je co-encadre avec Marie Cottrell la thèse CIFRE d'Anastasios Bellas. Cette thèse s'inscrit dans une collaboration avec l'industriel SNECMA qui est le premier constructeur européen de moteurs d'avion. Le sujet de la thèse d'Anastasios Bellas est la détection non supervisée de nouveautés dans des signaux vibratoires. Il s'agit de construire des méthodes statistiques capables de détecter dans un contexte non supervisé, préférablement de manière dynamique, des observations différentes des états déjà observés. Le but industriel étant d'anticiper la révision des moteurs présentant des anomalies avant qu'une panne, impliquant une immobilisation de l'avion, ne survienne.

Lors de cette première année de thèse, nous avons tout d'abord fait un état de l'art poussé sur la détection supervisée de nouveautés, le clustering robuste et le clustering « online ». Nous avons ensuite proposé une version robuste de l'algorithme HDDC, présenté au paragraphe 2.1, afin qu'il puisse détecter des anomalies parmi un ensemble de données de grande dimension. Ce travail a donné lieu à une publication [8] dans une conférence internationale. L'idée de cette robustification de HDDC est d'ajouter au sein de l'algorithme EM une étape T de « trimming » qui retire les observations jugées anormales sous le modèle courant. Cette étape T s'intercale entre les étapes E et M de l'algorithme original et, à l'itération q, retire la proportion α d'observations ayant les plus petites valeurs pour $\min_{k=1,\dots,K} D_k^{(q)}(x)$, où $D_k^{(q)}(x) = -2\log(\pi_k^{(q-1)}\phi(x;\hat{\theta}_k^{(q-1)}))$. Ainsi,

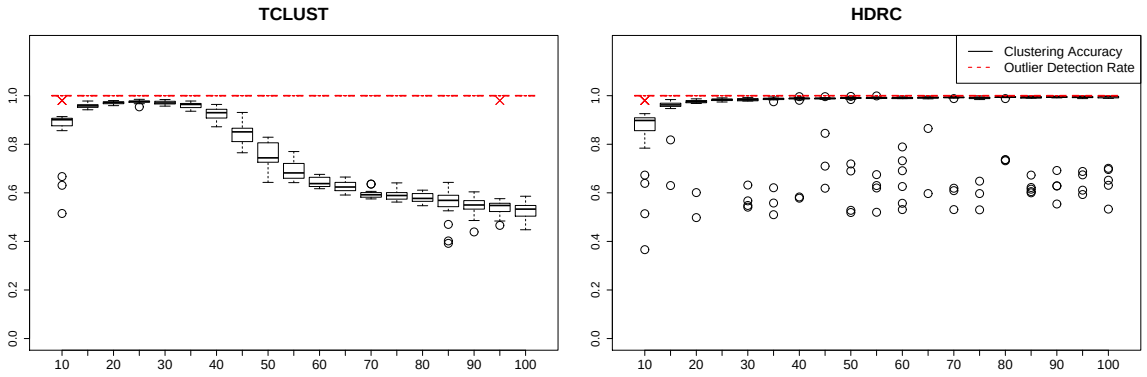


FIGURE 5.9: Taux de classification correcte (boxplots noirs) et le taux d'anomalies correctement détectées (boxplots rouges) pour HDRC et TCLUST sur données simulées (25 réplications).

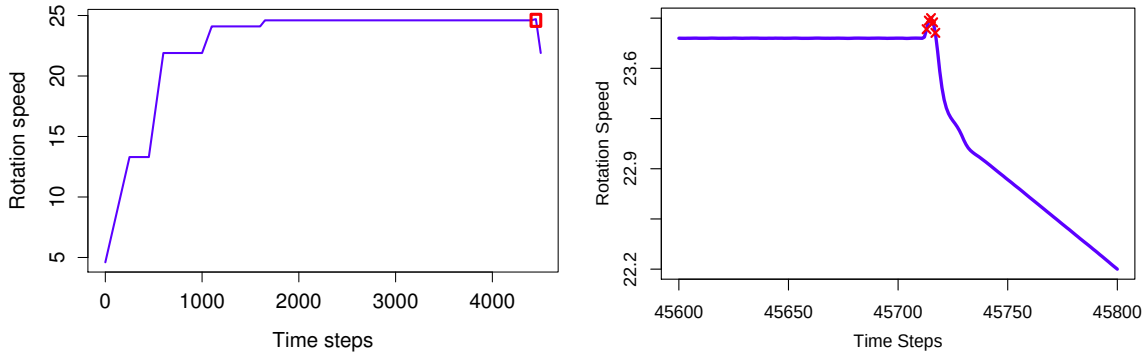


FIGURE 5.10: Vitesse de rotation du moteur en fonction du temps et anomalies détectées (croix rouges) par HDRC en utilisant les 173 variables.

l'étape M, qui suit cette nouvelle étape, effectue la mise à jour des paramètres du modèle sans les données détectées comme anomalies. Le choix du paramètre α est bien sûr un point critique de la méthode. Il n'est malheureusement pas possible d'utiliser dans ce cas les outils classiques de vraisemblance pénalisée car le nombre d'observations change en fonction de α. Nous avons donc retenu la solution empirique proposée par Garcia-Escudero *et al.* [39]. Nous avons baptisé *high-dimensional robust clustering* (HDRC) cette version robuste de HDDC.

Avant de mettre en œuvre l'algorithme HDRC sur les données de la SNECMA, nous avons vérifié sur simulations que la méthode est bien, comme attendu, robuste à la fois aux anomalies et à la dimension. Pour ce faire, nous avons simulé des données selon le modèle gaussien parcimonieux de HDDC et ajouté une proportion $\alpha = 0.05$ d'anomalies simulées selon une loi uniforme. La figure 5.9 présente les taux moyens de classification correcte et d'anomalies correctement détectées sur 25 réplications de la simulation pour HDRC et TCLUST [40], qui est une version robuste de k-means. On observe que les deux méthodes détectent correctement les anomalies mais que seule HDRC n'est pas sensible à la dimension des données.

Nous avons ensuite appliqué la méthode HDRC à un jeu de données fourni par le service *health monitoring* de SNECMA. Le jeu de données correspond au test en banc d'essai d'un moteur SNECMA pour lequel nous savons qu'il y a un fonctionnement anormal du moteur car le test a été arrêté par mesure de sécurité. Le jeu considéré contient 10 476 observations mesurées sur 173 variables décrivant le comportement du moteur au cours du test. L'image de gauche de la figure 5.10 illustre à l'aide de la vitesse de rotation du moteur le déroulement du test et l'image de

droite correspond à un zoom sur la zone encadrée à gauche. Les anomalies détectées par HDRC sont indiquées sur l'image de droite par des croix rouges et l'arrêt brutal qui suit nous confirme qu'il s'agit bien des anomalies à détecter.

Nous travaillons actuellement sur l'extension « online » du clustering robuste. Cela pose trois problèmes principaux : l'estimation online des paramètres du modèles, le choix dynamique du nombre de groupes et le réglage du paramètre α également de façon dynamique.

5.6 Logiciels : HDDA/C, LLN, AdaptReg, HDclassif et FisherEM

Dans le but de faciliter la diffusion des méthodologies que nous avons proposées, nous nous sommes efforcés de développer des paquets pour les logiciels R et Matlab implantant nos méthodes. Nous avons ainsi développé 2 *toolboxes* pour Matlab et 4 paquets pour R. Les *toolboxes* HDDA et HDDC pour Matlab implantent les méthodes de classification éponymes présentées au paragraphe 2.1. Notons que ces méthodes ont également été incorporées dans le logiciel MIXMOD, développé par l'Inria et l'Université de Franche-Comté, pour la classification générative des données quantitatives et qualitatives. Nous avons de plus récemment développé un paquet, baptisé HDclassif, qui propose les méthodes HDDA et HDDC pour le logiciel R. Nous avons en outre publié un article [B14] présentant l'usage pratique de ce paquet pour la classification des données de grande dimension. La méthode SLS de classification supervisée des nœuds d'un réseau, présentée au paragraphe 4.1, a également été implantée sous la forme d'un paquet R, baptisé LLN. Le paquet AdaptReg implante quant à lui la régression adaptative présentée au paragraphe 3.1. Enfin, le paquet FisherEM propose la méthode de clustering FisherEM, présentée au paragraphe 2.3, aux utilisateurs du logiciel R. Il est prévu d'ajouter prochainement à ce paquet les versions sparses de l'algorithme FisherEM.

6

Conclusion et perspectives

Les chapitres précédents reflètent l'état actuel de mes recherches sur les questions d'apprentissage statistique en grande dimension, adaptatif et sur données atypiques. Les contributions que mes coauteurs et moi-même avons apporté à ces trois thématiques ont toutefois soulevé de nouvelles questions ou pistes de recherche que nous souhaitons aborder dans les prochaines années. Je détaille ci-dessous quelques unes de ces pistes de recherche.

Convergence de HDDC quand les dimensions d_k peuvent évoluer Nous souhaitons dans un futur proche nous intéresser aux propriétés de convergence de l'algorithme HDDC dans le cas où les dimensions intrinsèques d_k, $k = 1, ..., K$, ne sont pas supposées être fixes au cours des itérations. En effet, autant la convergence de HDDC vers un maximum local de la vraisemblance est garantie pour des dimensions d_k fixées puisqu'alors HDDC est un algorithme de type EM, cette convergence n'est a priori plus garantie si les dimensions d_k ont la liberté de changer au cours des itérations de l'algorithme. Nous pensons qu'il sera possible de montrer que, dans ce cas, HDDC est un algorithme EM généralisé (GEM), éventuellement sous une hypothèse faible sur le schéma d'évolution des dimensions d_k au cours des itérations. Ce travail pourra en outre être généralisé aux « méthodes sœurs » de HDDC telles que les méthodologies proposées par [66, 67, 71, 97].

Sélection de modèles en sparsité Les travaux que nous avons mené sur la sélection de variables discriminantes à l'aide d'une approche par pénalisation de type ℓ_1 ont soulevé l'absence de travaux théoriques sur la sélection de modèles en sparsité. En effet, les critères de sélection de modèles actuels n'autorisent pas de prendre en compte proprement la sparsité éventuelle des modèles en compétition. De plus, la forme de la plupart de ces critères va en l'encontre de la sélection de modèles sparses. Des expériences préliminaires que nous avons mené ont également montré que les critères classiques AIC et BIC ne permettent pas de sélectionner de façon stable le modèle sparse à utiliser pour un jeu de données. Nous pensons que le critère ICL, qui prend en compte l'aspect classification en plus de l'aspect de modélisation, pourrait être un outil plus approprié pour sélectionner un modèle sparse parmi plusieurs. Cependant, il sera certainement nécessaire de proposer un nouveau critère dédié qui prenne en compte nativement la sparsité des modèles en compétition.

Composantes non observées en mélange de régression Comme nous l'avons vu au chapitre 3, le modèle de mélange de régression est un modèle fort utile, en Économie par exemple. Cependant, lors de l'étude d'un phénomène au cours du temps ou du point de vue géographique, il est naturel de se demander si le nombre de composantes estimé sur la population d'apprentissage reste valide pour la population de prédiction. Il se peut en particulier que certaines composantes n'aient pas été observées dans la population d'apprentissage ou, au contraire, qu'elles aient disparu dans la population de prédiction. Pour solutionner ce problème, nous pensons utiliser une approche par sélection de modèles qui comparera a posteriori des modèles avec un nombre variable de composantes et adaptés à partir du modèle appris sur la population d'apprentissage.

Détection online de nouveautés Concernant l'application au *health monitoring* en aéronautique, nous envisageons d'étendre rapidement la méthode non supervisée de détection de nouveautés, que nous avons proposé, au cas « online », *i.e.* les observations arrivent les unes après les autres au cours du temps et ne sont traitées qu'une seule fois par l'algorithme. Nous pourrons nous baser sur les travaux existants en inférence online avec l'algorithme EM mais les difficultés viendront principalement de la sélection online du nombre de composantes du modèle et de la proportion α d'anomalies à détecter. Concernant la détection online des anomalies, il sera certainement nécessaire de trouver un compromis entre « mise en mémoire » et « oubli » car certaines observations pourront être vues comme des anomalies à un instant T et des données « normales » à un instant $T + t$.

A plus long terme, je souhaiterais d'une part considérer l'inférence dans un cadre bayésien des méthodes de classification dans des sous-espaces, ce qui autoriserait notamment un choix de sous-espaces plus large. J'ambitionne d'autre part de faire un lien entre les méthodes génératives et les méthodes à noyaux dans le cadre de l'analyse des réseaux. Je voudrais également généraliser l'inférence dite « online » à nos méthodes de classification et de régression car le traitement des « flux de données » me semble être un enjeu futur important en apprentissage statistique. Enfin, Je prévois de maintenir un fort lien avec les applications car je sais, par expérience, que les problématiques applicatives appellent souvent des formulations théoriques élégantes et des solutions dont les champs d'usage sont généralement très larges.

Bibliographie

[1] E.M. Airoldi, D.M. Blei, S.E. Fienberg, and E.P. Xing. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9 :1981–2014, 2008. *Cité page 55*

[2] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6) :716–723, 1974. *4 citations pages 15, 19, 38 et 52*

[3] T. Alexandrov, J. Decker, B. Mertens, A.M. Deelder, R.A. Tollenaar, P. Maass, and H. Thiele. Biomarker discovery in MALDI-TOF serum protein profiles using discrete wavelet transformation. *Bioinformatics*, 25(5) :643–649, 2009. *Cité page 74*

[4] David M. Allen. The relationship between variable selection and data augmentation and a method for prediction. *Technometrics*, 16 :125–127, 1974. *Cité page 38*

[5] J. Baek, G. McLachlan, and L. Flack. Mixtures of Factor Analyzers with Common Factor Loadings : Applications to the Clustering and Visualisation of High-Dimensional Data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–13, 2009. *Cité page 21*

[6] M. Barker and W. Rayens. Partial least squares for discrimination. *J. Chemometrics*, 17 :166–173, 2003. *Cité page 75*

[7] S. Bashir and E. Carter. High breakdown mixture discriminant analysis. *Journal of Multivariate Analysis*, 93(1) :102–111, 2005. *Cité page 45*

[8] A. Bellas, C. Bouveyron, M. Cottrell, and J. Lacaille. Robust clustering of high-dimensional data. In *Proceedings of the 20th European Symposium on Artificial Neural Networks*, 2012. *Cité page 78*

[9] R. Bellman. *Dynamic Programming*. Princeton University Press, 1957. *Cité page 11*

[10] H. Bensmail and G. Celeux. Regularized Gaussian discriminant analysis through eigenvalue decomposition. *Journal of the American Statistical Association*, 91 :1743–1748, 1996. *2 citations pages 12 et 15*

[11] C. Bernard-Michel, S. DoutÃ©, M. Fauvel, L. Gardes, and S. Girard. Retrieval of Mars surface physical properties from OMEGA hyperspectral images using Regularized Sliced Inverse Regression. *Journal of Geophysical Research*, to appear, 2009. *Cité page 77*

[12] J.-P. Bibring and 42 co authors. *OMEGA : Observatoire pour la Minéralogie, l'Eau, les Glaces et l'Activité*, page 37 49. ESA SP-1240 : Mars Express : the Scientific Payload, 2004. *Cité page 77*

[13] C. Biernacki, F. Beninel, and V. Bretagnolle. A generalized discriminant rule when training population and test population differ on their descriptive parameters. *Biometrics*, 58(2) :387–397, 2002. *Cité page 36*

[14] C. Biernacki, G. Celeux, and G. Govaert. Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7) :719–725, 2000. *2 citations pages 15 et 52*

[15] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Workshop on Computational Learning Theory*, pages 92–100, 1998. *Cité page 50*

[16] H.-H. Bock. Probabilistic models in cluster analysis. *Computational Statistics and Data Analysis*, 23(1) :5–28, 1996. *Cité page 11*

[17] R. Boulet, B. Jouve, F. Rossi, and N. Villa. Batch kernel SOM and related Laplacian methods for social network analysis. *Neurocomputing*, 71(7–9) :1257–1273, March 2008. *Cité page 56*

[18] C. Bouveyron and H. Chipman. Visualization and classification of graph-structured data : the case of the Enron dataset. In *20th International Joint Conference on Neural Networks*, pages 1506–1511, 2007. *Cité page 55*

[19] C. Bouveyron, H. Chipman, and E. Côme. Supervised classification and visualization of social networks based on a probabilistic latent space model. In *7th International Workshop on Mining and Learning with Graphs, Leuven, Belgium*, 2009. *4 citations pages 55, 56, 57 et 60*

[20] C. Bouveyron, S. Girard, and M. Olteanu. Supervised classification of categorical data with uncertain labels for dna barcoding. In *17th European Symposium on Artificial Neural Networks*, pages 29–34, 2009. *Cité page 46*

[21] R. Cattell. The scree test for the number of factors. *Multivariate Behavioral Research*, 1(2) :145–276, 1966. *3 citations pages 15, 20 et 67*

[22] G. Celeux and J. Diebolt. The SEM algorithm : a probabilistic teacher algorithm from the EM algorithm for the mixture problem. *Computational Statistics Quaterly*, 2(1) :73–92, 1985. *Cité page 14*

[23] G. Celeux and G. Govaert. A Classification EM Algorithm for Clustering and Two Stochastic versions. *Computational Statistics and Data Analysis*, 14 :315–332, 1992. *2 citations pages 14 et 27*

[24] G. Celeux, M. Hurn, and C. Robert. Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association*, 95 :957–970, 2000. *Cité page 42*

[25] J. Couto. Kernel k-means for categorical data. In *Advances in Intelligent Data Analysis VI*, volume 3646 of *Lecture Notes in Computer Science*, pages 739–739. 2005. *Cité page 69*

[26] F. d'Alche Buc, I. Dagan, and J. Quinonero, editors. *The 2005 Pascal visual object classes challenge*. Proceedings of the first PASCAL Challenges Workshop. Springer, 2006. *2 citations pages 72 et 73*

[27] B. Dasarathy. Noising around the neighbourhood : a new system structure and classification rule for recognition in partially exposed environments. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2 :67–71, 1980. *Cité page 45*

[28] A. Delaigle and P. Hall. Defining probability density for a distribution of random functions. *The Annals of Statistics*, 38 :1171–1193, 2010. *Cité page 60*

[29] A. Dempster, N. Laird, and D. Robin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1) :1–38, 1977. *3 citations pages 14, 25 et 40*

[30] M.M. Dundar and D.A. Landgrebe. Toward an optimal supervised classifier for the analysis of hyperspectral data. *Geoscience and Remote Sensing, IEEE Transactions on*, 42(1) :271 – 277, jan. 2004. *Cité page 70*

[31] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32 :407–499, May 2004. *2 citations pages 30 et 31*

[32] M. Fan, H. Qiao, and B. Zhang. Intrinsic dimension estimation of manifolds by incising balls. *Pattern Recognition*, 42(5) :780–787, 2009. *Cité page 17*

[33] R.A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7 :179–188, 1936. *4 citations pages 12, 21, 26 et 47*

[34] B. Flury. Common principal components in k groups. *Journal of American Statistical Association*, 79 :892–897, 1984. *Cité page 21*

[35] D.H. Foley and J.W. Sammon. An optimal set of discriminant vectors. *IEEE Transactions on Computers*, 24 :281–289, 1975. *Cité page 24*

[36] C. Fraley and A. Raftery. MCLUST : Software for Model-Based Cluster Analysis. *Journal of Classification*, 16 :297–306, 1999. *Cité page 16*

[37] C. Fraley and A. Raftery. Model-based clustering, discriminant analysis and density estimation. *Journal of American Statistical Association*, 97 :611–631, 2002. *Cité page 12*

[38] J.H. Friedman. Regularized discriminant analysis. *The Journal of the American Statistical Association*, 84 :165–175, 1989. *Cité page 12*

[39] L.A. García-Escudero, A. Gordaliza, and C. Matrán. Trimming tools in exploratory data analysis. *Journal of Computational and Graphical Statistics*, 12(2) :434–449, 2003. *Cité page 79*

[40] L.A. García-Escudero, A. Gordaliza, C. Matrán, and A. Mayo-Iscar. A general trimming approach to robust cluster analysis. *The Annals of Statistics*, 36(3) :1324–1345, 2008. *Cité page 79*

[41] I. Guyon, U. Von Luxburg, and R. Williamson. Clustering : Science or art ? In *NIPS 2009 Workshop on Clustering Theory*, 2009. *Cité page 26*

[42] I. Guyon, N. Matic, and V. Vapnik. Discovering informative patterns and data cleaning. *Advances in Knowledge Discovery and Data Mining*, pages 181–203, 1996. *Cité page 45*

[43] M. Handcock, A. Raftery, and J. Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society, Series A*, 170(2) :1–22, 2007. *4 citations pages 55, 56, 58 et 69*

[44] T. Hastie, A. Buja, and R. Tibshirani. Penalized discriminant analysis. *Annals of Statistics*, 23 :73–102, 1995. *Cité page 12*

[45] T. Hastie and R. Tibshirani. Discriminant analysis by gaussian mixture. *Journal of the Royal Statistical Society*, 58(1) :155–176, 1996. *2 citations pages 46 et 47*

[46] D. Hawkins and G. McLachlan. High-breakdown linear discriminant analysis. *Journal of the American Statistical Association*, 92(437) :136–143, 1997. *Cité page 45*

[47] P. Hoff, A. Raftery, and M. Handcock. Latent spaces approaches to social network analysis. *Journal of the American Statistical Association*, 97(460) :1090–1098, 2002. *3 citations pages 55, 56 et 59*

[48] J. Jacques and C. Biernacki. Extension of model-based classification for binary data when training and test populations differ. *Journal of Applied Statistics*, 37(5) :749–766, 2010. *Cité page 36*

[49] G.M. James and C.A. Sugar. Clustering for sparsely sampled functional data. *Journal of the American Statistical Association*, 98(462) :397–408, 2003. *Cité page 61*

[50] I. T. Jolliffe. *Principal Component Analysis*. Springer, 2002. *Cité page 12*

[51] B. Krishnapuram, D. Williams, Y. Xue, A. Hartemink, L. Carin, and M. Figueiredo. On semi-supervised classification. In *NIPS*, 2004. *Cité page 50*

[52] N. Lawrence and B. Schölkopf. Estimating a kernel Fisher discriminant in the presence of label noise. In *Proc. of 18th International Conference on Machine Learning*, pages 306–313. Morgan Kaufmann, San Francisco, CA, 2001. *2 citations pages 45 et 48*

[53] E. Levina and P. Bickel. Maximum Likelihood Estimation of Intrinsic Dimension. In *17th Annual Conference on Neural Information Processing Systems*, 2005. *2 citations pages 17 et 20*

[54] Y. Li, L. Wessels, D. de Ridder, and M. Reinders. Classification in the presence of class noise using a probabilistic kernel Fisher method. *Pattern Recognition*, 40(12) :3349–3357, 2007. *Cité page 45*

[55] B.G. Lindsay. Mixture models : Theory, geometry and applications. In *NSF- CBMS Regional Conference Series in Probability and Statistics*, volume 5. Institute of Mathematical Statistics, 1995. *Cité page 24*

[56] S. A. Macskassy and F. Provost. Classification in networked data : A toolkit and a univariate case study. *Journal of Machine Learning Research*, 8 :935–983, may 2007. *Cité page 56*

[57] M. Markou and S. Singh. Novelty detection : A review - part 1 : Statistical approaches. *Signal Processing*, 83(12) :2481–2497, 2003. *Cité page 50*

[58] M. Markou and S. Singh. Novelty detection : A review - part 2 : Neural network based approaches. *Signal Processing*, 83(12) :2499–2521, 2003. *Cité page 50*

[59] C. Maugis, G. Celeux, and M.-L. Martin-Magniette. Variable selection for Clustering with Gaussian Mixture Models. *Biometrics*, 65(3) :701–709, 2009. *3 citations pages 28, 29 et 32*

[60] C. Maugis, G. Celeux, and M.-L. Martin-Magniette. Variable selection in model-based clustering : A general variable role modeling. *Computational Statistics and Data Analysis*, 53 :3872–3882, 2009. *2 citations pages 28 et 29*

[61] G. McLachlan. Iterative reclassification procedure for constructing an asymptotically optimal rule of allocation in discriminant analysis. *Journal of the American Statistical Association*, (70) :365–369, 1975. *Cité page 50*

[62] G. McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*. Wiley, New York, 1992. *2 citations pages 11 et 52*

[63] G. McLachlan and T. Krishnan. *The EM algorithm and extensions*. Wiley Interscience, New York, 1997. *2 citations pages 11 et 14*

[64] G. McLachlan and D. Peel. *Finite Mixture Models*. Wiley Interscience, New York, 2000. *Cité page 11*

[65] G. McLachlan, D. Peel, and R. Bean. Modelling high-dimensional data by mixtures of factor analyzers. *Computational Statistics and Data Analysis*, (41) :379, 2003. *Cité page 17*

[66] G. McLachlan, D. Peel, and R. Bean. Modelling high-dimensional data by mixtures of factor analyzers. *Computational Statistics and Data Analysis*, 41 :379–388, 2003. *Cité page 81*

[67] P. McNicholas and B. Murphy. Parsimonious Gaussian mixture models. *Statistics and Computing*, 18(3) :285–296, 2008. *2 citations pages 17 et 81*

[68] S. Mika, G. Ratsch, J. Weston, B. Schölkopf, and K.R. Müllers. Fisher discriminant analysis with kernels. In *Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop*, August 1999. *Cité page 70*

[69] J. Mingers. An empirical comparison of pruning methods for decision tree induction. *Journal of Machine Learning*, 4(2) :227–243, 1989. *Cité page 45*

[70] T. Minka. Automatic choice of dimensionality for PCA. In *13th Annual Conference on Neural Information Processing Systems*, 2000. *2 citations pages 17 et 20*

[71] A. Montanari and C. Viroli. Heteroscedastic Factor Mixture Analysis. *Statistical Modeling : An International journal*, 10(4) :441–460, 2010. *2 citations pages 17 et 81*

[72] J.L. Moreno. *Who shall survive ? : a new approach to the problem of Human interrelations*. Nervous and Mental Disease Publishing, Washington DC, 1934. *Cité page 55*

[73] M.E.J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review Letter*, 69, 2004. *Cité page 56*

[74] T. O'Neill. Normal discrimination with unclassified observations. *Journal of the American Statistical Association*, (73) :821–826, 1978. *Cité page 50*

[75] W. Pan and X. Shen. Penalized model-based clustering with application to variable selection. *Journal of Machine Learning Research*, 8 :1145–1164, 2007. *Cité page 29*

[76] T. Pavlenko. On feature selection, curse of dimensionality and error probability in discriminant analysis. *Journal of Statistical Planning and Inference*, 115 :565–584, 2003. *Cité page 11*

[77] T. Pavlenko and D. Von Rosen. Effect of dimensionality on discrimination. *Statistics*, 35(3) :191–213, 2001. *Cité page 11*

[78] Z. Qiao, L. Zhou, and J.Z. Huang. Sparse linear discriminant analysis with applications to high dimensional low sample size data. *International Journal of Applied Mathematics*, 39(1), 2009. *Cité page 31*

[79] A. Raftery and N. Dean. Variable selection for model-based clustering. *Journal of the American Statistical Association*, 101(473) :168–178, 2006. *4 citations pages 16, 28, 29 et 32*

[80] J. O. Ramsay and B. W. Silverman. *Functional data analysis*. Springer Series in Statistics. Springer, New York, second edition, 2005. *4 citations pages 60, 61, 63 et 64*

[81] F. Rossi and N. Villa-Vialaneix. Représentation d'un grand réseau à partir d'une classification hiérarchique de ses sommets. *Journal de la Société Française de Statistique*, 152(3) :34–65, 2011. *Cité page 56*

[82] P.J. Rousseeuw and A. Leroy. *Robust Regression and Outlier Detection*. Wiley, New York, 1987. *Cité page 45*

[83] S. Sampson. *A novitiate in a period of change : An experimental and case study of relationships*. PhD thesis, Department of Sociology, Cornell University, 1968. *Cité page 58*

[84] B. Scholkopf and A. Smola. *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001. *Cité page 70*

[85] B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, 2002. *Cité page 16*

[86] J. Schott. Dimensionality reduction in quadratic discriminant analysis . *Computational Statistics and Data Analysis*, 66 :161–174, 1993. *Cité page 12*

[87] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2) :461–464, 1978. *5 citations pages 15, 17, 19, 38 et 52*

[88] D. Scott and J. Thompson. Probability density estimation in higher dimensions. In *Fifteenth Symposium in the Interface*, pages 173–179, 1983. *Cité page 12*

[89] B. Shölkopf, R. Williamson, A. Smola, J. Taylor, and J. Platt. Support vector method for novelty detection. In *Advances in Neural Information Processing Systems*, pages 582–588, 2000. *2 citations pages 50 et 53*

[90] A. Smola and R. Kondor. Kernels and regularization on graphs. In *Proc. Conf. on Learning Theory and Kernel Machines*, pages 144–158, 2003. *Cité page 69*

[91] A. Storkey and M. Sugiyama. *Mixture regression for covariate shift*, pages 1337–1344. Advances in Neural Information Processing Systems 19. MIT Press, Cambridge, 2007. *Cité page 36*

[92] M. Sugiyama. Active learning in approximately linear regression based on conditional expectation of generalization error. *Journal of Machine Learning Research*, 7 :141–166, 2006. *Cité page 36*

[93] M. Sugiyama, T. Idé, S. Nakajima, and J. Sese. Semi-supervised local Fisher discriminant analysis for dimensionality reduction. *Machine Learning*, 78 :35–61, 2009. *Cité page 49*

[94] M. Sugiyama and K-R. Müller. Input-dependent estimation of generalization error under covariate shift. *Statistics & Decisions*, 23, 2005. *Cité page 36*

[95] M. Sugiyama and Krauledat M. Müller, K-R. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8 :985–1005, 2007. *Cité page 36*

[96] D. Tax and R. Duin. Outlier detection using classifier instability. In *Advances in Pattern Recognition*, pages 251–256, 1999. *2 citations pages 50 et 53*

[97] E. Tipping and C. Bishop. Mixtures of Probabilistic Principal Component Analysers. *Neural Computation*, 11(2) :443–482, 1999. *3 citations pages 17, 18 et 81*

[98] M. Tipping and C. Bishop. Probabilistic Principal Component Analysis. *Journal of the Royal Statistical Society, Series B*, 3(61) :611–6222, 1999. *2 citations pages 17 et 18*

[99] D. Tyler. Asymptotic Inference for Eigenvectors. *Annals of Statistics*, 9(4) :725–736, 1981. *Cité page 19*

[100] D.M. Witten and R. Tibshirani. A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490) :713–726, 2010. *2 citations pages 29 et 32*

[101] D.M. Witten, R. Tibshirani, and T. Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistic*, 10(3) :515–534, 2009. *Cité page 31*

[102] S. Wold. Pattern recognition by means of disjoint principal component models. *Pattern Recognition*, 8 :127–139, 1976. *Cité page 75*

[103] B. Xie, W. Pan, and X. Shen. Penalized mixtures of factor analyzers with application to clustering high-dimensional microarray data. *Bioinformatics*, 26(4) :501–508, 2010. *Cité page 29*

[104] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society*, 67 :301–320, 2005. *Cité page 30*

[105] H. Zou, T. Hastie, and R. Tibshirani. On the degrees of freedom of the Lasso. *Annals of Statistics*, 35(5) :2173–2192, 2007. *Cité page 31*