Table des matières

Principales notations

K, K_{sup}	:	nombre de classes et nombre maximal de classes
$\mathcal{M}$, Δ	:	modèle et ensemble de modèles
$\boldsymbol{x}_i$, $\mathcal{X}$	:	une donnée de $\mathbb{R}^d$ et un ensemble $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$
p_k, $\boldsymbol{\mu}_k$, $\boldsymbol{\Sigma}_k$	:	proportion, centre et matrice de variance de la classe k
$\phi_k(\boldsymbol{x}\|\boldsymbol{a}_k)$	:	densité gaussienne pour la classe k avec $\boldsymbol{a}_k = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
λ_k, $\boldsymbol{D}_k$, $\boldsymbol{A}_k$	:	volume, matrice orientation et matrice forme de la classe k
$f(\boldsymbol{x}\|\boldsymbol{\theta})$	:	densité mélange paramétrée par $\boldsymbol{\theta} = (p_1, \ldots, p_K, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_K)$
z_i, $\boldsymbol{z}$	:	classe d'appartenance de $\boldsymbol{x}_i$ et une partition de $\mathcal{X}$
$u(\boldsymbol{x})$	:	règle de classement de $\mathbb{R}^d$ dans $\{1, \ldots, K\}$
$t_k(\boldsymbol{x}_i\|\boldsymbol{\theta})$, t_{ik}	:	probabilités que $\boldsymbol{x}_i$ appartienne à la classe k
$\boldsymbol{t}$, $\boldsymbol{c}$	:	partitions floues
$\mathcal{L}(a\|b)$	:	coût de choix de a alors que b vrai
$\mathcal{R}(a\|\boldsymbol{x})$, $\mathcal{R}(a)$	:	risques conditionnel et moyen de a
$\mathrm{P}_e(u)$	:	erreur de classement obtenue avec u
$\pi(\boldsymbol{\theta})$, $\pi(\boldsymbol{\theta}\|\boldsymbol{x})$	:	loi a priori et loi conditionnelle de $\boldsymbol{\theta}$
$\mathrm{KL}(f_1, f_2)$	:	divergence de Kullback-Leibler entre les distributions f_1 et f_2
$\mathrm{L}(\boldsymbol{\theta})$	:	log-vraisemblance de $\boldsymbol{\theta}$
$\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{z})$	:	log-vraisemblance classifiante de $(\boldsymbol{\theta}, \boldsymbol{z})$
i.i.d.	:	indépendamment et identiquement distribué
MAP	:	maximum a posteriori
$q(\mathcal{M})$	:	nombre de paramètres libres dans $\mathcal{M}$
$\boldsymbol{W}$, $\boldsymbol{W}_k$	:	matrice d'inertie intraclasse puis de la classe k
$\mathrm{E}^\pi[\boldsymbol{\theta}\|\boldsymbol{x}]$	:	espérance conditionnelle de $\boldsymbol{\theta}$ avec la loi $\pi(\boldsymbol{\theta}\|\boldsymbol{x})$
$\mathrm{E}_X[\boldsymbol{\theta}]$	:	espérance $\boldsymbol{\theta}$ avec la loi la variable aléatoire $\boldsymbol{X}$
$\mathbb{1}_{[a,b]}(x)$	:	fonction indicatrice
$a \propto b$	:	a est proportionnel à b
$\mathcal{U}_{[a,b]}(x)$	:	loi uniforme unidimensionnelle sur $[a, b]$
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	:	loi normale et ses paramètres

Introduction générale

De l'idée de classifier au modèle de mélange gaussien

La classification est une méthode d'analyse des données et, à ce titre, elle propose une vue réductrice et simplifiée d'un ensemble de données. Le moyen qu'elle utilise est le regroupement des éléments similaires dans une même classe et les classes obtenues sont alors qualifiées d'homogènes. Les groupes créés doivent être interprétables par une personne du domaine dont sont issues les données pour permettre à cette personne de mieux les comprendre. La classification se scinde naturellement en deux sous-ensembles bien distincts : la *création de groupes* (on parle aussi de classification) et le *classement dans des groupes* déjà existants. La création de groupes est donc une démarche qui précède généralement le classement. Le présent travail s'appuie sur une hypothèse de mélange gaussien pour permettre *l'automatisation* des deux démarches de la classification : on parle de *classification automatique* lorsque l'on automatise la création de groupes, et on parle de *discrimination* lorsque l'on automatise le classement. Dans le cadre gaussien, l'objectif sera de proposer et de comparer des méthodes qui favorisent l'automatisation des deux concepts : on parle dans ce cas de *choix de modèles* (par exemple, il faut choisir le nombre de classes en classification automatique). Voyons maintenant que ces préoccupations, bien qu'anciennes à la base, révèlent des aspects très contemporains.

L'idée de créer des groupes a évolué au fil du temps en se systématisant. C'est une démarche ancienne, Aristote proposant déjà des classifications en logique, éthique, politique et sur les animaux (il regroupa près de 500 animaux en espèces). Notre classification actuelle des animaux et des plantes fut établie du temps de Linné (1763). Il voyait cette méthode comme un outil descriptif, servant à nommer et à stocker les connaissances. Darwin (1859) interpréta ensuite cet arbre de classification comme représentant l'évolution des espèces. Chaque nœud de l'arbre correspond à un ancêtre et les sous-branches s'en détachant forment de nouvelles espèces par un mécanisme de mutation et de sélection naturelle. La classification ainsi obtenue est hiérarchique, c'est-à-dire correspond à une suite de partitions emboîtées. Plus récemment, dans le journal le Monde du 15 mars 1997, on pouvait voir un exemple de classification (appelée typologie en marketing) du grand public en 7 classes dans son comportement face à l'informatique : la classe des développeurs, des créatifs, des joueurs, des pilotes, des communicants, des utilitaristes et des encyclopédistes.

> Bernard Cathelat, sociologue inventeur des «socio-styles», s'est penché sur la typologie du grand public confronté à l'informatique. Pour lui, les familles déjà équipées font partie des «deus ex machina» passionnées de «hard et de soft», et des «créatives» attirées par l'ordinateur comme «outil d'expression et de communication». Il estime que les plus motivés pour franchir le pas sont les «surfers, jeunes passionnés de jeux vidéo», mais qui attendent la baisse des prix du multimédia. Quant aux «pilotes», ils sont «aisés et diplomés», mais restent «mieux équipés au bureau qu'à la maison». Ils investiront s'ils sont motivés par l'éducation des enfants ou avec «l'objectif de maîtrise gestionnaire du quotidien».
>
> Reste le profil «tribal», c'est-à-dire en quête d'un nouveau lien social et la catégorie «utilitariste» qui cherche à alléger la vie pratique. Ceux là sont, selon Bernard Cathelat, «freinés par les coûts et intimidés par une technologie encore complexe». Enfin, «les encyclopédistes», plus âgés et conservateurs, ne se laisseront séduire que par «le biais de la culture et des connaissances».

Cette classification pourrait servir à identifier les destinataires d'une campagne publicitaire de vente de matériel informatique. Remarquons que les classes formées ne définissent pas nécessairement ici une partition du grand public, un même individu pouvant appartenir à plusieurs classes à la fois, comme la classe des créatifs et celle des joueurs. Il s'agit donc d'une classification floue ou non exclusive. Une fois ces groupes définis, tout nouveau candidat peut être classé dans une ou plusieurs des 7 classes existantes (c'est l'aspect classement de la classification). Aujourd'hui, l'informatique est un instrument qui participe au développement des démarches de classification. Les raisons sont les suivantes :

- **la demande de traitement des données augmente** car l'informatique et les moyens numériques de communication associés permettent le stockage, la manipulation et le transfert de quantités importantes de données. Les faibles prix d'accès à ces techniques (matériellement et humainement) ont favorisé la collecte des données dans de nombreux domaines : mesures physiologiques, physiques et économiques, résultats sportifs, sondages d'opinion, ... Le besoin d'extraire l'information pertinente de ces données est alors un enjeu d'actualité ;

- **la capacité de traitement des données augmente** car la puissance de calcul s'accroit et permet d'implanter des techniques autrefois lourdes en calcul : algorithmes d'optimisation de critères, manipulation de matrices, graphiques, ...

Dans le cas de données mutivariées de l'espace euclidien, l'hypothèse de mélange gaussien est très utilisée maintenant (Wolfe [91], Bock [19]). Elle permet de retrouver, voire de généraliser, des méthodes classiques et non probabilistes de classification ainsi que d'apporter une grande cohérence. Une classe est représentée par une loi normale pondérée ; elle est donc caractérisée par trois paramètres : une *proportion*, un *centre* et une *dispersion* autour de ce centre (donnée par la matrice de variance). Le nombre de gaussiennes dans le mélange correspond alors au nombre de classes dans les données et la définition rigoureuse d'une classe qui résulte de cette hypothèse facilite l'automatisation des démarches de classification. Précisons que nous nous limiterons à la classification qui produit (en création de classes) ou utilise (en classement) des partitions (on parle de classification exclusive), donc une donnée appartient à une seule et unique classe et aucune classe n'est vide. Nous ne considérerons donc ni les classifications non exclusives, ni les structures hiérarchiques.

Choisir un modèle dans le cas gaussien

La classification automatique gaussienne se heurte a des problèmes importants qui limitent son aspect automatique. En effet, l'utilisateur est généralement obligé de définir le nombre de classes. D'autre part, Celeux et Govaert [38] ont proposé une décomposition spectrale des matrices de variance qui conduit à contrôler facilement un ensemble de propriétés des classes. L'utilisateur peut ainsi introduire des informations a priori assez variées et géométriquement interprétables sur les classes (classes de même forme, classes de même orientation, etc.) : ce sont les *modèles gaussiens*. L'introduction de telles contraintes, si ces dernières approchent bien la réalité, permet d'améliorer la *qualité de la partition* obtenue. Il est intéressant, dans le cas où l'utilisateur ne peut fournir une telle information par manque de connaissance du résultat, de choisir automatiquement le type de contraintes à sa place en comparant plusieurs a priori possibles. Le choix du nombre de classes ou du modèle gaussien, voire des deux simultanément, correspond à choisir un modèle.

En discrimination, on connaît déjà une partition des données en classes et il s'agit alors de *classer* toute nouvelle donnée dans un groupe existant : on ne crée donc généralement pas de nouveau groupe (du moins, nous ne considérerons pas cette possibilité). L'hypothèse de mélange gaussien est aussi employée pour définir les classes (McLachlan [69]) et on retrouve de nouveau la nécessité de choisir un a priori sur les classes : un modèle gaussien adapté permet de commettre moins d'erreur lors du classement des données.

Objectif du travail

L'objectif du travail est d'étudier des méthodes classiques et d'en proposer des nouvelles pour :

- choisir le nombre de classes et/ou le modèle gaussien en classification automatique gaussienne ;
- choisir le modèle gaussien en discrimination gaussienne.

La recherche du nombre de classes est un problème ancien et délicat. Notre originalité sera de le considérer sous un angle classification, c'est-à-dire de rechercher des *classes bien séparées*. Par contre, la détection du modèle gaussien est plus récente et elle est donc peu traitée dans la littérature. L'angle classification sera aussi notre méthode d'approche pour comparer des critères existants et en proposer de nouveaux.

Plan de la thèse

Le chapitre 1 pose les concepts fondamentaux. Nous présentons le mélange gaussien multivarié, ainsi que différentes façons classiques d'utiliser cette hypothèse en discrimination et en classification automatique. Nous introduisons ensuite les modèles gaussiens et leur intérêt en classification. Ces modèles traduisent une information a priori géométrique de l'utilisateur sur les paramètres du mélange.

Différentes méthodes d'estimation sont rappelées au chapitre 2. Nous voyons tout d'abord comment estimer le paramètre du mélange : maximum de vraisemblance et algorithmes associés (EM, etc.), point de vue bayésien et méthodes MCMC, principe de la théorie du codage. Ensuite nous considérons l'estimation dite mixte et ayant pour but d'estimer conjointement le paramètre du mélange *et* la partition. La vraisemblance classifiante est alors détaillée. Nous montrons qu'elle généralise des critères classiques en classification automatique puis nous énonçons des algorithmes la maximisant (CEM, etc.). Les principaux algorithmes précédents ont été finalement réunis dans un même programme écrit dans le langage Splus. Ce programme constitue notre boîte à outils de base pour les expériences du travail.

Le chapitre 3 est le chapitre clé de la thèse car il rappelle les principales méthodes de choix de modèles et en propose de nouvelles. L'approche mélange est rappelée avec les tests d'hypothèses et les critères d'information qui pénalisent la vraisemblance par la complexité du modèle (critères de type AIC). Nous proposons aussi une méthode de détection d'un coude dans la courbe du maximum de vraisemblance grâce à une hypothèse de précision sur les données. Ces méthodes peuvent alors être employées soit en classification automatique, soit en discrimination. Mais en classification automatique, nous proposons une approche originale qui recherche un modèle (nombre de classes ou modèle gaussien) produisant des classes bien séparées. Cette approche consiste à pénaliser la vraisemblance par une mesure de classifiabilité des données. On retrouve alors le critère de vraisemblance classifiante de Symons et aussi le critères des k-means de Sebestyen. En discrimination, nous choisissons l'approche classique du critère de validation croisée et des méthodes pour accélérer le calcul de ce critère sont présentées dans le cas de certains modèles gaussiens.

Dans le chapitre 4, nous comparons le comportement pour la détection du modèle gaussien de bon nombre des critères du chapitre précédent. La première comparaison se fait dans un cadre de classification automatique avec des données simulées et avec des données réelles : nous concluons à la bonne tenue du critère AIC3 de Bozdogan. Nous plaçant ensuite dans un cadre de discrimination, nous comparons les critères de type AIC au critère de validation croisée. À partir de nombreuses données simulées, nous déduisons le bon comportement de la validation croisée ainsi que, de nouveau, du critère AIC3.

Le dernier chapitre est l'occasion de comparer, en s'appuyant sur des données simulées et sur des données réelles, le comportement des critères pour la détection du nombre de classes en classification automatique (le modèle gausien est supposé connu). Nous voyons tout l'intérêt qu'apportent les critères pénalisant la vraisemblance par la classifiabilité des données : robustesse à un mauvais modèle, rare surestimation du nombre de classes. Parmi ces critères, il semble que ce soit le maximum de vraisemblance classifiante, un critère très simple donc, qui soit le plus performant.

Chapitre 1

Classification et modèle de mélange

1.1 Introduction

Ce chapitre présente la classification automatique et la discrimination sous une hypothèse de mélange gaussien. Cette hypothèse est aujourd'hui classique. Elle permet à l'utilisateur d'introduire facilement des informations a priori à forte connotation géométrique.

Dans la section 1.2 nous rappelons la définition et les propriétés d'un modèle de mélange et plus particulièrement du mélange gaussien multivarié. Dans le contexte gaussien, nous posons alors le problème de l'estimation du paramètre du mélange à partir d'un échantillon.

La section 1.3 introduit tout d'abord la problématique de la discrimination et la situe ensuite dans un cadre de modèle gaussien de mélange. Nous voyons en détail la méthode du Maximum a Posteriori (MAP), pour trouver une règle de classement lorsque le paramètre du mélange est connu. Deux méthodes utilisant la méthode du MAP sont présentées pour estimer la règle de classement : la méthode prédictive et celle du «plug-in». Ensuite nous étudions la classification automatique lorsque la structure recherchée est une partition. Nous énonçons plusieurs utilisations de l'hypothèse de mélange gaussien pour estimer la partition : l'approche mélange, l'approche classification et l'approche mixte.

Finalement, la section 1.4 montre l'intérêt d'introduire de l'information a priori sur les paramètres du mélange. Introduire de l'a priori permet de rendre plus robustes les estimations soit du paramètre du mélange, soit de la règle de classement, soit de la partition. Utilisant une décomposition spectrale des matrices de variance de Celeux et Govaert [38], nous voyons une manière d'introduire très facilement de l'information a priori géométriquement interprétable. Cette décomposition conduit à la définition de 14 types de contraintes différentes sur les paramètres, contraintes que nous appelons modèles gaussiens. Nous proposons aussi des contraintes sur les proportions qui, agglomérées aux 14 modèles déjà cités, conduisent à la définition de 28 modèles gaussiens.

1.2 Le modèle de mélange

Les mélanges de distributions sont un moyen d'obtenir des distributions relativement complexes, tout en s'appuyant à la base sur des lois très simples, comme les lois normales. Nous présentons la définition générale de ces mélanges pour nous restreindre rapidement au cas gaussien.

1.2.1 Définition

Une densité de mélange correspond à la superposition de densités plus simples. Soient $\boldsymbol{X}$ et $\boldsymbol{Z}$ deux variables aléatoires et soient $\phi(\boldsymbol{x}|\boldsymbol{z})$ la densité de la distribution de $\boldsymbol{X}$ conditionnellement à $\boldsymbol{Z}$ et $h(\boldsymbol{z})$ la densité de la distribution de $\boldsymbol{Z}$. La forme générale de la densité de mélange s'écrit

$$f(\boldsymbol{x}) = \int \phi(\boldsymbol{x}|\boldsymbol{z})h(\boldsymbol{z})d\boldsymbol{z}.$$

Quand Z ne prend qu'un ensemble fini de valeurs $\{1,\ldots,K\}$, on pose $Z = \boldsymbol{Z}$. Dans ce cas, la distribution de Z est discrète et affecte une probabilité positive à chacune de ces valeurs. L'intégrale est alors remplacée par une somme pour donner une densité de *mélange fini*. Cette densité s'énonce

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} p_k \phi_k(\boldsymbol{x}),$$

avec

$$\phi_k(\boldsymbol{x}) = \phi(\boldsymbol{x}|Z=k) \qquad \text{et} \qquad p_k = \mathrm{P}(Z=k).$$

Les probabilités p_k sont appelées proportions du mélange et sont assujetties aux deux contraintes suivantes : $\forall k \in \{1,\ldots,K\}$, $p_k > 0$ et $\sum_{k=1}^{K} p_k = 1$.

Une propriété importante que doivent posséder les mélanges est l'*identifiabilité*. Un mélange est identifiable s'il existe une correspondance unique entre la distribution du mélange et les composantes de ce mélange. Par exemple, un mélange de lois uniformes n'est pas identifiable. Pour le constater, prenons X une variable aléatoire réelle et prenons la loi uniforme sur $[a_k,b_k]$ que nous notons $\phi_k(x) = \mathcal{U}_{[a_k,b_k]}(x) = \mathbb{1}_{[a_k,b_k]}(x)\frac{1}{b_k-a_k}$. On a

$$f(x) = \begin{cases} 0.4\,\mathcal{U}_{[0,1]}(x) + 0.6\,\mathcal{U}_{[1,2]}(x) \\ \text{ou} \\ 0.8\,\mathcal{U}_{[0,2]}(x) + 0.2\,\mathcal{U}_{[1,2]}(x) \end{cases}.$$

Donc la loi $f(x)$ peut être ici décomposée de deux façons différentes.

1.2.2 Mélange gaussien multivarié

Définition

L'utilisation de mélanges gaussiens est très classique et le mélange résultant est toujours identifiable. Dans la situation multivariée, la variable aléatoire $\boldsymbol{X}$ est dans l'espace euclidien $\mathbb{R}^d$ et la densité de la composante k $(1 \leq k \leq K)$ s'écrit

$$\phi_k(\boldsymbol{x}|\boldsymbol{a}_k) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_k|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)\right), \tag{1.1}$$

avec $\boldsymbol{a}_k = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, $\boldsymbol{\mu}_k$ moyenne de la classe k ($\boldsymbol{\mu}_k \in \mathbb{R}^d$) et $\boldsymbol{\Sigma}_k$ sa matrice de variance-covariance (matrice symétrique définie positive $d \times d$).

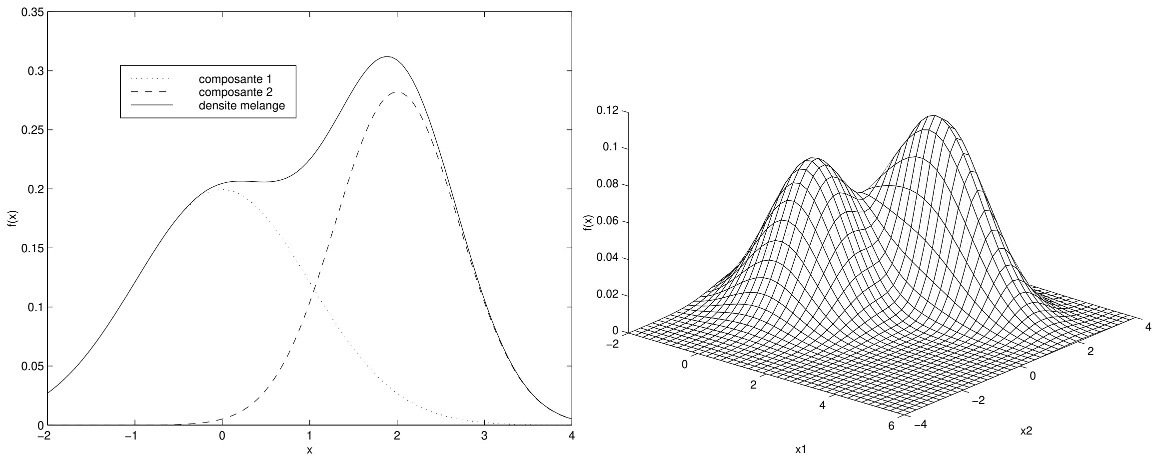
La densité de mélange s'exprime alors par

$$f(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{k=1}^{K} p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k). \tag{1.2}$$

Cette densité est paramétrée par

$$\boldsymbol{\theta} = \left((p_k)_{k=1,\ldots,K-1}, (\boldsymbol{a}_k)_{k=1,\ldots,K}\right). \tag{1.3}$$

La figure 1.1 propose des illustrations de mélange gaussien.



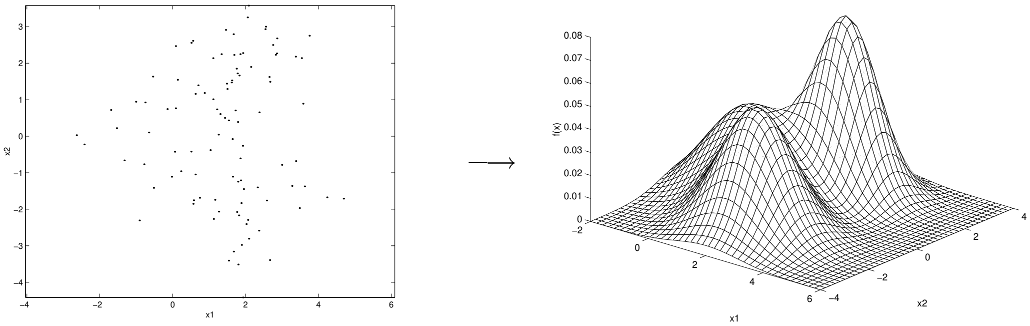
(a) : $d = 1$ (b) : $d = 2$

FIG. 1.1 – *Exemple d'un mélange gaussien univarié (a) et bivarié (b).*

Estimation du paramètre du mélange

L'estimation du paramètre du mélange consiste à inférer une valeur de $\boldsymbol{\theta}$ à partir d'un n–échantillon $\mathcal{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ dont chaque élément $\boldsymbol{x}_i$ de $\mathbb{R}^d$

$(1 \leq i \leq n)$ est une réalisation i.i.d. de la loi mélange $f(\boldsymbol{x}|\boldsymbol{\theta})$ (figure 1.2). Nous noterons $\hat{\boldsymbol{\theta}}$ un estimateur de $\boldsymbol{\theta}$. En 1894, Pearson [72] introduisit l'estimation du paramètre du mélange dans le cas d'un mélange bimodal univarié. Il utilisa la méthode des moments pour estimer le paramètre. D'autres méthodes seront présentées au chapitre 2. Généralement, le nombre de composantes est connu par avance.



données $\mathcal{X}$ $\longrightarrow$ paramètre $\hat{\boldsymbol{\theta}}$

FIG. 1.2 – *Principe de l'estimation du paramètre du mélange.*

1.3 Classification et mélange gaussien

La classification regroupe deux grandes approches distinctes : la discrimination et la classification automatique. Dans les deux situations, l'hypothèse de mélange gaussien est classiquement employée. Cette section utilise des estimateurs obtenus par des méthodes qui sont décrites au chapitre 2. Nous présentons tout d'abord les notions de base que nous utilisons par la suite : la définition d'une classe gaussienne et le type de données utilisé.

1.3.1 Notions de base

Classe gaussienne

La classification, que ce soit en discrimination ou en classification automatique, repose sur la définition de deux concepts :

- **la classe** ;
- **une mesure de ressemblance** (ou de dissemblance) entre des éléments d'un espace et la classe.

Un élément de l'espace est dit appartenir à une classe s'il est proche de cette classe au sens de la mesure de ressemblance. Il semble donc naturel d'affecter l'élement à la classe la plus proche si on hésite entre plusieurs classes. Les éléments appartenant à une même classe sont dits semblables. Lorsque l'espace euclidien $\mathbb{R}^d$ est considéré, une classe est souvent définie par son centre. Une mesure de ressemblance entre tout élément de $\mathbb{R}^d$ et la classe peut alors être la distance euclidienne entre cet élément et le centre de la classe. On affecterait tout élément à la classe dont le centre est le plus proche. La classe gaussienne est une extension de la définition précédente. Dans ce cas, une classe est représentée par une loi normale multivariée (équation 1.1). Elle est caractérisée par un *centre* ($\boldsymbol{\mu}$) et une *dispersion* autour de ce centre (la matrice de variance $\boldsymbol{\Sigma}$). Une mesure de ressemblance entre la classe gaussienne et tout élément de $\mathbb{R}^d$ peut être la distance de Mahalanobis $\sqrt{(\boldsymbol{x}-\boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}$.

Dans ce contexte gaussien, on ajoute généralement une quantité à chacune des classes. Il s'agit de la *proportion* p_k ($1 \leq k \leq K$, si K classes) qui représente le taux du nombre d'éléments qui appartiennent à la classe k. Implicitement plusieurs classes sont représentées par un mélange gaussien et il y a autant de classes que de composantes dans le mélange. Cette approche permet de s'affranchir de la notion de ressemblance entre un élément et une classe en considérant ce problème sous un angle probabiliste. Par exemple, affecter un élément à une classe parmi plusieurs classes en compétition revient soit à choisir la classe la plus proche par une mesure de ressemblance réaliste, à trouver, soit à choisir simplement la classe la plus probable pour cet élément (nous y reviendrons). L'utilisation de la théorie des probabilités nous ouvre alors un horizon plus large.

Les données

Les données considérées dans le travail correspondent à un échantillon de n éléments (les individus) sur lesquelles on dispose de d variables réelles et d'une variable qualitative. Cette variable qualitative a un nombre fini K de modalités notées 1 à K. Chaque modalité correspond à une caractéristique différente des individus. Par exemple, la modalité 1 représente la classe des individus sains, la modalité 2 étant celle des individus malades. La variable qualitative définit une partition des individus en K classes, c'est-à-dire que chaque individu appartient à une et une seule classe (donc n'a qu'une seule modalité) et aucune classe n'est vide (toutes les modalités sont représentées au moins par un individu). Les données sont généralement disposées dans un tableau individus–variables.

Exemple Pour illustrer les propos précédents et introduire certaines notations importantes que nous employerons par la suite, nous présentons dans le tableau 1.1 un exemple de tableau individus–variables. Il s'agit de 22 avions légers (les individus) auquels sont associés deux variables quantitatives réelles, le poids et la vitesse de l'appareil. La variable qualitative possède trois modalités, chacune correspondant au type de propulsion qu'il utilise : 1–pour turboréacteurs, 2–pour turbopropulseurs et 3–pour moteurs à pistons.

Nous notons $\boldsymbol{x}_i$ un vecteur à deux composantes. La première correspond au poids de l'avion n°i, et la deuxième à sa vitesse. Nous notons aussi z_i un scalaire dont la valeur est égale à la modalité du type de propulsion de cet avion. Par exemple, nous avons

$$\boldsymbol{x}_1 = \begin{pmatrix} 3870 \\ 760 \end{pmatrix} \qquad \text{et} \qquad z_1 = 1.$$

L'ensemble des vecteurs $\boldsymbol{x}_i$ $(1 \leq i \leq 22)$ est rassemblé dans l'ensemble $\mathcal{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{22})$ et l'ensemble des scalaires est groupé dans le vecteur $\boldsymbol{z} = (z_1, \ldots, z_{22})$. Le tableau de données est alors résumé par le couple $(\mathcal{X}, \boldsymbol{z})$. Les z_i seront souvent appelés *les labels* des éléments $\boldsymbol{x}_i$.

Le vecteur $\boldsymbol{z}$ définit une partition de l'ensemble des 22 individus en trois classes : la classe des turboréacteurs (classe 1), la classe des turbopropulseurs (classe 2) et la classe des moteurs à pistons (classe 3). La partition est donnée par la réunion des trois classes et chaque classe contient les éléments suivants :

$$\text{classe } 1 = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_7\}, \quad \text{classe } 2 = \{\boldsymbol{x}_8, \ldots, \boldsymbol{x}_{12}\}, \quad \text{classe } 3 = \{\boldsymbol{x}_{13}, \ldots, \boldsymbol{x}_{22}\}.$$

1.3.2 Discrimination

Position du problème

La discrimination est une méthode de classification supervisée. On l'appelle aussi classification «avec professeur» ou bien analyse discriminante. Seule la discrimination exclusive est considérée.

Nous disposons d'un ensemble de n individus $\mathcal{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ et d'une partition de cet ensemble représentée par un ensemble de n labels $\boldsymbol{z} = (z_1, \ldots, z_n)$. Les individus $\boldsymbol{x}_i$ $(1 \leq i \leq n)$ sont dans l'espace $\mathbb{R}^d$ et chaque label z_i prend une valeur dans $\{1, \ldots, K\}$. Chaque label z_i correspond au numéro de la classe d'appartenance du point $\boldsymbol{x}_i$, parmi les K classes possibles. L'objectif de la discrimination est de *prédire* la classe de tout nouveau point $\boldsymbol{x}$ de $\mathbb{R}^d$ à partir de cet ensemble d'apprentissage $(\mathcal{X}, \boldsymbol{z})$. Dans l'exemple précédent des avions, il s'agirait de trouver la modalité z_{23} d'un 23$^{\text{e}}$ avion lorsque seul le vecteur $\boldsymbol{x}_{23}$ est connu. En d'autres termes, il s'agit d'estimer une règle de classement $\hat{u}$:

$$\begin{array}{cccc} \hat{u} : & \mathbb{R}^d & \longrightarrow & \{1, \ldots, K\} \\ & \boldsymbol{x} & \longmapsto & \hat{u}(\boldsymbol{x}) \end{array}. \tag{1.4}$$

Cette règle $\hat{u}$ produit une partition de l'espace $\mathbb{R}^d$ (figure 1.3).

Le modèle de mélange gaussien est alors très utilisé dans ce contexte (McLachlan [69]). Une classe est caractérisée par une distribution gaussienne, c'est-à-dire

n°	Nom	Poids (kg)	Vitesse (km/h)	Type de propulsion
1	500 Citation Cessna	3870	760	1
2	Gates Learjet 24E	3180	943	1
3	Gates Learjet 36A	4140	943	1
4	Gulfstream 2 Gramman	3670	980	1
5	HFB 320 Hansa	5420	930	1
6	Jet Star 2 Lockheed-Ga	10950	930	1
7	Sabreliner 600 Rockwell	5090	920	1
8	B99 Airliner Beech	2610	450	2
9	B100 King-Air Beech	3230	495	2
10	Cheyenne PA31	2200	515	2
11	Mitsubishi MU 2L	3430	510	2
12	Turbo commander 690A	2820	515	2
13	B55 Baron Beech	1430	360	3
14	Bonanza A36 Beech	950	315	3
15	402B Businessliner Cessna	1950	380	3
16	177 Cardinal Cessna	690	240	3
17	PA Cherokee Piper	570	220	3
18	Citabria 7KCab	515	210	3
19	PA Navajo Piper	1980	400	3
20	207 Skywagon Cessna	890	260	3
21	Super Viking 300A	1000	300	3
22	Trainer Grummun	440	220	3

source : quid 1977

Légende du type de propulsion :

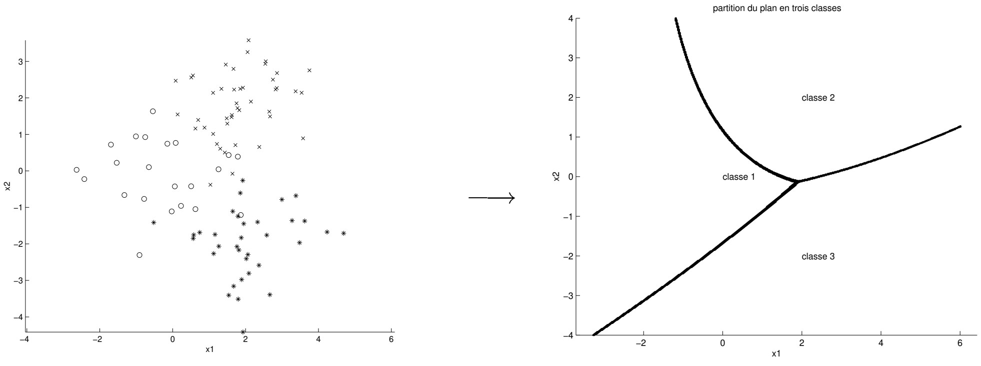
1 – turboréacteurs, 2 – turbopropulseurs, 3 – moteurs à pistons

TAB. 1.1 – *Caractéristiques de quelques avions légers.*

un centre et une dispersion autour de ce centre. On suppose que chacun des n couples $(\boldsymbol{x}_i, z_i)$ est une réalisation i.i.d. de la loi jointe

$$f(\boldsymbol{x}, k|\boldsymbol{\theta}) = p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k).$$

On suppose donc que les éléments de la classe k proviennent de manière indépendante de la loi de densité $\phi_k(\boldsymbol{x}|\boldsymbol{a}_k)$ et que la classe k a une probabilité a priori p_k de se produire. Si le paramètre $\boldsymbol{\theta}$ du mélange était connu alors chaque nouvel individu $\boldsymbol{x}$ de $\mathbb{R}^d$ serait classé par la méthode de MAP que nous présentons maintenant.



données $(\mathcal{X}, \boldsymbol{z})$ règle $\hat{u}$

FIG. 1.3 – *Principe de la discrimination.*

Classement par MAP

Connaissant le paramètre $\boldsymbol{\theta}$, l'affectation de tout élément $\boldsymbol{x}$ de $\mathbb{R}^d$ à une classe k ($k \in \{1, \ldots, K\}$) peut être réalisée en utilisant la méthode du *maximum a posteriori*, dite aussi méthode du MAP. Cette méthode consiste simplement à affecter l'élément $\boldsymbol{x}$ à la classe la plus probable a posteriori (figure 1.4). La probabilité a posteriori que $\boldsymbol{x}$ appartienne à la classe k se calcule directement par le théorème de Bayes :

$$\begin{aligned} t_k(\boldsymbol{x}|\boldsymbol{\theta}) &= \mathrm{P}(Z = k|\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{\theta}) \\ &= \frac{p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k)}{f(\boldsymbol{x}|\boldsymbol{\theta})}. \end{aligned} \tag{1.5}$$

Le MAP semble être une méthode de simple «bon sens», mais puise en réalité sa justification en théorie bayésienne de la décision. Soit une règle de classement u et soit la fonction $\mathcal{L}$ de coût 0–1 associant un coût nul aux bons classements et 1 aux mauvais :

$$\begin{cases} \mathcal{L}(i|i) = 0 & i = 1, \ldots, K \\ \mathcal{L}(i|j) = 1 & i, j = 1, \ldots, K\ ,\ i \neq j \end{cases}.$$

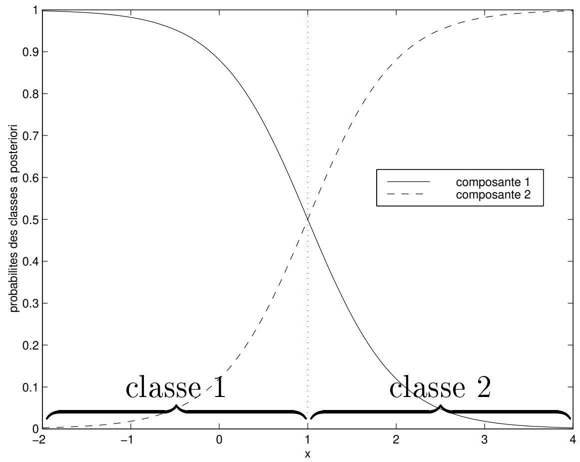


FIG. 1.4 – *Principe du MAP dans le cas réel avec deux classes.*

Le risque conditionnel s'écrit

$$\begin{aligned}\mathcal{R}(u|\boldsymbol{x}) &= \sum_{k=1}^{K} \mathcal{L}(u(\boldsymbol{x})|k) t_k(\boldsymbol{x}|\boldsymbol{\theta}) \qquad (1.6)\\ &= \sum_{k=1,k\neq u(x)}^{K} t_k(\boldsymbol{x}|\boldsymbol{\theta})\\ &= 1 - t_{u(x)}(\boldsymbol{x}|\boldsymbol{\theta}).\end{aligned}$$

Moyennant sur tous les $\boldsymbol{x}$, on obtient le risque moyen qui s'interprète comme une probabilité d'erreur de classement P_e

$$\begin{aligned}\mathrm{P}_e(u) &= \mathrm{E}_X[\mathcal{R}(u|\boldsymbol{X})] \qquad (1.7)\\ &= 1 - \mathrm{E}_X[t_{u(X)}(\boldsymbol{X}|\boldsymbol{\theta})].\end{aligned}$$

La règle de classement optimale u_θ, ou règle de Bayes, est celle qui minimise l'erreur de classement P_e. Pour ce faire, il suffit de minimiser le risque conditionnel $\mathcal{R}(u|\boldsymbol{x})$ pour chaque $\boldsymbol{x}$. D'où la méthode du MAP

$$\forall \boldsymbol{x} \in \mathbb{R}^d \ , \ \ u_\theta(\boldsymbol{x}) = \arg\min_k t_k(\boldsymbol{x}|\boldsymbol{\theta}).$$

Généralement le vrai paramètre $\boldsymbol{\theta}$ est inconnu et on ne dispose donc pas des probabilités a posteriori $t_k(\boldsymbol{x}|\boldsymbol{\theta})$ pour calculer la regle optimale u_θ. La méthode du plug-in et la méthode prédictive sont deux principes couramment utilisés pour estimer les probabilités a posteriori $t_k(\boldsymbol{x}|\boldsymbol{\theta})$ et ainsi déduire un estimateur de u_θ. Nous présentons ces deux méthodes, mais dans la suite du travail nous nous restreindrons à la méthode du plug-in.

Méthode du plug-in

La méthode du plug-in (McLachlan [69]) est certainement la plus utilisée. Elle consiste à estimer le paramètre de mélange $\boldsymbol{\theta}$ par un estimateur $\hat{\boldsymbol{\theta}}$. Cette estimation se fait à partir des données $(\mathcal{X}, \boldsymbol{z})$ et non pas à partir de $\mathcal{X}$ seulement comme dans l'estimation du paramètre du mélange présentée précedemment. On verra le principe de cette estimation au chapitre suivant. Ensuite, connaissant $\hat{\boldsymbol{\theta}}$, on estime la vraie probabilité a posteriori par

$$\hat{t}_k(\boldsymbol{x}|\boldsymbol{\theta}) = t_k(\boldsymbol{x}|\hat{\boldsymbol{\theta}}).$$

On remplace donc $\boldsymbol{\theta}$ par sa valeur estimée $\hat{\boldsymbol{\theta}}$ (principe du «plug-in») ; on déduit alors, par la méthode du MAP, une estimation de la règle de classement :

$$\hat{u}_\theta = u_{\hat{\theta}}.$$

Méthode prédictive

La méthode prédictive a pour but d'estimer les probabilités a posteriori $t_k(\boldsymbol{x}|\boldsymbol{\theta})$, mais cette fois sans estimer directement le paramètre $\boldsymbol{\theta}$. Cette méthode s'inscrit dans une démarche bayésienne (Ripley p. 45–55 [77]). On estime alors les probabilités a posteriori $t_k(\boldsymbol{x}|\boldsymbol{\theta})$, non plus en estimant $\boldsymbol{\theta}$, mais en moyennant sur l'espace Θ des $\boldsymbol{\theta}$. Il est donc nécessaire de définir une loi a priori $\pi(\boldsymbol{\theta})$ sur $\boldsymbol{\theta}$. On a

$$\begin{aligned} \hat{t}_k(\boldsymbol{x}) &= \mathrm{P}(k|\boldsymbol{x}, \mathcal{X}, \boldsymbol{z}) \\ &\propto \int_\Theta t_k(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}|\boldsymbol{x}, \mathcal{X}, \boldsymbol{z})d\boldsymbol{\theta}, \end{aligned}$$

avec

$$\pi(\boldsymbol{\theta}|\boldsymbol{x}, \mathcal{X}, \boldsymbol{z}) \propto f(\boldsymbol{x}|\boldsymbol{\theta})f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}).$$

Les couples $(\boldsymbol{x}_i, z_i)$ étant des réalisations de couples de variables aléatoires $(\boldsymbol{X}_i, Z_i)$ indépendants entre eux, la loi jointe de $(\mathcal{X}, \boldsymbol{z})$ s'exprime par

$$f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta}) = \prod_{i=1}^{n} f(\boldsymbol{x}_i, z_i|\boldsymbol{\theta}). \tag{1.8}$$

Une fois obtenues les probabilités $\mathrm{P}(k|\boldsymbol{x}, \mathcal{X}, \boldsymbol{z})$, on estime la règle de Bayes par la méthode du MAP. Nous discuterons du choix de la loi a priori $\pi(\boldsymbol{\theta})$ lors de l'estimation bayésienne en mélange au chapitre suivant. Généralement, les approches du plug-in et de la prédiction sont *asymptotiquement* (la taille de l'échantillon tend vers l'infini) équivalentes.

1.3.3 Classification automatique

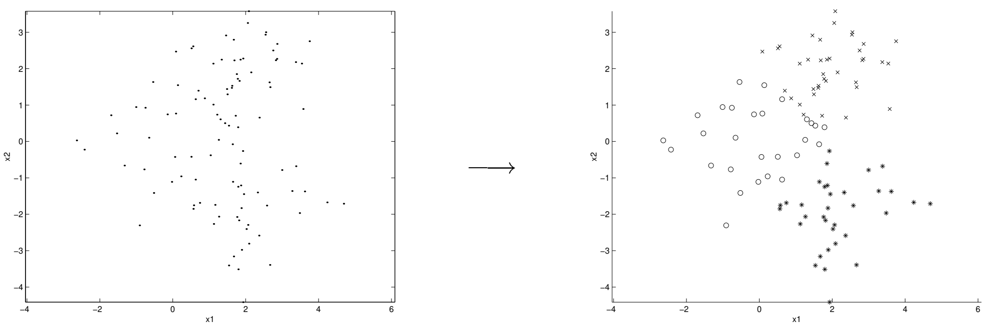
Position du problème

La classification automatique, ou clustering, est une méthode de classification non supervisée. Nous ne considérons que la classification automatique exclusive, c'est-à-dire que nous groupons les données en partition. Nous n'étudions donc pas des structures hiérachiques (suite de partitions emboîtées), ni des appartenances floues (classification non exclusive).

Nous disposons d'un ensemble de n données $\mathcal{X} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$ de $\mathbb{R}^d$ et l'objectif est de trouver une partition en K classes de cet ensemble. Il s'agit d'estimer les labels $z_i \in \{1, \dots, K\}$ de chaque point $\boldsymbol{x}_i$ donc d'obtenir une partition estimée $\hat{\boldsymbol{z}} = (\hat{z}_1, \dots, \hat{z}_n)$ de $\mathcal{X}$. On constatera la différence fondamentale entre discrimination et classification automatique. Dans le cas présent, on ne cherche plus à *prédire* l'appartenance de tout point de $\mathbb{R}^d$, mais seulement à *décrire* l'appartenance des $\boldsymbol{x}$ de $\mathcal{X}$ (figure 1.5).

Dans l'exemple des avions légers, on ne dispose que des données concernant le poids et la vitesse de chacun des 22 appareils. Le type de propulsion de chaque appareil est inconnu. L'objectif de la classification automatique est alors de retrouver les trois classes donc les labels $\boldsymbol{z}$. Un expert du domaine interprétera ensuite les classes comme correspondant au type de propulsion employé par chaque avion.

Le modèle de mélange gaussien sert souvent de base pour estimer $\hat{\boldsymbol{z}}$ (Wolfe [91], Bock [20]). On suppose que chaque $\boldsymbol{x}_i$ $(1 \leq i \leq n)$ est une réalisation i.i.d. de la loi mélange $f(\boldsymbol{x}|\boldsymbol{\theta})$. On peut aussi voir chaque couple $(\boldsymbol{x}_i, z_i)$ comme une réalisation i.i.d. de la loi jointe $f(\boldsymbol{x}, k|\boldsymbol{\theta})$. Dans ce cas on dit que les z_i sont inobservables car inconnus. Il existe alors plusieurs façons d'utiliser l'hypothèse de mélange pour estimer $\boldsymbol{z}$. Nous présenterons trois principes importants : la méthode mélange, la méthode classification et la méthode mixte. Seules les méthodes mélange et mixte seront ensuite considérées dans notre travail.



données $\mathcal{X}$ partition $\hat{\boldsymbol{z}}$

FIG. 1.5 – *Principe de la classification automatique.*

Approche mélange

L'approche mélange consiste à se placer dans le contexte de l'estimation du paramètre $\boldsymbol{\theta}$ à partir de $\mathcal{X}$. On obtient ainsi un estimateur $\hat{\boldsymbol{\theta}}$. Ensuite, on applique la méthode du plug-in pour estimer les $n \times K$ probabilités a posteriori t_{ik} ($i = 1, \ldots, K$, $k = 1, \ldots, K$) avec la notation

$$t_{ik} = t_k(\boldsymbol{x}_i|\boldsymbol{\theta}) \tag{1.9}$$

et on utilise la méthode du MAP pour déduire la partition $\hat{\boldsymbol{z}}$.

Approche classification

L'approche classification, souvent appelée classification bayésienne (Binder [18]), propose d'estimer $\boldsymbol{z}$ directement à partir de $\mathcal{X}$, sans estimation de $\boldsymbol{\theta}$. Comme la méthode prédictive en discrimination, on se fixe une loi a priori $\pi(\boldsymbol{\theta})$ sur le paramètre et on moyenne sur l'espace Θ. La probabilité a posteriori de la partition $\boldsymbol{z}$ dans le cadre bayésien s'écrit

$$\mathrm{P}(\boldsymbol{z}|\mathcal{X}) \propto \int_{\Theta} f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta}$$

avec $f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta})$ densité jointe de $(\mathcal{X}, \boldsymbol{z})$ donnée par l'équation 1.8.

Le but étant de prendre une décision $\hat{\boldsymbol{z}}$ au sujet de $\boldsymbol{z}$, on se munit d'une fonction coût $\mathcal{L}(\hat{\boldsymbol{z}}|\boldsymbol{z})$ et on choisit la partition $\hat{\boldsymbol{z}}$ qui minimise le risque de Bayes conditionnel

$$\mathcal{R}(\hat{\boldsymbol{z}}|\mathcal{X}) = \sum_{(\boldsymbol{z})} \mathcal{L}(\hat{\boldsymbol{z}}|\boldsymbol{z})\mathrm{P}(\boldsymbol{z}|\mathcal{X})$$

avec $(\boldsymbol{z})$ l'ensemble des partitions de $\mathcal{X}$ en K classes. Le coût 0–1 est très simple et assez naturel dans ce contexte (d'autres coûts plus sophistiqués sont aussi possibles). Rappelons qu'on donne la valeur 1 si $\hat{\boldsymbol{z}} = \boldsymbol{z}$ et 0 sinon. La règle de Bayes associée à ce coût est celle qui maximise la probabilité de la partition a posteriori $\mathrm{P}(\boldsymbol{z}|\mathcal{X})$.

Approche mixte

Enfin, l'approche mixte cherche à faire une estimation *simultanée* du couple de paramètres $(\boldsymbol{\theta}, \boldsymbol{z})$. Il est vrai que l'approche mélange permet aussi d'obtenir un estimateur $(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{z}})$, mais elle obtient en réalité premièrement un estimateur $\hat{\boldsymbol{\theta}}$ (en moyennant sur la partition $\boldsymbol{z}$ avec distribution a priori obtenue à partir des probabilités $p_1, \ldots, p_K$) pour seulement ensuite en déduire un estimateur $\hat{\boldsymbol{z}}$ par plug-in et MAP. De même avec l'approche classification, une fois obtenu un estimateur $\hat{\boldsymbol{z}}$ (en moyennant sur le paramètre $\boldsymbol{\theta}$ avec loi a priori $\pi(\boldsymbol{\theta})$), on pourrait obtenir un estimateur $\hat{\boldsymbol{\theta}}$. Dans l'approche mixte, nous estimons conjointement les deux paramètres, sans moyenner ni sur la partition, ni sur le paramètre du mélange.

Remarquons que cette approche mixte a aussi été qualifiée d'approche classification par Celeux et Govaert [38], pour marquer la différence avec l'approche mélange.

1.4 Modèles gaussiens

Les modèles gaussiens correspondent à un ensemble de contraintes sur les matrices de variance ou sur les proportions. Nous présentons 28 modèles différents en utilisant une décomposition spectrale des matrices de variance.

1.4.1 A priori sur le paramètre du mélange

Principe

Tout utilisateur, qu'il soit dans un cadre de discrimination ou de classification automatique via un modèle gaussien de mélange, dispose parfois d'une *connaissance a priori* sur le paramètre du mélange. Les connaissances a priori les plus classiques portent sur les proportions du mélange ou sur les matrices de variance. Pour les proportions, l'utilisateur peut savoir que les proportions sont égales. Pour les matrices de variance, l'utilisateur dispose généralement d'un choix entre quatre types de contraintes (Hartigan [56], Bock [19], Dubuisson [45], Bozdogan [26] par exemple)

- **modèle linéaire sphérique** : $\boldsymbol{\Sigma}_1 = \cdots = \boldsymbol{\Sigma}_K = \sigma^2 \boldsymbol{I}$ avec σ^2 inconnu et $\boldsymbol{I}$ matrice identité $d \times d$;

- **modèle linéaire diagonal** : $\boldsymbol{\Sigma}_1 = \cdots = \boldsymbol{\Sigma}_K = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_d^2)$ avec $(\sigma_1^2, \ldots, \sigma_d^2)$ inconnus et $\mathrm{diag}(\sigma_1^2, \ldots, \sigma_d^2)$ matrice diagonale d'éléments diagonaux $(\sigma_1^2, \ldots, \sigma_d^2)$;

- **modèle linéaire général** : $\boldsymbol{\Sigma}_1 = \cdots = \boldsymbol{\Sigma}_K = \boldsymbol{\Sigma}$ avec $\boldsymbol{\Sigma}$ une matrice symétrique définie positive ;

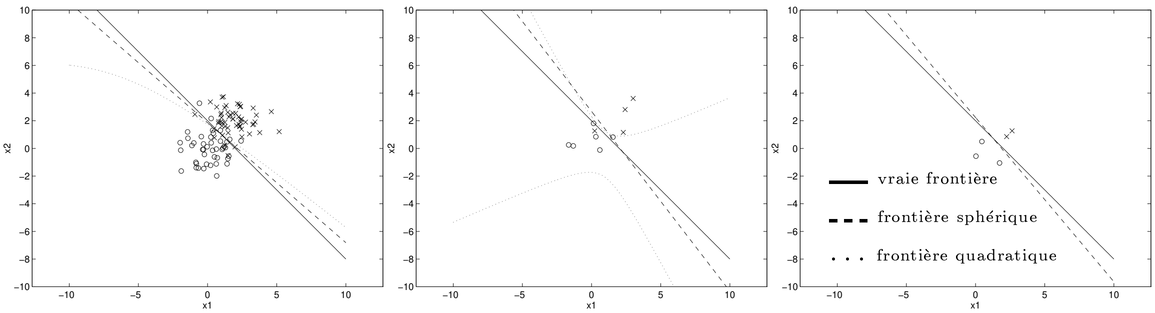
- **modèle quadratique** : aucune contrainte sur les $\boldsymbol{\Sigma}_k$.

Dans les méthodes bayésiennes (l'approche prédiction en discrimination et l'approche classification en classification automatique) la notion d'information a priori a déjà été introduite par la loi a priori $\pi(\boldsymbol{\theta})$. Tout a priori se traduira donc au travers de cette loi. Nous verrons au chapitre suivant comment tenir compte de telles informations pour les autres méthodes : plug-in, mélange et mixte.

Intérêt

L'avantage d'introduire de l'information a priori de l'utilisateur est de mieux estimer la quantité d'intérêt, c'est-à-dire le paramètre de mélange $\boldsymbol{\theta}$ en estimation mélange, la règle u en discrimination et la partition $\boldsymbol{z}$ en classification automatique. Il s'agit aussi de protéger le modèle de mélange contre une surparamétrisation. Ce phénomène de surparamétrisation intervient quand le nombre de paramètres libres est trop grand pour le nombre n de données.

La figure 1.6 illustre l'influence du modèle et de la taille de l'échantillon sur la frontière de discrimination calculée par la méthode du plug-in : quand la taille de l'échantillon diminue, on constate que la frontière quadratique diverge plus vite que la frontière linéaire de la vraie frontière (cas (a) et (b)). Le cas (c) avec seulement 5 points ne fournit pas de frontière quadratique, la solution étant singulière. L'exemple suivant étudie, à l'aide de simulations, l'influence du modèle et de la taille de l'échantillon, non seulement en discrimination, mais aussi en classification automatique et en estimation mélange.



(a) : $n = 100$ (b) : $n = 10$ (c) : $n = 5$

FIG. 1.6 – *Influence de la taille d'échantillon sur l'estimation des frontières linéaire sphérique et quadratique (les proportions du mélange sont égales).*

Exemple Étudions le comportement des trois quantités d'intérêt $\boldsymbol{\theta}$, u et $\boldsymbol{z}$ sur des données simulées. Soit un mélange bivarié de deux composantes avec les caractéristiques suivantes :

$$p_1 = p_2 = 0.5, \qquad \boldsymbol{\mu}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad \boldsymbol{\mu}_2 = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \qquad \boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \boldsymbol{I}.$$

Nous simulons 30 échantillons de 200 points et 30 échantillons de 40 points. Notons $\mathcal{X}$ l'ensemble des points de $\mathbb{R}^d$ générés et $\boldsymbol{z}$ les labels de ces points.

En estimation par modèle de mélange et en classification automatique, l'échantillon se limite à $\mathcal{X}$, les labels z_i étant ignorés. Chaque échantillon fournit deux estimateurs du paramètre mélange : $\hat{\boldsymbol{\theta}}_{sphe}$ pour le cas linéaire sphérique et $\hat{\boldsymbol{\theta}}_{quad}$ pour le cas quadratique. Les proportions ne sont pas estimées car on suppose savoir qu'elles sont égales. On obtient aussi deux estimateurs de la partition par la méthode mélange ($\hat{\boldsymbol{z}}^{mel}_{sphe}$ et $\hat{\boldsymbol{z}}^{mel}_{quad}$) et deux estimateurs de la partition par la méthode mixte ($\hat{\boldsymbol{z}}^{mix}_{sphe}$ et $\hat{\boldsymbol{z}}^{mix}_{quad}$). En discrimination, l'échantillon devient le couple $(\mathcal{X}, \boldsymbol{z})$, les labels étant inclus. On obtient deux estimateurs de la règle u par la méthode du plug-in : $\hat{u}_{sphe}$ et $\hat{u}_{quad}$.

La performance de chaque estimateur est alors mesurée. Pour le paramètre du mélange, on utilise une estimation de la divergence de Kullback-Leibler entre

$\boldsymbol{\theta}$ et $\hat{\boldsymbol{\theta}}$:

$$\mathrm{KL}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}) = \int \ln \left[\frac{f(\boldsymbol{x}|\boldsymbol{\theta})}{f(\boldsymbol{x}|\hat{\boldsymbol{\theta}})} \right] f(\boldsymbol{x}|\boldsymbol{\theta}) d\boldsymbol{x}. \tag{1.10}$$

Pour la partition $\hat{\boldsymbol{z}}$, on utilise l'erreur de classement (pourcentage de points mal classés)

$$\mathrm{erreur}(\hat{\boldsymbol{z}}) = \frac{\mathrm{card}\{\boldsymbol{x}_i / \hat{z}_i \neq z_i\}}{n}. \tag{1.11}$$

La mesure de la performance de $\hat{u}$ est mesurée par une estimation de la probabilité d'erreur de classement $\mathrm{P}_e(\hat{u})$ (équation 1.7).

Le tableau 1.2 donne les valeurs des moyennes de ces mesures de performance, pour les différentes tailles d'échantillon et les différents modèles. On remarque que la performance de tous les estimateurs précédents est meilleure avec le modèle sphérique qu'avec le modèle quadratique. Le modèle sous-jacent étant sphérique, il semble donc avantageux de traduire cette information a priori. On remarque aussi que l'augmentation de la taille de l'échantillon semble atténuer l'effet du modèle. Introduire de l'information a priori est donc surtout nécessaire pour de petits échantillons.

n	modèle	$\widehat{\mathrm{KL}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$	$\mathrm{erreur}(\hat{\boldsymbol{z}}^{mel})$	$\mathrm{erreur}(\hat{\boldsymbol{z}}^{mix})$	$\hat{\mathrm{P}}_e(\hat{u})$
40	sphérique	0.0760	0.0967	0.0892	0.0812
	quadratique	0.1929	0.1100	0.0925	0.0874
100	sphérique	0.0116	0.0840	0.0830	0.0777
	quadratique	0.0245	0.0872	0.0832	0.0786

TAB. 1.2 – *Effet du modèle et de la taille de l'échantillon sur les estimateurs.*

1.4.2 Décomposition des matrices de variance

Principe de la décomposition

Dans le contexte de la classification automatique, Banfield et Raftery [7] proposent de généraliser le type précédent de contraintes sur les matrices de variance. Ils utilisent la décomposition spectrale suivante de la matrice de variance de la classe k

$$\boldsymbol{\Sigma}_k = \lambda_k \boldsymbol{D}_k \boldsymbol{A}_k \boldsymbol{D}_k', \tag{1.12}$$

avec

- λ_k : valeur propre la plus grande de Σ_k ($\lambda_k \in \mathbb{R}_+^*$) ;
- $\boldsymbol{D}_k$: matrice orthogonale des vecteurs propres de $\boldsymbol{\Sigma}_k$;

- $\boldsymbol{A}_k = \text{diag}\{a_{1k}, \ldots, a_{dk}\}$: matrice des valeurs propres normalisées avec $1 = a_{1k} \geq a_{2k} \geq \cdots \geq a_{dk} > 0$.

La décomposition ainsi définie est unique. L'avantage essentiel de la démarche est la facilité d'interprétation de ces différents termes d'un point de vue géométrique donc intuitif. Le paramètre λ_k s'interprète comme le volume de la classe k, $\boldsymbol{D}_k$ comme son orientation et $\boldsymbol{A}_k$ comme sa forme. Il ne faut d'ailleurs pas confondre le volume avec la proportion des classes qui, elle, est représentée par p_k : le volume correspond à l'espace occupé par la classe tandis que la proportion est relative au nombre de points de la classe. Il y a bien indépendance des deux termes, une classe pouvant occuper un volume important mais avec une proportion de points très faible (classe de faible densité). Cette décomposition fut aussi utilisée dans un contexte de discrimination par Flury et al. [48].

Dans notre travail, nous considérons une autre décomposition légèrement différente, mais aussi unique, et introduite par Celeux et Govaert [38] à la suite des travaux de Banfield et Raftery :

- $\lambda_k = |\boldsymbol{\Sigma}_k|^{1/d}$ ($\lambda_k \in \mathbb{R}_+^*$) ;
- $\boldsymbol{D}_k$: définition inchangée ;
- $\boldsymbol{A}_k = \text{diag}\{a_{1k}, \ldots, a_{dk}\}$: matrice des valeurs propres normalisées avec les a_{ik} ($1 \leq i \leq d$) classés par ordre décroissant sur la diagonale et $|\boldsymbol{A}_k| = 1$.

L'interprétation des termes reste strictement identique à la précédente mais elle permet de faciliter certains calculs qui interviendront pour l'estimation.

Celeux et Govaert obtiennent 14 modèles, du plus parcimonieux au moins parcimonieux, en permettant à certains paramètres, et non nécessairement à tous, de varier ou non entre les classes. Le modèle imposant un même volume entre toutes les classes est noté $[\lambda]$, le modèle permettant des volumes différents entre classes est noté $[\lambda_k]$. En adoptant la même notation pour les autres paramètres ils notent de manière synthétique $[\lambda_k D_k A D_k']$ un modèle à volume et orientation libres entre les classes mais à forme identique. À partir de maintenant et jusqu'à la fin de la thèse, nous appelons ces 14 modèles *modèles gaussiens*.

Les 14 modèles peuvent être regroupés en 3 grandes familles (voir le tableau 1.3 pour le détail des modèles) :

- **famille sphérique** : la matrice forme est l'identité ($\boldsymbol{A} = \boldsymbol{I}$) et donc la matrice orientation $\boldsymbol{D}$ ne joue aucun rôle. En dimension 2, l'ellipse de concentration est un cercle ;
- **famille diagonale** : $\boldsymbol{B} = \boldsymbol{D}\boldsymbol{A}\boldsymbol{D}'$ est diagonale (donc $\boldsymbol{\Sigma}$ aussi) avec $\boldsymbol{D}$ matrice de permutation de la base canonique. En dimension 2, l'ellipse de concentration est orientée suivant les axes ;
- **famille générale** : elle regroupe tous les autres modèles. On notera $\boldsymbol{C} = \boldsymbol{D}\boldsymbol{A}\boldsymbol{D}'$.

Famille	Modèles	Nombre de paramètres		
		Cas général	$K = 2$ et $d = 2$	
			p_k égales	p_k libres
Sphérique	$[\lambda I]$	$\alpha + 1$	5	6
	$[\lambda_k I]$	$\alpha + K$	6	7
Diagonale	$[\lambda B]$	$\alpha + d$	6	7
	$[\lambda_k B]$	$\alpha + d + K - 1$	7	8
	$[\lambda B_k]$	$\alpha + Kd - (K-1)$	7	8
	$[\lambda_k B_k]$	$\alpha + Kd$	8	9
Générale	$[\lambda C]$	$\alpha + \beta$	7	8
	$[\lambda_k C]$	$\alpha + \beta + K - 1$	8	9
	$[\lambda D A_k D']$	$\alpha + \beta + (K-1)(d-1)$	8	9
	$[\lambda_k D A_k D']$	$\alpha + \beta + (K-1)d$	9	10
	$[\lambda D_k A D_k']$	$\alpha + K\beta - (K-1)d$	8	9
	$[\lambda_k D_k A D_k']$	$\alpha + K\beta - (K-1)(d-1)$	9	10
	$[\lambda C_k]$	$\alpha + K\beta - (K-1)$	9	10
	$[\lambda_k C_k]$	$\alpha + K\beta$	10	11

TAB. 1.3 – *Liste des modèles par famille. Nous avons $\alpha = Kd + K - 1$ si les proportions sont libres et $\alpha = Kd$ sinon. Nous notons aussi $\beta = (d(d+1)/2)$.*

Illustration géométrique de la décomposition

Pour illustrer la signification géométrique forte qui existe au travers de la décomposition des matrices de variance, plaçons nous dans le plan. $\boldsymbol{D}_k$ est alors la matrice de rotation habituelle. On peut exprimer les matrices $\boldsymbol{A}_k$ et $\boldsymbol{D}_k$ de la manière suivante, α_k étant l'angle de rotation :

$$\boldsymbol{A}_k = \begin{pmatrix} a_k & 0 \\ 0 & 1/a_k \end{pmatrix}, \quad \boldsymbol{D}_k = \begin{pmatrix} \cos\alpha_k & -\sin\alpha_k \\ \sin\alpha_k & \cos\alpha_k \end{pmatrix}. \tag{1.13}$$

L'*ellipse de concentration* (ou de variance) de la classe k associée à cette matrice de variance est représentée sur la figure 1.7. Cette ellipse a pour centre $\boldsymbol{\mu}_k$ et contient ξ % des points que génère la loi $\phi_k(\boldsymbol{x}|\boldsymbol{a}_k)$. En pratique, tracer l'ellipse de concentration de la classe k revient à dessiner la courbe d'équation

$$\sqrt{(\boldsymbol{x} - \boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_k)} = \frac{\xi}{100}.$$

Cette courbe est composée des points équidistants du centre $\boldsymbol{\mu}_k$ au sens de la distance de Mahalanobis.

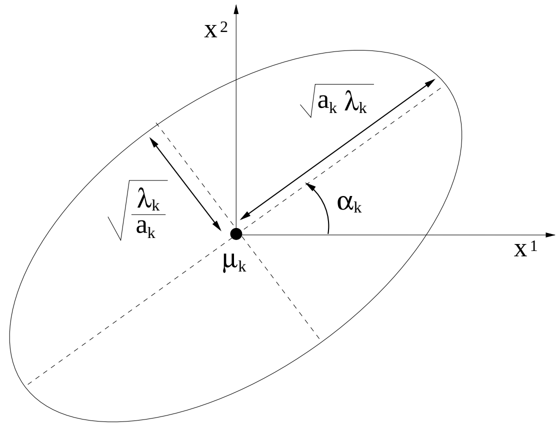


FIG. 1.7 – *Interprétation géométrique de la décomposition spectrale.*

Rôle particulier du volume

Biernacki [12] et Celeux et Govaert [38] ont montré l'intérêt tout particulier qu'apporte le volume λ. Laisser le volume libre entre classes (modèles $[p_{(k)}\lambda_k \ldots]$) n'augmente que faiblement le nombre de paramètres total du modèle de mélange : cette augmentation est indépendante de la dimension d de l'espace et ne dépend que linéairement du nombre de classes. Malgré ce faible coût en terme d'accroissement du nombre de paramètres libres, l'utilisation de modèles à volumes libres permet d'augmenter significativement le pouvoir discriminateur du modèle à volumes égaux. Par exemple, pour deux classes, le modèle $[\lambda I]$ ne fournit qu'une règle de discrimination linéaire tandis que le modèle $[\lambda_k I]$ fournit une règle sphérique (figure 1.8). Le coût à payer n'est alors que d'un seul nouveau paramètre (voir le tableau 1.3).

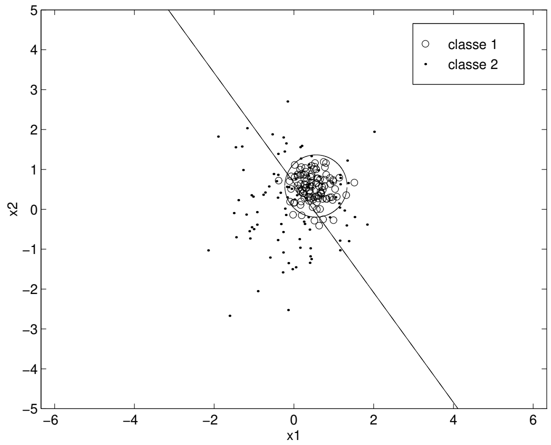


FIG. 1.8 – *Frontière linéaire (modèle $[p\lambda I]$) et sphérique (modèle $[p\lambda_k I]$).*

Approche concurrente par régularisation

Dans un cadre de discrimination, Friedman [50] proposa une méthode de régularisation (*regularized discriminant analysis* aussi appelée RDA) qui réalise un compromis entre une règle de discrimination linéaire ($\hat{\mathbf{\Sigma}}_1 = \cdots = \hat{\mathbf{\Sigma}}_K = \hat{\mathbf{\Sigma}}$) et quadratique ($\hat{\mathbf{\Sigma}}_k, 1 \leq k \leq K$, libres entre classes). L'estimateur $\hat{\mathbf{\Sigma}}_k(\beta)$ de $\mathbf{\Sigma}_k$ s'exprime par

$$\hat{\mathbf{\Sigma}}_k(\beta) = \frac{(1-\beta)(n_k - 1)\hat{\mathbf{\Sigma}}_k + \beta(n-K)\hat{\mathbf{\Sigma}}}{(1-\beta)(n_k-1) + \beta(n-K)}$$

où β ($0 \leq \beta \leq 1$) est un terme contrôlant le degré de mélange entre l'estimateur du cas linéaire ($\hat{\mathbf{\Sigma}}$) et quadratique ($\hat{\mathbf{\Sigma}}_k$). Le terme n_k correspond au nombre de points de l'échantillon appartenant à la classe k. De la même manière, Friedman réalise un compromis, mesuré par un paramètre γ ($0 \leq \gamma \leq 1$), entre $\hat{\mathbf{\Sigma}}_k(\beta)$ et la matrice identité $d \times d$ $\boldsymbol{I}$ avec la matrice $\hat{\mathbf{\Sigma}}_k(\beta, \gamma)$:

$$\hat{\mathbf{\Sigma}}_k(\beta,\gamma) = (1-\gamma)\hat{\mathbf{\Sigma}}_k(\beta) + \gamma \frac{\mathrm{tr}(\hat{\mathbf{\Sigma}}_k(\beta))}{d}\boldsymbol{I}.$$

Ainsi, un couple ($\beta \simeq 0, \gamma \simeq 0$) représente une règle de discrimination proche du cas quadratique, ($\beta \simeq 1, \gamma \simeq 0$) une règle proche du cas linéaire, etc. Les paramètres β et γ peuvent être soit estimés automatiquement à partir de l'ensemble d'apprentissage ($\mathcal{X}, \boldsymbol{z}$) par validation croisée, soit fixés a priori. Il ressort que la RDA donne de bons résultats mais par contre ne fournit pas des règles de classement facilement interprétables : comment interpréter, par exemple, un couple ($\beta \simeq 0.3, \gamma \simeq 0.3$) ? Les 14 modèles gaussiens semblent plus aisément manipulables car plus intuitifs.

1.4.3 Contraintes sur les proportions

Comme nous l'avons signalé, nous pouvons également contraindre les proportions. Nous considérons alors les deux configurations suivantes :

- soit les proportions sont forcées à être identiques entre toutes les classes ($p_1 = \cdots = p_K = 1/K$), donc elles n'ont pas à être estimées ;
- soit les proportions sont laissées libres et doivent être estimées.

En agglomérant les différentes contraintes sur les matrices de variance et celles sur les proportions on obtient finalement $2 \times 14 = 28$ modèles distincts : $[p\lambda I]$, $[p_k\lambda I]$, $[p\lambda C]$, $[p_k\lambda C]$, etc. Maintenant, lorsque nous parlerons des «14 modèles gaussiens», nous ferons implicitement référence aux 14 contraintes sur les matrices de variance avec la contrainte des *proportions égales*. Lorsque nous parlerons des «28 modèles gaussiens», tous les types de contraintes seront concernés : les *proportions seront égales ou libres*.

1.5 Conclusion

Les principes fondamentaux de la classification automatique et de la discrimination ont été présentés, en faisant divers usages de l'hypothèse de mélange gaussien. Il est maintenant utile de présenter concrètement des outils permettant d'estimer soit le paramètre de mélange (nécessaire pour la méthode du plug-in ou pour l'approche mélange), soit conjointement le paramètre mélange et la partition (nécessaire pour l'approche mixte). Ces méthodes d'estimation doivent tenir compte de l'a priori de l'utilisateur au travers des 28 modèles gaussiens. C'est l'objectif du chapitre suivant.

Chapitre 2

Méthodes d'estimation

2.1 Introduction

Ce chapitre propose de rappeler des méthodes d'estimation du paramètre du mélange gaussien multivarié, ou conjointement de ce paramètre et de la partition (méthode mixte de la classification automatique). Les solutions présentées prennent en compte les contraintes imposées par les 28 modèles gaussiens définies au chapitre précédent.

La section 2.2 présente la méthode du maximum de vraisemblance comme une première façon d'estimer le paramètre du mélange. Nous donnons l'expression de la vraisemblance dans le cas où les labels sont inconnus et dans les cas où ils sont connus (cas de la discrimination) et nous rappelons les propriétés du maximum de vraisemblance. L'algorithme EM est ensuite présenté comme un algorithme classique de maximisation de ce critère. Nous détaillons ses propriétés et nous insistons sur l'étape de maximisation qui permet de tenir compte du modèle gaussien choisi. Ensuite, nous présentons les algorithmes SEM et SAEM qui ont été développés pour remédier à certains défauts de EM (dépendance vis-à-vis des paramètres initiaux, mauvaise convergence parfois).

Nous considérons ensuite l'estimation du paramètre du mélange sous un angle bayésien. Le paradigme bayésien est brièvement présenté et les problèmes d'estimation de mélange qui lui sont propres sont rappelés. Les méthodes d'échantillonnage bayésien (souvent appelées méthodes «MCMC» pour *Monte Carlo Marcov chain*) sont un recours dans ce cas. La philosophie de l'approche théorie du codage est également énoncée et nous voyons de fortes similitudes avec les calculs bayésiens. Asymptotiquement et sous des conditions adéquates, les estimateurs de ces deux dernières approches sont équivalents à l'estimateur du maximum de vraisemblance.

La section 2.3 concerne l'estimation mixte, c'est-à-dire l'estimation simultanée du paramètre du mélange et de la partition. La vraisemblance classifiante est le critère à maximiser dans ce cas, et ses propriétés sont détaillées. Elle généralise des critères classiques comme le critère des «k-means» (Sebestyen [83]). L'algorithme CEM est un outil d'optimisation de la vraisemblance classifiante et nous faisons des rapprochements entre celui-ci et l'algorithme EM. Au travers d'un exemple, nous voyons l'intérêt de considérer l'approche mixte face à l'approche mélange : l'estimation de la partition est plus robuste par l'approche mixte. L'algorithme CAEM est enfin présenté comme une version de CEM permettant de s'affranchir des conditions initiales dont dépend CEM.

2.2 Estimation du paramètre du mélange

Il s'agit de rappeler des méthodes d'estimation du paramètre du mélange $\boldsymbol{\theta}$: maximum de vraisemblance, estimation bayésienne, théorie du codage. L'estimation se fait à partir de l'information disponible : données $\mathcal{X}$ seules, données $\mathcal{X}$ avec labels $\boldsymbol{z}$, modèle gaussien, nombre de classes.

2.2.1 Maximum de vraisemblance

Définition de la vraisemblance

Données sans labels Plaçons nous dans le cas où seul l'échantillon $\mathcal{X}$ est disponible, les labels $\boldsymbol{z}$ étant inconnus. Cette situation se produit en estimation mélange ou en classification automatique. Les n oberservations $\boldsymbol{x}_i$ de $\mathcal{X}$ étant indépendantes, on peut exprimer la densité de $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ par le produit de toutes les densités marginales :

$$\begin{aligned} f(\mathcal{X}|\boldsymbol{\theta}) &= f(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n|\boldsymbol{\theta}) \\ &= \prod_{i=1}^{n} f(\boldsymbol{x}_i|\boldsymbol{\theta}). \end{aligned} \tag{2.1}$$

Elle correspond à la vraisemblance (*likelihood*) du paramètre $\boldsymbol{\theta}$. Nous utiliserons plutôt, pour des raisons de calcul, le logarithme naturel de la vraisemblance qui s'écrit :

$$\mathrm{L}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \ln \left(\sum_{k=1}^{K} p_k \phi_k(\boldsymbol{x}_i|\boldsymbol{a}_k) \right). \tag{2.2}$$

Par abus de langage, le logarithme naturel de la vraisemblance sera souvent appelé simplement vraisemblance.

Données avec labels Si les labels $\boldsymbol{z}$ sont connus, c'est le cas en discrimination, l'échantillon complet $(\mathcal{X}, \boldsymbol{z})$ est utilisé pour estimer $\boldsymbol{\theta}$. La densité jointe de $(\mathcal{X}, \boldsymbol{z})$ s'écrit (équation 1.8)

$$f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta}) = \prod_{i=1}^{n} f(\boldsymbol{x}_i, z_i|\boldsymbol{\theta})$$

et correspond à la vraisemblance de $\boldsymbol{\theta}$. Le logarithme de la vraisemblance s'énonce :

$$\begin{aligned}
\mathrm{L}(\boldsymbol{\theta}) &= \sum_{i=1}^{n} \ln(p_{z_i}\phi_{z_i}(\boldsymbol{x}_i|\boldsymbol{a}_{z_i})) \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik} \ln(p_k\phi_k(\boldsymbol{x}_i|\boldsymbol{a}_k)),
\end{aligned} \tag{2.3}$$

avec $z_{ik} = 1$ ou 0 selon que $z_i = k$ ou non.

Par la suite, nous noterons la partition $\boldsymbol{z}$ de deux manières différentes :

- **un vecteur** de taille n : $\boldsymbol{z} = (z_1, \ldots, z_n)$. C'est la notation actuelle ;
- **une matrice binaire stochastique** de taille $n \times K$: $\boldsymbol{z} = \{z_{ik}\}$ avec $1 \leq i \leq n$ et $1 \leq k \leq K$ et $z_{ik} = 1$ ou 0 selon que $z_i = k$ ou non.

Par exemple, les deux notations suivantes représentent la même partition de trois points en deux classes ($\boldsymbol{x}_1$ et $\boldsymbol{x}_3$ dans la classe 1 et $\boldsymbol{x}_2$ dans la classe 2) :

$$\boldsymbol{z} = (1\ 2\ 1) \qquad \text{ou} \qquad \boldsymbol{z} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Données avec labels partiellement connus Parfois on ne connaît que partiellement les labels des données $\mathcal{X}$ (McLachlan [69]) : les données $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell$ ($1 \leq \ell \leq n$) ont les labels $\boldsymbol{z} = (z_1, \ldots, z_\ell)$ et les données restantes $\boldsymbol{x}_{\ell+1}, \ldots, \boldsymbol{x}_n$ ne sont pas étiquetées. La vraisemblance de $\boldsymbol{\theta}$ s'exprime alors comme une composée des deux vraisemblances précédentes :

$$\mathrm{L}(\boldsymbol{\theta}) = \sum_{i=1}^{\ell}\sum_{k=1}^{K} z_{ik} \ln(p_k\phi_k(\boldsymbol{x}_i|\boldsymbol{a}_k)) + \sum_{i=\ell+1}^{n} \ln\left(\sum_{k=1}^{K} p_k\phi_k(\boldsymbol{x}_i|\boldsymbol{a}_k)\right).$$

Nous n'utiliserons pas, par la suite, ce schéma, mais on voit que l'on pourrait facilement prendre en compte des informations partielles sur les labels.

Propriétés du maximum de vraisemblance

L'estimation par maximum de vraisemblance est une des méthodes les plus utilisées. Elle consiste à prendre comme estimateur du paramètre $\boldsymbol{\theta}$, la valeur qui rend maximale la vraisemblance $\mathrm{L}(\boldsymbol{\theta})$.

Si la vraisemblance (donc son logarithme) est deux fois dérivable par rapport à $\boldsymbol{\theta}$, ce qui est la cas dans nos situations, on obtient $\hat{\boldsymbol{\theta}}$ en résolvant les équations données par l'annulation du gradient et l'obtention d'une matrice hessienne non définie positive. Dans le cas où $\boldsymbol{\theta}$ est réduit à un seul paramètre monodimensionnel θ, cela conduit à résoudre le système

$$\left\{ \begin{array}{l} \frac{\partial}{\partial\theta}\mathrm{L}(\theta) = 0 \\ \frac{\partial^2}{\partial\theta^2}\mathrm{L}(\theta) < 0 \end{array} \right. .$$

Cet estimateur a beaucoup de propriétés statistiques désirables (voir Saporta p. 301 et 302 [80] à titre de référence). Par exemple, sous des conditions très générales, les estimateurs obtenus par cette technique sont consistants (ils convergent en probabilité vers les vraies valeurs des paramètres), ils sont asymptotiquement sans biais et sont asymptotiquement gaussiens. De plus l'existence d'une solution consistante des équations de vraisemblance a été prouvée par Kiefer (1978) [64] sous des hypothèses générales. Par contre le maximum de vraisemblance est théoriquement non borné (voir Robert [79] par exemple). En effet, si on prend autant de classes gaussiennes que de points dans l'échantillon et qu'on centre chaque classe en chaque point, la valeur de la vraisemblance diverge vers l'infini si la variance de chaque classe tend vers zéro. D'autre part, la résolution des équations aux dérivées partielles pour trouver l'optimum ne peut se faire que rarement analytiquement et on doit recourir souvent à des méthodes itératives. Ce sont certaines de ces méthodes que nous présentons maintenant.

2.2.2 Algorithmes maximisant la vraisemblance

Si les labels sont inconnus, différents algorithmes existent pour maximiser la vraisemblance : EM, SEM, SAEM. Si les labels sont connus, ces algorithmes sont inutiles : il suffit d'utiliser une seule fois l'étape de maximisation (étape M de EM) pour obtenir l'optimum.

Partition floue Dans la suite, nous aurons besoin d'une nouvelle notation. Nous noterons $\boldsymbol{c} = \{c_{ik}\}$ une matrice à n lignes et K colonnes ($i = 1, \ldots, n$, $k = 1, \ldots, K$) dont chaque élément $c_{ik} \in [0, 1]$ et dont la somme des éléments des lignes vaut 1 ($\sum_{k=1}^{K} c_{ik} = 1$). La matrice $\boldsymbol{c}$ sera appelée *partition floue* par opposition à la partition $\boldsymbol{z}$ ($z_{ik} \in \{0, 1\}$). La matrice $\boldsymbol{t} = \{t_{ik}\}$ des probabilités à posteriori est typiquement une partition floue.

L'algorithme EM

Présentation La méthode itérative la plus connue est sans doute l'algorithme EM (*Estimation - Maximisation*) de Dempster et al. [42] et Redner et Walker [76]. Outre la phase d'initialisation qui consiste à choisir des paramètres initiaux $\boldsymbol{\theta}^{\circ}$, il se décompose en deux étapes (décrites à la m^{e} itération) :

1. **Étape E (estimation) :** estimation de toutes les probabilités d'appartenance des $\boldsymbol{x}_i$ aux classes sachant les paramètres $\boldsymbol{\theta}^{m-1}$. Ce sont les t_{ik}^m de l'équation 1.9. On définit la partition floue $\boldsymbol{c}^m = \boldsymbol{t}^m$;

2. **Étape M (maximisation) :** choix de l'estimateur $\boldsymbol{\theta}^m$ qui maximise la vraisemblance $\mathrm{L}(\boldsymbol{\theta})$, sachant $\boldsymbol{c}^m$.

L'algorithme s'arrête à la stationnarité de la vraisemblance.

Cet algorithme a montré son efficacité dans l'estimation des paramètres avec beaucoup de variétés de composantes. Les exemples sont nombreux : des lois normales multivariées par Wolfe [91], une loi binomiale et de Poisson dans Everitt chap. 4 [47], ... De plus, dans le cas où le nombre de classes est fixé, EM est facile à programmer.

Etape M avec contraintes Il est très important de remarquer que la phase M de maximisation dépend de la paramétrisation du modèle de mélange. Dans le cas gaussien, on peut donc maximiser $\mathrm{L}(\boldsymbol{\theta})$ à chaque itération en voulant respecter des contraintes sur les matrices de variance $\boldsymbol{\Sigma}_k$ (voir la section 1.4.2) et sur les proportions p_k. Celeux et Govaert [38] ont détaillé la phase M pour les 14 modèles correspondant aux contraintes sur les matrices de variance. Pour certains modèles le calcul des matrices de variance est direct (par exemple $[\lambda I]$), tandis que d'autres nécessitent l'emploi d'un algorithme itératif (par exemple $[\lambda_k B]$).

La phase M de l'itération m dans le cas gaussien peut se résumer aux calculs suivants, en posant $n_k^m = \sum_{i=1}^n c_{ik}^m$ (population de la classe k) :

- les centres : $\hat{\boldsymbol{\mu}}_k^m = \bar{\boldsymbol{x}}_k = \frac{\sum_{i=1}^n c_{ik}^m \boldsymbol{x}_i}{n_k^m}$;
- les proportions : $\hat{p}_k^m = \frac{n_k^m}{n}$ si les proportions sont libres, $\hat{p}_k^m = \frac{1}{K}$ sinon ;
- les matrices de variances : se reporter à l'article de Celeux et Govaert. Par exemple, pour le modèle $[\lambda I]$, on a $\hat{\boldsymbol{\Sigma}}_k^m = \mathrm{tr}(\boldsymbol{W}(\boldsymbol{c}^m))/(nd)\boldsymbol{I}$ pour $1 \leq k \leq K$. Le terme $\boldsymbol{W}(\boldsymbol{c}^m)$ désigne la matrice d'inertie intraclasse

$$\boldsymbol{W}(\boldsymbol{c}^m) = \sum_{k=1}^K \boldsymbol{W}_k(\boldsymbol{c}_k^m) \tag{2.4}$$

avec $\boldsymbol{c}_k^m$ la k^{e} colonne de $\boldsymbol{c}^m$ et $\boldsymbol{W}_k(\boldsymbol{c}_k^m)$ la matrice d'inertie de la classe k définie par

$$\boldsymbol{W}_k(\boldsymbol{c}_k^m) = \sum_{i=1}^n c_{ik}^m(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_k^m)(\boldsymbol{x}_i - \hat{\boldsymbol{\mu}}_k^m)'. \tag{2.5}$$

Remarquons que n_k^m correspond à la «population floue» de la classe k car $\boldsymbol{c}^m$ est une partition floue (chaque $c_{ik}^m \in [0,1]$). D'autres algorithmes, qui sont des variantes de EM et que l'on décrira par la suite (SEM ou CEM par exemple), peuvent avoir une partition dure (chaque $c_{ik}^m \in \{0,1\}$) donc n_k^m s'interprétera comme une «vraie population». L'intérêt d'introduire la partition floue $\boldsymbol{c}^m$, est de traiter conjointement l'étape M de ces deux situations (voir Celeux et Govaert [38]), car la notion de partition floue inclus celle de partition :

$$\begin{aligned}
\boldsymbol{z} \text{ partition} \quad &\Leftrightarrow \quad \forall i \in \{1,\ldots,n\}, k \in \{1,\ldots,K\}, \quad z_{ik} \in \{0,1\} \\
&\Rightarrow \quad \forall i \in \{1,\ldots,n\}, k \in \{1,\ldots,K\}, \quad z_{ik} \in [0,1] \\
&\Leftrightarrow \quad \boldsymbol{z} \text{ partition floue.}
\end{aligned}$$

Croissance du critère Une explication intéressante de la propriété de convergence est donnée par Hathaway [58]. EM peut s'interpréter comme une *méthode d'optimisation alternée* sur la partition floue $\boldsymbol{c}$ et sur le paramètre $\boldsymbol{\theta}$. Posons

$$\mathrm{L}(\boldsymbol{\theta}, \boldsymbol{c}) = \mathrm{C}(\boldsymbol{\theta}, \boldsymbol{c}) + \mathrm{E}(\boldsymbol{c}) \tag{2.6}$$

avec $\mathrm{C}(\boldsymbol{\theta}, \boldsymbol{c})$ un terme de «type vraisemblance classifiante» (voir la section 2.3.1)

$$\mathrm{C}(\boldsymbol{\theta}, \boldsymbol{c}) = \sum_{i=1}^{n} \sum_{k=1}^{K} c_{ik} \ln(p_k \phi_k(\boldsymbol{x}_i | \boldsymbol{a}_k)), \tag{2.7}$$

et $\mathrm{E}(\boldsymbol{c})$ un terme d'entropie

$$\mathrm{E}(\boldsymbol{c}) = -\sum_{i=1}^{n} \sum_{k=1}^{K} c_{ik} \ln c_{ik}. \tag{2.8}$$

À l'itération m, l'étape E choisit la partition floue $\boldsymbol{c}^m$ qui maximise $\mathrm{L}(\boldsymbol{\theta}^{m-1}, \boldsymbol{c})$ sachant $\boldsymbol{\theta}^{m-1}$. Utilisant une méthode classique d'optimisation, on peut alors montrer que $\boldsymbol{c}^m = \boldsymbol{t}^m$ (avec $\boldsymbol{t}^m = \boldsymbol{t}(\boldsymbol{\theta}^{m-1})$), c'est-à-dire que $\boldsymbol{c}^m$ correspond à la matrice des probabilités a posteriori avec paramètre $\boldsymbol{\theta}^{m-1}$. On a donc

$$\mathrm{L}(\boldsymbol{\theta}^{m-1}, \boldsymbol{c}^m) \geq \mathrm{L}(\boldsymbol{\theta}^{m-1}, \boldsymbol{c}^{m-1}).$$

L'étape M choisit ensuite $\boldsymbol{\theta}^m$ qui maximise $\mathrm{L}(\boldsymbol{\theta}, \boldsymbol{c}^m)$ sachant $\boldsymbol{c}^m$ (cela revient de manière équivalente à maximiser $\mathrm{C}(\boldsymbol{\theta}, \boldsymbol{c}^m)$ sachant $\boldsymbol{c}^m$). On a alors l'inégalité

$$\mathrm{L}(\boldsymbol{\theta}^m, \boldsymbol{c}^m) \geq \mathrm{L}(\boldsymbol{\theta}^{m-1}, \boldsymbol{c}^m).$$

Passant à l'itération suivante $m+1$, à l'étape E on a de même $\mathrm{L}(\boldsymbol{\theta}^m, \boldsymbol{c}^{m+1}) \geq \mathrm{L}(\boldsymbol{\theta}^m, \boldsymbol{c}^m)$. D'où la relation

$$\mathrm{L}(\boldsymbol{\theta}^m, \boldsymbol{c}^{m+1}) \geq \mathrm{L}(\boldsymbol{\theta}^m, \boldsymbol{c}^m) \geq \mathrm{L}(\boldsymbol{\theta}^{m-1}, \boldsymbol{c}^m) \geq \mathrm{L}(\boldsymbol{\theta}^{m-1}, \boldsymbol{c}^{m-1}). \tag{2.9}$$

Puisque $\boldsymbol{c}^m = \boldsymbol{t}^m$, on remarque enfin qu'on a l'égalité (calcul direct)

$$\mathrm{L}(\boldsymbol{\theta}^{m-1}) = \mathrm{L}(\boldsymbol{\theta}^{m-1}, \boldsymbol{c}^m),$$

ce qui induit bien la croissance du critère de vraisemblance

$$\mathrm{L}(\boldsymbol{\theta}^m) \geq \mathrm{L}(\boldsymbol{\theta}^{m-1}).$$

Propriétés de convergence Wu [93] a mené une étude détaillée de la convergence de EM. Sous des conditions assez générales et vérifiées dans beaucoup de situations pratiques, il a montré que la suite des valeurs de vraisemblance convergeait vers une valeur stationnaire de la vraisemblance. Mais, en général, on ne peut affirmer que EM converge vers un optimum local de la vraisemblance (à moins de supposer des conditions très contraignantes). Parfois l'algorithme ne peut aboutir qu'à un col de la fonction de vraisemblance. En effet, le résultat dépend du choix du paramètre initial $\boldsymbol{\theta}^\circ$et surtout des positions initiales $\boldsymbol{\mu}_k^\circ$ $(1 \leq k \leq K)$ pour le cas gaussien. On peut voir à ce sujet McLachlan et Basford [70] ou Dempster et al. [42].

Stratégies d'utilisation Pour atténuer la dépendance du résultat par rapport à ce choix initial, plusieurs techniques sont possibles. La plus simple est d'initialiser avec les vrais paramètres, ou au moins avec les vrais centres lorsque cela est possible. C'est bien sûr exclus pour les données réelles. Une autre solution est de lancer EM avec plusieurs choix aléatoires pour les centres initiaux, puis de ne retenir comme estimateur de $\boldsymbol{\theta}$ que celui ayant donné la plus grande vraisemblance $\mathrm{L}(\boldsymbol{\theta})$. D'autres méthodes existent : voir une initialisation utilisant l'algorithme des *k-means* (Sebestyen [83]) dans Celeux et Soromenho [40], etc. De toute façon, il est nécessaire de faire un compromis entre le temps de calcul nécessaire pour trouver un plus ou moins bon $\boldsymbol{\theta}^\circ$ et le temps mis par EM à converger. Si EM est assez rapide, choisir le meilleur de plusieurs essais de $\boldsymbol{\theta}^\circ$ peut être une option satisfaisante. Notons toutefois que EM peut être très lent, surtout aux alentours d'un col de la vraisemblance (ou point selle). Plusieurs milliers d'itérations peuvent être nécessaires dans certains cas.

Des algorithmes ont été développés pour chercher le maximum global de $\mathrm{L}(\boldsymbol{\theta})$. On peut citer à ce titre l'algorithme SEM qui est une version stochastique de EM. C'est cet algorithme que nous décrivons maintenant.

L'algorithme SEM

Présentation L'algorithme SEM (*Stochastique* EM) de Celeux et Diebolt [33] vise à répondre aux principales limitations de EM, surtout si les classes sont peu séparées : lenteur éventuelle, importance cruciale de la position initiale, convergence vers des cols de vraisemblance, nombre de classes. Il incorpore une étape stochastique (étape S) entre les étapes E et M de EM. Cette étape S repose sur un *principe d'affectation aléatoire* (*Random Imputation Principle*), c'est-à-dire que l'on choisit aléatoirement l'appartenance des points aux classes en tenant compte des probabilités d'appartenance t_{ik}.

L'algorithme part d'une position initiale $\boldsymbol{\theta}^\circ$ avec un nombre K de classes non inférieur au «vrai» nombre et sa m^{e} itération s'énonce :

1. **Étape E (estimation) :** estimation des probabilités t_{ik}^m pour chaque $\boldsymbol{x}_i$ sachant $\boldsymbol{\theta}^{m-1}$;

2. **Étape S (stochastique) :** tirage pour chaque $\boldsymbol{x}_i$ de la variable $\boldsymbol{z}_i^m = (z_{ik}^m, k = 1, \ldots, K)$ selon une loi multinomiale d'ordre un et de paramètre $(t_{ik}^m, k = 1, \ldots, K)$. On définit une partition $\boldsymbol{c}^m$ à partir des $\boldsymbol{z}_i^m$. S'il existe k tel que $\mathrm{card}\{\boldsymbol{x}_i / z_{ik}^m = 1\} \leq d + 1$, l'algorithme est réinitialisé sur la base de $K - 1$ classes. Sinon passer à l'étape M;

3. **Étape M (maximisation) :** maximisation de $\mathrm{L}(\boldsymbol{\theta})$ sachant $\boldsymbol{c}^m$ pour trouver $\boldsymbol{\theta}^m$ optimum (voir EM).

L'algorithme s'arrête après un nombre d'itérations choisi par l'utilisateur.

Estimation du nombre de classes L'algorithme SEM, grâce au mécanisme de réinitialisation décrit à la fin de l'étape S, permet une estimation correcte du nombre K de classes si son initialisation se fait avec un nombre de classes non inférieur au «vrai» nombre et si le nombre de points n'est pas trop faible. Dans ce dernier cas, il y a sous-estimation de K car l'influence des perturbations aléatoires est trop forte. Le principe du mécanisme de réinitialisation de l'étape S assure en outre que la suite $(\boldsymbol{\theta}^m)$ reste dans un compact sur lequel la vraisemblance L est de classe $\mathcal{C}^2$ donc bornée. Il existe aussi des versions de SEM où le seuil identifiant des classes «trop petites» est fixé par l'utilisateur.

Propriétés de convergence L'ensemble des propriétés de SEM a été étudié par Celeux et Diebolt [34]. La suite des paramètres $(\boldsymbol{\theta}^m)$ engendrée par SEM ne converge pas ponctuellement à cause des perturbations aléatoires introduites dans l'étape S mais converge en loi vers l'unique distribution stationnaire. Cette suite est une chaîne de Markov homogène et si, pour tout $\boldsymbol{\theta}$, $t_{ik}(\boldsymbol{\theta})$ est strictement positif, elle est irréductible. La moyenne de la loi stationnaire approche l'optimum global par un optimum local *significatif* des équations de maximum de vraisemblance, ceci lorsque la taille de l'échantillon tend vers l'infini ($n \to \infty$). La variance de cette loi décroit vers 0 dans ce cas limite. L'analyse statistique de la loi stationnaire de SEM fournit aussi des indicateurs précieux pour évaluer l'incertitude attachée à l'estimation des paramètres.

Il est à noter que SEM est moins fiable que EM pour l'identification de modèles à partir de très petits échantillons : dans ce cas, les perturbations aléatoires de l'étape S prennent trop d'importance. D'autre part, contrairement à EM, le résultat est indépendant de la position initiale : la suite ne séjourne pas indéfiniment au voisinage d'un col de la vraisemblance et donc il n'y a plus de situations de convergence lente.

Stratégie d'utilisation En pratique, l'algorithme SEM se décompose en deux phases distinctes : la phase d'apprentissage et la phase de travail. La première fait tourner SEM quelques douzaines d'itérations en espérant ainsi que la séquence $(\boldsymbol{\theta}^m)$ aura atteint la stationnarité. Ensuite la deuxième phase consiste à enregistrer les valeurs suivantes de $(\boldsymbol{\theta}^m)$ pour calculer finalement leur moyenne empirique et parfois leur variance. La moyenne donne ainsi une estimation ponctuelle et la variance une évaluation de la précision de ces estimateurs.

Lien avec CEM et avec Bayes L'algorithme SEM peut aussi s'interpréter comme une version stochastique de l'algorithme de classification qu'est CEM (voir la section 2.3.2) : l'étape S est simplement une version stochastique de l'étape C (*Classification*) de CEM (Celeux et Govaert [36]).

D'autre part, Diebolt et Robert [44] ont fait ressortir des liens entre SEM et l'échantillonnage bayésien (voir la section 2.2.3 sur l'approche bayésienne). En effet, SEM peut être vu comme une version simplifiée de l'algorithme de Data Augmentation de Tanner et Wong [87] avec des lois a priori non informatives. Un parallèle similaire peut se faire avec l'échantillonnage de Gibbs.

L'algorithme SAEM

Présentation L'algorithme SAEM (*Simulated Annealing* EM) de Celeux et Diebolt [35] occupe une place intermédiaire entre EM et SEM : c'est une version de type *recuit simulé* de EM et conjointement une adaptation de l'algorithme stochastique SEM. Il répond, comme SEM, aux limitations bien connues de EM, mais se comporte mieux que SEM pour traiter de petits échantillons car il réduit l'intensité des perturbations au fil des itérations. De plus il est plus simple à appréhender que SEM par sa convergence ponctuelle (et non plus en loi).

Entre les étapes S et M de SEM, une étape A (*Annealing*) est introduite. Elle permet de réduire les perturbations aléatoires de l'étape S et ceci de plus en plus fortement avec le nombre d'itérations grâce à une suite de réels positifs (γ^m) qui décroit vers 0 lorsque le nombre d'itérations tend vers l'infini, en partant de $\gamma^\circ = 1$. Cette suite joue le même rôle que la température dans les algorithmes de recuit simulé. L'algorithme part d'une position initiale $\boldsymbol{\theta}^\circ$ avec un nombre K de classes non inférieur au «vrai» nombre et avec $\gamma^\circ = 1$. Sa m^{e} itération s'énonce :

1. **Étape E (estimation) :** estimation des probabilités t_{ik}^m pour chaque $\boldsymbol{x}_i$ sachant $\boldsymbol{\theta}^{m-1}$;

2. **Étape S (stochastique) :** tirage pour chaque $\boldsymbol{x}_i$ de la variable $\boldsymbol{z}_i^m = (z_{ik}^m, k = 1, \ldots, K)$ selon une loi multinomiale d'ordre un et de paramètre $(t_{ik}^m, k = 1, \ldots, K)$. S'il existe k tel que $\mathrm{card}\{\boldsymbol{x}_i / z_{ik}^m = 1\} \leq d + 1$, l'algorithme est réinitialisé sur la base de $K - 1$ classes. Sinon passer à l'étape A ;

3. **Étape A (annealing) :** calcul de la partition floue d'éléments $c_{ik}^m = t_{ik}^m + \gamma^{m-1}(z_{ik}^m - t_{ik}^m)$. Une nouvelle température γ^m est calculée ;

4. **Étape M (maximisation) :** maximisation de $\mathrm{L}(\boldsymbol{\theta})$ sachant $\boldsymbol{c}^m$ pour trouver $\boldsymbol{\theta}^m$ optimum (voir EM).

L'algorithme s'arrête à la stationnarité de la vraisemblance. Le principe de SAEM est donc de commencer comme l'algorithme *stochastique* SEM, pour aboutir en limite ($m \to \infty$) à l'algorithme *déterministe* EM.

On se reportera à la partie sur SEM pour les remarques concernant l'estimation du nombre de classes.

Propriétés de convergence Toute suite d'estimés par SAEM converge presque sûrement vers un maximum local de la fonction de vraisemblance, dans le cas d'un mélange dont les composants ont des densités de probabilités appartenant à la famille exponentielle. Mais le comportement théorique et pratique de l'algorithme dépend fortement de la vitesse de convergence de la suite (γ^m) vers 0 ainsi que de la décroissance régulière de celle-ci. Celeux et Diebolt [35] ont étudié ce sujet et constatent par des simulations qu'une lente décroissance de (γ^m) est préférable à une décroissance trop rapide. Ils proposent une version «rapide» (linéaire) et «lente» de (γ^m) qui sont respectivement $\gamma^m = -\frac{m}{m_{\max}}+1$ et $\gamma^m = \cos\left(m\frac{\pi}{2}/m_{\max}\right)$, avec $m_{\max}$ le nombre maximum d'itérations effectuées.

2.2.3 Estimation bayésienne

Coût, estimateur de Bayes et loi a priori

L'inférence bayésienne s'inscrit dans le cadre de la théorie bayésienne de la décision et consiste à prendre une décision $\hat{\boldsymbol{\theta}}$ concernant le vrai paramètre inconnu $\boldsymbol{\theta} \in \Theta$, et ceci au vu de l'échantillon $\mathcal{X}$. La décision $\hat{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}}(\mathcal{X})$ est donc une fonction de $\mathbb{R}^d$ dans l'espace des décisions et correspond à notre estimateur.

Étant donné un coût $\mathcal{L}$, fonction de $\{$espace des décisions$\} \times \Theta$ dans $[0, +\infty[$ qui associe donc un réel positif à chaque décision, et étant donnée une loi a priori $\pi(\boldsymbol{\theta})$ sur les paramètres, l'estimateur de Bayes $\hat{\boldsymbol{\theta}}$ de $\boldsymbol{\theta}$ est celui qui minimise le risque a posteriori

$$\begin{aligned} \mathcal{R}(\hat{\boldsymbol{\theta}}|\mathcal{X}) &= \int_\Theta \mathcal{L}(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta})\pi(\boldsymbol{\theta}|\mathcal{X})d\boldsymbol{\theta} \\ &= \mathrm{E}^\pi[\mathcal{L}(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta})|\mathcal{X}]. \end{aligned}$$

On a noté $\pi(\boldsymbol{\theta}|\mathcal{X})$ la loi a posteriori des paramètres, loi obtenue par le théorème de Bayes

$$\begin{aligned} \pi(\boldsymbol{\theta}|\mathcal{X}) &= \frac{f(\mathcal{X}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\int_\Theta f(\mathcal{X}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta}} \\ &= \frac{f(\mathcal{X},\boldsymbol{\theta})}{f(\mathcal{X})}, \end{aligned} \tag{2.10}$$

avec $f(\mathcal{X}|\boldsymbol{\theta})$ la loi jointe des $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ donnée par l'équation 2.1. Deux questions se posent alors : le choix du coût $\mathcal{L}$ et le choix de la loi a priori π.

Remarque sur les données avec labels Nous ne considérerons pas cette situation dans le cas bayésien mais le principe de base est le même que lorsque les labels sont absents. On minimise le risque conditionnel

$$\mathcal{R}(\hat{\boldsymbol{\theta}}|\mathcal{X}, \boldsymbol{z}) = \mathrm{E}^\pi[\mathcal{L}(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta})|\mathcal{X}, \boldsymbol{z}]$$

avec la loi a posteriori

$$\pi(\boldsymbol{\theta}|\mathcal{X}) = \frac{f(\mathcal{X}, \boldsymbol{z}, \boldsymbol{\theta})}{f(\mathcal{X}, \boldsymbol{z})}.$$

L'extension est aussi aisée pour les données avec labels partiellement connus.

Choix du coût

Il existe des fonctions de coût usuelles comme le coût quadratique, $\mathcal{L}(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}) = (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})'$, qui est sans doute le plus utilisé et le coût 0–1 surtout utilisé pour la théorie classique des tests d'hypothèse et proposé par Neyman et Pearson. Il vaut 0 si la décision est correcte, 1 sinon. De tels coûts sont bien sûr arbitraires et dépendent du problème considéré donc sont choisis au cas par cas. Souvent, le choix du coût quadratique est fondé sur sa simplicité.

Estimateur pour le coût quadratique Il est prouvé (voir Robert p.68 [79] ou Bernardo et Smith p.257-258 [10]) que l'estimateur de Bayes $\hat{\boldsymbol{\theta}}$ associé à la loi a priori π et au coût quadratique $\mathcal{L}(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}) = (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})'$ est la moyenne a posteriori :

$$\begin{aligned} \hat{\boldsymbol{\theta}} &= \int_{\Theta} \boldsymbol{\theta} \pi(\boldsymbol{\theta}|\mathcal{X}) d\boldsymbol{\theta} \\ &= \mathrm{E}^{\pi}[\boldsymbol{\theta}|\mathcal{X}], \end{aligned} \tag{2.11}$$

donc ne nécessite que le calcul de moments a posteriori.

Estimateur pour le coût 0–1 L'estimateur de Bayes associé au coût 0–1 est le mode de la loi a posteriori $\pi(\boldsymbol{\theta}|\mathcal{X})$ (voir Bernardo et Smith p.257-258 [10]). En fait, il est équivalent de prendre le mode de $f(\mathcal{X}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$ (proportionnalité avec $\pi(\boldsymbol{\theta}|\mathcal{X})$) :

$$\hat{\boldsymbol{\theta}} = \{\boldsymbol{\theta} : \max_{\theta} f(\mathcal{X}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})\}. \tag{2.12}$$

Choix de la loi a priori

Information a priori Le choix de la loi a priori des paramètres peut s'effectuer de diverses manières (voir Robert chap. 3 par exemple [79] pour une vue d'ensemble), selon le degré d'information a priori disponible : lois a priori non informatives de Jeffreys basées sur l'information de Fisher, méthode du maximum d'entropie, méthodes bayésiennes empiriques et hiérarchiques, loi a priori conjuguées, expérience venant de données précédentes ou d'experts, etc. De toute façon, l'influence de cette loi s'atténue pour l'inférence bayésienne quand la taille de l'échantillon augmente. Asymptotiquement et si la loi a priori vérifie certaines conditions, seule l'information tirée de l'échantillon intervient.

Lois conjuguées Le cas des lois a priori conjuguées est à la frontière entre statistiques bayésienne et fréquentiste. Ces lois sont souvent employées par souci de *simplification*. Une famille $\mathcal{F}$ de lois sur l'ensemble des paramètres Θ est conjuguée si, pour toute loi a priori $\pi(\boldsymbol{\theta}) \in \mathcal{F}$, la loi a posteriori $\pi(\boldsymbol{\theta}|\mathcal{X})$ appartient aussi à $\mathcal{F}$. Pour illustrer ce propos, prenons par exemple

$$f(x|\mu) = \mathcal{N}(x|\mu, \sigma^2) \quad (\sigma^2 \text{ connu})$$

et

$$\pi(\mu) = \mathcal{N}(\mu|\nu, \tau^2) \quad \in \mathcal{F}$$

loi appartenant à la famille exponentielle $\mathcal{F}$. On obtient alors la loi a posteriori

$$\pi(\mu|x) = \mathcal{N}(\mu|\rho(\sigma^2\nu + \tau^2 x), \rho\sigma^2\tau^2) \quad \in \mathcal{F} \tag{2.13}$$

(avec $\rho^{-1} = \sigma^2 + \tau^2$) qui appartient également à la famille exponentielle $\mathcal{F}$. Les termes ν et τ^2, paramètres de la loi a priori, sont appelés *hyperparamètres.*

Raiffa et Schlaifer [75], qui sont à l'origine des lois a priori conjuguées, justifient cette approche par un principe d'invariance. Lorsque l'observation $\mathcal{X} \sim f(\mathcal{X}|\boldsymbol{\theta})$ vient modifier $\pi(\boldsymbol{\theta})$ en $\pi(\boldsymbol{\theta}|\mathcal{X})$, l'information apportée par $\mathcal{X}$ sur $\boldsymbol{\theta}$ est *limitée* donc ne doit pas remettre en cause la forme de $f(\mathcal{X}|\boldsymbol{\theta})$ mais uniquement ses paramètres. Une telle approche a ses détracteurs parmi les bayésiens, ceux-ci considérant que l'information fournie par $\mathcal{X}$ doit toujours être prédominante par rapport à la loi a priori $\pi(\boldsymbol{\theta})$. Mais il n'en reste pas moins qu'elle permet de faire un *choix automatique* de lois a priori et aussi d'exprimer *explicitement* les lois a posteriori. Un choix cohérent est ensuite bien sûr nécessaire en ce qui concerne les hyperparamètres, c'est-à-dire les paramètres des lois a priori, pour tenir compte le plus possible à travers eux de toute information a priori disponible.

modèles gaussiens On a vu à la section 1.4 que les modèles gaussiens représentaient une information a priori sur l'espace Θ du paramètre $\boldsymbol{\theta}$. Dans un contexte bayésien, l'information a priori disponible sur les paramètres est introduite au travers de $\pi(\boldsymbol{\theta})$ et donc chaque modèle gaussien aura une loi a priori associée différente. Utilisant des loi conjuguées, Bensmail et al. [9] proposent ainsi 8 lois a priori assorties à 8 modèles parmi les 14 modèles gaussiens.

Estimateurs explicites

La minimisation du risque a posteriori $\mathcal{R}(\hat{\boldsymbol{\theta}}|\mathcal{X})$ sur $\hat{\boldsymbol{\theta}}$ est résolue explicitement dans le cas du coût quadratique (équation 2.11) mais nécessite le calcul de moments a posteriori, c'est-à-dire la connaissance de la loi a posteriori. L'utilisation de lois conjuguées apporte une réponse simplificatrice de ce côté en facilitant aussi les calculs car la loi a posteriori $\pi(\boldsymbol{\theta}|\mathcal{X})$ (équation 2.10) est alors explicite comme on vient de le voir. Nous nous placerons donc maintenant dans le cas où le coût est quadratique et la loi a priori est conjuguée.

Dans cette situation, la moyenne a posteriori est directement calculable parfois. C'est le cas lorsque la loi a posteriori est classique comme la loi normale de l'équation 2.13. Sinon l'estimation de l'intégrale servant au calcul des moments a posteriori peut se faire par une méthode de Monte-Carlo. On génère m réalisations $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m$ suivant la loi a posteriori $\pi(\boldsymbol{\theta}|\mathcal{X})$ et la loi des grands nombres assure alors la convergence en probabilité ($m \to \infty$)

$$\frac{1}{m}\sum_{i=1}^{m}\boldsymbol{\theta}_i \longrightarrow \int_{\Theta}\boldsymbol{\theta}\pi(\boldsymbol{\theta}|\mathcal{X})d\boldsymbol{\theta} = \mathrm{E}^{\pi}[\boldsymbol{\theta}|\mathcal{X}]. \tag{2.14}$$

Malheureusement, en mélange, la loi a posteriori, quoique explicite, n'est pas calculable pour une taille d'échantillon raisonnable. Ainsi la suite $(\boldsymbol{\theta}_m)$ ne peut être générée comme on le voit sur l'exemple suivant.

Difficulté d'estimation en mélange

Reprenons l'exemple traité par Robert [79] (exemple 9.5 du chapitre 9) qui montre cette impossiblité pratique de mener à bien le calcul explicite de l'estimateur de Bayes pour un mélange. Le n échantillon est supposé provenir d'un mélange de deux lois gaussiennes univariées

$$f(x|\boldsymbol{\theta}) = p\phi_1(x|\boldsymbol{a}_1) + (1-p)\phi_2(x|\boldsymbol{a}_2).$$

Robert prenant des lois a priori conjuguées et réutilisant les travaux de Diebolt et Robert [43] obtient la loi a posteriori

$$\pi(\boldsymbol{\theta}|\mathcal{X}) = \sum_{(\boldsymbol{z})} \pi(\boldsymbol{\theta}|\boldsymbol{z})\mathrm{P}(\boldsymbol{z}|\mathcal{X})$$

où la somme est prise sur l'ensemble $(\boldsymbol{z})$ des partitions de $\mathcal{X}$ en deux classes. Le terme $\mathrm{P}(\boldsymbol{z}|\mathcal{X})$ désigne la probabilité a posteriori de la partition $\boldsymbol{z}$. L'estimateur de Bayes sous coût quadratique s'écrit ensuite

$$\hat{\boldsymbol{\theta}} = \sum_{(\boldsymbol{z})} \mathrm{E}^{\pi}[\boldsymbol{\theta}|\boldsymbol{z}]\mathrm{P}(\boldsymbol{z}|\mathcal{X})$$

et correspond à la somme pondérée des estimateurs de Bayes pour chaque partition. Les estimateurs sont donc explicites mais par contre sont incalculables pour une taille d'échantillon raisonnable ($n > 40$) puisqu'ils nécessitent le calcul de 2^n termes, correspondant à toutes les partitions possibles.

L'échantillonnage bayésien : méthodes MCMC

Il est possible de résoudre le problème précédent en générant une séquence $(\boldsymbol{\theta}_m)$ à partir de la loi a priori $\pi(\boldsymbol{\theta})$ plutôt qu'à partir de la loi a posteriori. Dans ce cas la méthode de Monte-Carlo donne la convergence en probabilité suivante

$$\frac{1}{m}\sum_{i=1}^{m} \boldsymbol{\theta}_i f(\mathcal{X}|\boldsymbol{\theta}_i) \longrightarrow \int_{\Theta} \boldsymbol{\theta} f(\mathcal{X}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta},$$

puis les moments a posteriori sont estimés par

$$\mathrm{E}^{\pi}[\boldsymbol{\theta}|\mathcal{X}] \approx \frac{\frac{1}{m}\sum_{i=1}^{m} \boldsymbol{\theta}_i f(\mathcal{X}|\boldsymbol{\theta}_i)}{\frac{1}{m}\sum_{i=1}^{m} f(\mathcal{X}|\boldsymbol{\theta}_i)}.$$

L'inconvénient est que la convergence est beaucoup moins rapide que lorsque l'on génère à partir de la loi a posteriori. Même l'usage d'une méthode de Monte-Carlo avec fonction d'importance (Geweke [54]) n'est pas satisfaisante car trop dépendante du choix de la fonction d'importance, fonction difficile à définir. Cette fonction doit être en effet proche de loi a posteriori pour donner de bons résultats...

L'échantillonnage bayésien est une méthode de Monte-Carlo qui permet de générer une séquence $(\boldsymbol{\theta}_m)$ du vecteur aléatoire suivant approximativement la loi a posteriori $\pi(\boldsymbol{\theta}|\mathcal{X})$ lorsque celle-ci n'est pas disponible comme dans la situation mélange. L'idée de base est de construire une chaîne de Markov dont la distribution stationnaire est $\pi(\boldsymbol{\theta}|\mathcal{X})$. Obtenant $\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^t$ réalisations d'une telle chaîne de Markov, on a alors le résultat asymptotique $(t \to \infty)$

$$\boldsymbol{\theta}^t \to \boldsymbol{\theta} \sim \pi(\boldsymbol{\theta}|\mathcal{X}) \quad \text{en distribution.}$$

Par itération de m chaînes de Markov avec initialisations différentes (ou bien en laissant suffisammant d'espace entre les $\boldsymbol{\theta}^t$ retenus), on obtient un échantillon $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m$ directement utilisable pour estimer $\mathrm{E}^\pi[\boldsymbol{\theta}|\mathcal{X}]$ par l'équation 2.14.

Cette technique est fondée sur les travaux de Geman et Geman (1984) [53], Tanner et Wong (1987) [87] et Gelfand et Smith (1990) [52], et a pour nom générique MCMC (*Monte Carlo Markov chain*). Le terme MCMC regroupe plusieurs types d'algorithmes : la méthode de data augmentation (Tanner et Wong [87]), la méthode de Gibbs (variante de la précédente) et la méthode de Metroplis-Hastings (Hastings [57]). Elles utilisent à la base les structures hiérarchiques du modèle bayésien.

Dans le cas des mélanges, une structure hiérarchique intéressante qui fait apparaître explicitement la partition $\boldsymbol{z}$, est donnée par

$$\pi(\boldsymbol{\theta}|\mathcal{X}) = \sum_{(\boldsymbol{z})} \pi(\boldsymbol{\theta}|\boldsymbol{z})\mathrm{P}(\boldsymbol{z}|\mathcal{X}).$$

Bensmail et al. [9] utilisent alors une méthode MCMC proche de l'algorithme de Gibbs. Partant de paramètres initiaux $\boldsymbol{\theta}^\circ = (\boldsymbol{p}^\circ, \boldsymbol{a}^\circ)$ avec $\boldsymbol{p}^\circ = (p_1^\circ, \ldots, p_{K-1}^\circ)$ et $\boldsymbol{a}^\circ = (\boldsymbol{a}_1^\circ, \ldots, \boldsymbol{a}_K^\circ)$, l'algorithme itère en trois étapes (décrites à la t^{e} itération) :

1. simuler $\boldsymbol{z}^t$ suivant leur probabilité a posteriori conditionnellement à $\boldsymbol{\theta}^{t-1}$ donnée par les t_{ik} ;
2. simuler $\boldsymbol{p}^t$ selon sa distribution a posteriori conditionnellement à $\boldsymbol{z}^t$;
3. simuler $\boldsymbol{a}^t$ suivant sa distribution a posteriori conditionnellement à $\boldsymbol{z}^t$.

Utilisant des lois a priori conjuguées pour les paramètres, $\boldsymbol{p}$ suit a priori une loi de Dirichlet, les moyennes $\boldsymbol{\mu}_k$ conditionnellement aux matrices de variance $\boldsymbol{\Sigma}_k$ suivent a priori une loi normale multivariée. Les matrices de variance, conditionnellement à $\boldsymbol{z}$, suivent une loi a priori qui dépend du modèle gaussien considéré. Bensmail et al. expriment cette distribution pour 8 modèles parmi les 14 possibles : $[\lambda I]$, $[\lambda_k I]$, $[\lambda C]$, $[\lambda_k C]$, $[\lambda D_k A D_k']$, $[\lambda_k D_k A D_k']$, $[\lambda_k D A_k D']$ et $[\lambda_k C_k]$.

Quand le nombre d'itérations devient grand ($t \to \infty$), le couple ($\boldsymbol{\theta}^t, \boldsymbol{z}^t$) tend en distribution vers un vecteur aléatoire de densité jointe $\pi(\boldsymbol{\theta}, \boldsymbol{z}|\mathcal{X})$. En particulier $\boldsymbol{\theta}^t$ tend en distribution vers la loi $\pi(\boldsymbol{\theta}|\mathcal{X})$. Le choix de la valeur de t est assez empirique mais la convergence semble assez rapide en pratique (de l'ordre de quelques centaines d'itérations).

Lien avec la vraisemblance

Dans le cas où la taille de l'échantillon devient très grande ($n \to \infty$), l'estimateur par maximum de vraisemblance et l'estimateur de Bayes deviennent équivalents sous certaines hypothèses de régularité (voir par exemple Bernardo p. 287 et suivantes [10]). La convergence est d'autant plus rapide que les lois sont non informatives.

2.2.4 Utilisation de la théorie du codage

Le principe MDL

Soit un alphabet, ou ensemble dénombrable de symboles, $\mathcal{A} = \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_\ell\}$ et supposons que les données $\mathcal{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ soient discrètes et composées uniquement des symboles de $\mathcal{A}$ ($\forall i \in \{1, \ldots, n\},\ \boldsymbol{x}_i \in \mathcal{A}$). Pour fixer les idées, $\mathcal{A}$ peut correspondre aux lettres classiques de l'alphabet et $\mathcal{X}$ à un texte écrit avec les lettres de cet alphabet. L'idée fondamentale de l'approche MDL (*Minimum Description Length*) est de *diminuer la taille* du texte $\mathcal{X}$. Pour ce faire, on va *coder* le texte en remplaçant chaque symbole de $\mathcal{X}$ par un équivalent en binaire : son code. N'importe qu'elle autre base conviendrait. Le code résultant doit répondre à deux exigences :

1. être un «code préfixe» (voir Rissanen p.22 [78]) ;

2. être un code de longueur minimale, donc que la séquence des symboles binaires 0 ou 1 soit la plus courte possible.

Il semble naturel d'assigner un code court (peu de symboles binaires) à un symbole de $\mathcal{A}$ arrivant très fréquemment dans $\mathcal{X}$ et, au contraire, un code plus long à un symbole plus rare. Cela fournit la connexion de base entre la théorie du codage et l'approche par modèle statistique. Si $f(\boldsymbol{x})$ est la fonction fréquence du symbole $\boldsymbol{x} \in \mathcal{A}$ dans le texte, des résultats classiques de la théorie du codage (voir Rissanen p.26) montrent que la relation entre la longueur $h(\mathcal{X})$ optimale du code binaire et la fonction fréquence est

$$h(\mathcal{X}) = -\sum_{i=1}^{n} \log_2 f(\boldsymbol{x}_i),$$

où «$\log_2$» dénote le logarithme à base 2. Par commodité, nous utiliserons dans la suite le logarithme népérien, «ln». Ce choix permet ainsi d'éliminer une constante dans les calculs.

Supposons maintenant que f soit une fonction paramétrée par $\boldsymbol{\theta}$. Dans ce cas, la longueur minimale de code associée à un $\boldsymbol{\theta}$ fixé est donnée par $h(\mathcal{X}|\boldsymbol{\theta}) = -\mathrm{L}(\boldsymbol{\theta})$ où $\mathrm{L}(\boldsymbol{\theta}) = \ln f(\mathcal{X}|\boldsymbol{\theta})$ est la vraisemblance de $\boldsymbol{\theta}$ (section 2.2.1). La valeur du paramètre $\boldsymbol{\theta}$ qui minimise cette longueur de code $h(\mathcal{X}|\boldsymbol{\theta})$ est donc égale à la valeur de l'estimateur du maximum de vraisemblance. Si $\hat{\boldsymbol{\theta}}$ dénote l'estimateur du maximum de vraisemblance, la longueur minimale de code sera ainsi $h(\mathcal{X}|\hat{\boldsymbol{\theta}})$.

Mais le code $h(\mathcal{X}|\boldsymbol{\theta})$ est incomplet car, pour pouvoir décoder le texte correctement, il manque une information sur $\boldsymbol{\theta}$. En effet, le nouveau texte codé doit contenir à son début le codage de $\boldsymbol{\theta}$, pour qu'ensuite le lecteur ait la clé de décodage pour le reste du texte codé. Pour ce faire, supposons que $\boldsymbol{\theta}$ soit élément d'un nouvel alphabet $\Theta = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m\}$ (dénombrable même si infini). Procédant de la même manière qu'avec les symboles de l'alphabet $\mathcal{A}$, il suffit de connaître une fonction fréquence π sur Θ pour coder $\boldsymbol{\theta}$. La longueur de code résultant du codage du texte et des paramètres s'énonce alors comme la somme de la longueur du code de $\mathcal{X}$ et de la longueur du code de $\boldsymbol{\theta}$:

$$h(\mathcal{X}, \boldsymbol{\theta}) = -\mathrm{L}(\boldsymbol{\theta}) - \ln \pi(\boldsymbol{\theta})$$

et la longueur minimum de code associée, ou MDL, s'écrit

$$\begin{aligned} \mathrm{MDL}(\mathcal{X}, \pi) &= \min_{\theta \in \Theta} h(\mathcal{X}, \boldsymbol{\theta}) \\ &= h(\mathcal{X}, \hat{\boldsymbol{\theta}}), \end{aligned}$$

$\hat{\boldsymbol{\theta}}$ étant la valeur de $\boldsymbol{\theta}$ qui minimise la longueur de code $h(\mathcal{X}, \boldsymbol{\theta})$.

Remarque sur les données avec labels Lorsque les labels $\boldsymbol{z}$ sont connus, le texte est composé du couple $(\mathcal{X}, \boldsymbol{z})$ et chaque élément $(\boldsymbol{x}_i, z_i)$ est représenté dans l'alphabet $\mathcal{A}$. La fonction fréquence f porte sur les couples $(\boldsymbol{x}, k)$ de $\mathcal{A}$. La longueur de code s'écrit alors

$$h(\mathcal{X}, \boldsymbol{z}, \boldsymbol{\theta}) = -\mathrm{L}(\boldsymbol{\theta}) - \ln \pi(\boldsymbol{\theta})$$

avec $\mathrm{L}(\boldsymbol{\theta})$ la vraisemblance avec labels connus définie en (2.3). On pourrait aussi étendre ce principe aux données dont les labels sont partiellement connus.

Extension hiérarchique : HMDL

La densité $\pi(\boldsymbol{\theta})$ peut elle-même dépendre d'hyperparamètres $\boldsymbol{\psi} \in \Psi$ et on a donc $\pi(\boldsymbol{\theta}) = \pi_1(\boldsymbol{\theta}|\boldsymbol{\psi})$. L'ensemble Ψ doit alors être codé suivant une loi a priori $\pi_2(\boldsymbol{\psi})$ sur les hyperparamètres. La longueur du code s'écrit alors

$$h(\mathcal{X}, \boldsymbol{\theta}, \boldsymbol{\psi}) = -\mathrm{L}(\boldsymbol{\theta}) - \ln \pi_1(\boldsymbol{\theta}|\boldsymbol{\psi}) - \ln \pi_2(\boldsymbol{\psi})$$

et HMDL s'écrit

$$\begin{aligned}\mathrm{HMDL}(\mathcal{X},\pi_1,\pi_2) &= \min_{\theta\in\Theta,\psi\in\Psi} h(\mathcal{X},\boldsymbol{\theta},\boldsymbol{\psi}) \\ &= h(\mathcal{X},\hat{\boldsymbol{\theta}},\hat{\boldsymbol{\psi}}),\end{aligned}$$

avec $\hat{\boldsymbol{\theta}}$ et $\hat{\boldsymbol{\psi}}$ valeurs respectives de $\boldsymbol{\theta}$ et $\boldsymbol{\psi}$ qui minimisent la longueur de code $h(\mathcal{X},\boldsymbol{\theta},\boldsymbol{\psi})$. On pourra se reporter à Bryant [29].

MDL dans le cas continu

Replaçons nous dans le contexte classique avec n données $\mathcal{X}=(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n)$ dans $\mathbb{R}^d$ et $\boldsymbol{\theta}\in\mathbb{R}^q$. L'approche MDL définie précédemment n'est valable que pour des objets discrets à coder, donc des $\boldsymbol{x}_i$ et $\boldsymbol{\theta}$ discrets. Suivant la démarche de Rissanen p.54 et de Bryant [29], nous *tronquons* les $\boldsymbol{x}_i$ et $\boldsymbol{\theta}$ à une précision finie. La j^{e} $(1\leq j\leq d)$ composante de chaque $\boldsymbol{x}_i$ $(1\leq i\leq n)$ est tronquée à la précision $\epsilon_j=2^{r_j}$ où r_j est le nombre de symbôles binaires pris dans la troncature. De même, la l^{e} $(1\leq l\leq q)$ composante de $\boldsymbol{\theta}$ est tronquée à la précision $\delta_l=2^{s_l}$. On approxime alors les éléments différentiels infinitésimaux $d\boldsymbol{x}$ et $d\boldsymbol{\theta}$ par

$$d\boldsymbol{x}\approx\prod_{j=1}^{d}\epsilon_j \;\text{ et }\; d\boldsymbol{\theta}\approx\prod_{l=1}^{q}\delta_l.$$

Notant $f(\boldsymbol{x}|\boldsymbol{\theta})$ et $\pi(\boldsymbol{\theta})$ les densités respectives de $\boldsymbol{x}$ et $\boldsymbol{\theta}$ dans le cas continu, elles sont remplacées respectivement dans le cas discret par les probabilités

$$f(\boldsymbol{x}|\boldsymbol{\theta})d\boldsymbol{x} \;\text{ et }\; \pi(\boldsymbol{\theta})d\boldsymbol{\theta}.$$

On obtient ainsi

$$h(\mathcal{X},\boldsymbol{\theta},\epsilon,\delta)=-\mathrm{L}(\boldsymbol{\theta})-\ln\pi(\boldsymbol{\theta})-\sum_{j=1}^{d}\ln\epsilon_j-\sum_{l=1}^{q}\ln\delta_l,$$

et

$$\mathrm{MDL}(\mathcal{X},\pi,\epsilon,\delta)=\min_{\theta\in\Theta}h(\mathcal{X},\boldsymbol{\theta},\epsilon,\delta) \tag{2.15}$$

avec $\hat{\boldsymbol{\theta}}$ la valeur de $\boldsymbol{\theta}$ qui minimise $h(\mathcal{X},\boldsymbol{\theta})$. Cette valeur est elle aussi tronquée. On pourra remarquer une similitude avec l'estimateur de Bayes sous coût 0–1 donné à l'équation 2.12. Une différence importante apparaît néanmoins entre les deux approches, les bayésiens n'ayant pas de terme de précision.

Dans le cas asymptotique, Rissanen a montré (p.56) que

$$2\mathrm{MDL}(\mathcal{X},\pi,\epsilon,\delta)\approx 2\mathrm{MDL}(\mathcal{X})=-2\max_{\theta\in\Theta}\mathrm{L}(\boldsymbol{\theta})+q\ln(n). \tag{2.16}$$

Ainsi $\mathrm{MDL}(\mathcal{X},\pi,\epsilon,\delta)$ ne dépend plus ni de la loi π, ni des précisions et, asymptotiquement, minimiser la longueur de code revient à maximiser la vraisemblance.

Remarque sur les données avec labels Lorsque les labels $\boldsymbol{z}$ sont connus, on utilise la même démarche pour obtenir

$$2\mathrm{MDL}(\mathcal{X}, \boldsymbol{z}) = -2 \max_{\theta \in \Theta} \mathrm{L}(\boldsymbol{\theta}) + q \ln(n)$$

avec $\mathrm{L}(\boldsymbol{\theta})$ vraisemblance de l'équation 2.3.

Application aux mélanges et lien avec Bayes

Prenons la densité $f(\boldsymbol{x}|\boldsymbol{\theta})$ égale à la densité de mélange donnée par l'équation 1.2 du chapitre 1. À précisions ϵ_j et δ_l fixées, on minimise $h(\mathcal{X}, \boldsymbol{\theta}, \epsilon, \delta)$ sur $\boldsymbol{\theta}$ pour obtenir une valeur $\hat{\boldsymbol{\theta}}$ (tronquée à la précision près comme on l'a vu). Le choix de la densité $\pi(\boldsymbol{\theta})$ peut se faire en remarquant le lien étroit au niveau conceptuel qui existe entre cette loi et la loi a priori sur les paramètres des bayésiens. On peut donc prendre $\pi(\boldsymbol{\theta})$ comme une loi conjuguée à la manière bayésienne (voir la section 2.2.3). Dans le cas d'un mélange gaussien, les lois a priori dépendrons de la paramétrisation des matrices de variance $\boldsymbol{\Sigma}_k$ tout comme dans le cas bayésien.

Le problème de l'optimisation peut se faire itérativement et on pourra se reporter aux travaux de Bryant [29]. Pour éviter ce problème d'optimisation, de choix de la loi de codage des paramètres et aussi celui des précisions, il est toujours possible de se ramener au cas asymptotique donné par l'équation 2.16. On aura alors une approximation grossière du «vrai» MDL, approximation qui peut parfois s'avérer suffisante.

2.3 Estimation mixte

Rappelons que l'estimation mixte est utilisée en classification automatique et a pour objectif d'estimer conjointement le paramètre du mélange $\boldsymbol{\theta}$ et la partition $\boldsymbol{z}$. Maximiser la vraisemblance classifiante pour atteindre cet objectif est alors très classique dans cette situation.

2.3.1 Maximum de vraisemblance classifiante

Définition de la vraisemblance classifiante

La matrice de partition $\boldsymbol{z}$ est explicitement considérée comme un paramètre supplémentaire inconnu. Chaque couple $(\boldsymbol{x}_i, z_i)$ $(1 = 1 \ldots, n)$ avec $z_i \in \{1, \ldots, K\}$ est supposé être une réalisation i.i.d. du couple de variables aléatoires $(\boldsymbol{X}, Z)$ de densité $f(\boldsymbol{x}, k|\boldsymbol{\theta})$. La densité jointe $f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta})$ des couples de variables $(\boldsymbol{X}_i, Z_i)$ $(1 = 1 \ldots, n)$ est alors le produit de la densité de chaque couple (équation 1.8). Elle correspond à la vraisemblance classifiante (*classification likelihood*) de $(\boldsymbol{\theta}, \boldsymbol{z})$. Son logarithme naturel s'identifie à la log-vraisemblance classifiante de Symons

[86]

$$\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{z}) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \ln(p_k \phi_k(\boldsymbol{x}_i | \boldsymbol{a}_k)). \tag{2.17}$$

On remarque que cette vraisemblance est identique à la vraisemblance de $\boldsymbol{\theta}$ obtenue en (2.3) quand les labels étaient connus.

Il existe aussi, ce que l'on appelle la vraisemblance classifiante restreinte (notée CL_R) et qui correspond implicitement à la vraisemblance classifiante sous la contrainte de proportions égales. CL_R fut antérieure à CL (voir Scott et Symons, 1971 [82]) et s'écrit :

$$\mathrm{CL}_R(\boldsymbol{\theta}, \boldsymbol{z}) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \ln(\phi_k(\boldsymbol{x}_i | \boldsymbol{a}_k)).$$

En fait CL peut se voir comme une version pénalisée de CL_R (Celeux et Govaert, 1992 [36]) :

$$\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{z}) = \mathrm{CL}_R(\boldsymbol{\theta}, \boldsymbol{z}) + \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \ln p_k.$$

Propriétés du maximum de vraisemblance classifiante

Il est à noter que, contrairement au maximum de vraisemblance, l' estimateur $\hat{\boldsymbol{\theta}}$ du maximum de vraisemblance classifiante est inconsistant (Bryant et Williamson [30]). Donc, lorsque la taille de l'échantillon tend vers l'infini, $\hat{\boldsymbol{\theta}}$ ne tend pas nécessairement vers $\boldsymbol{\theta}$: L'estimateur est biaisé, même asymptotiquement.

La maximisation de la vraisemblance classifiante $\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{z})$ est typiquement une *démarche de classification* puisque qu'elle consiste à chercher non seulement un estimateur des paramètres $\boldsymbol{\theta}$, mais aussi et conjointement un estimateur des appartenances $\boldsymbol{z}$. Celeux et Govaert [38] ont pour cette raison appelé *approche classification* l'estimation des paramètres par maximum de vraisemblance classifiante, et *approche mélange* l'estimation par maximum de vraisemblance.

De même que pour chercher le maximum de vraisemblance, les méthodes analytiques échouent généralement pour estimer les paramètres optimums. Des recours à des méthodes itératives, comme les algorithmes CEM ou CAEM sont donc une planche de salut dans ce cas.

Lien avec des critères classiques

Maximiser la vraisemblance classifiante est équivalent dans certains cas à optimiser des critères classiques qui, parfois, ne faisaient pas appel explicitement à celle-ci, ni d'ailleurs à une hypothèse de mélange gaussien. Ces critères étaient généralement construits à partir de notions assez géométriques. C'est le cas par exemple du critère d'inertie intraclasse ou des k-means $\mathrm{tr}(\boldsymbol{W}(\boldsymbol{z}))$ avec $\boldsymbol{W}(\boldsymbol{z})$ matrice d'inertie intraclasse (équation 2.4). Celeux et Govaert [36] ont montré que la partition obtenue par minimisation de ce critère et la partition obtenue par maximisation de la vraisemblance classifiante avec modèle $[p\lambda I]$ étaient identiques.

En effet, la vraisemblance classifiante dans le cas gaussien général en utilisant la décomposition spectrale des matrices de variance (section 1.4.2) s'exprime par

$$\begin{aligned}\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{z}) &= -\frac{1}{2}\sum_{k=1}^{K}\frac{1}{\lambda_k}\sum_{i=1}^{n} z_{ik}(\boldsymbol{x}_i-\boldsymbol{\mu}_k)'\boldsymbol{D}_k\boldsymbol{A}_k^{-1}\boldsymbol{D}_k'(\boldsymbol{x}_i-\boldsymbol{\mu}_k)\\ &\quad -\frac{d}{2}\sum_{k=1}^{K} n_k\ln(\lambda_k)+\sum_{k=1}^{K} n_k\ln(p_k)-\frac{nd}{2}\ln(2\pi).\end{aligned}$$

Sous les contraintes du modèle sphérique $[p\lambda I]$, on obtient

$$\begin{aligned}\mathrm{CL}_{[p\lambda I]}(\boldsymbol{\theta}, \boldsymbol{z}) &= -\frac{1}{2\lambda}\sum_{k=1}^{K}\sum_{i=1}^{n} z_{ik}(\boldsymbol{x}_i-\boldsymbol{\mu}_k)'(\boldsymbol{x}_i-\boldsymbol{\mu}_k)\\ &\quad -\frac{nd}{2}\ln(\lambda)-n\ln(K)-\frac{nd}{2}\ln(2\pi).\end{aligned}$$

Soient $\hat{\boldsymbol{z}}_{CL}$ et $\hat{\boldsymbol{\theta}}_{CL}$ les estimateurs respectivement de $\boldsymbol{z}$ et $\boldsymbol{\theta}$ par maximisation de la vraisemblance classifiante. Nous sommes certains que $\hat{\boldsymbol{\theta}}_{CL}$ maximise $\mathrm{CL}(\boldsymbol{\theta}, \hat{\boldsymbol{z}}_{CL})$ donc que $\hat{\boldsymbol{\theta}}_{CL}$ est l'estimateur du maximum de vraisemblance $\mathrm{CL}(\boldsymbol{\theta}, \hat{\boldsymbol{z}}_{CL})$ quand $\hat{\boldsymbol{z}}_{CL}$ est connue. Sinon $\mathrm{CL}(\hat{\boldsymbol{\theta}}_{CL}, \hat{\boldsymbol{z}}_{CL})$ ne serait pas optimum. Ainsi, exprimons, pour toute partition $\boldsymbol{z}$, l'estimateur du maximum de vraisemblance $\hat{\boldsymbol{\theta}}$ associé à cette partition. On a par simple calcul

$$\hat{\boldsymbol{\mu}}_k = \frac{\sum_{i=1}^{n} z_{ik}\boldsymbol{x}_i}{n_k}\ (1 \leq k \leq K) \qquad \text{et} \qquad \hat{\lambda} = \frac{\mathrm{tr}(\boldsymbol{W}(\boldsymbol{z}))}{nd}.$$

Utilisant ces estimateurs dans $\mathrm{CL}_{[p\lambda I]}$, on a en utilisant la linéarité de la trace et le fait que $\mathrm{tr}((\boldsymbol{x}_i-\boldsymbol{\mu}_k)(\boldsymbol{x}_i-\boldsymbol{\mu}_k)') = (\boldsymbol{x}_i-\boldsymbol{\mu}_k)'(\boldsymbol{x}_i-\boldsymbol{\mu}_k)$:

$$\mathrm{CL}_{[p\lambda I]}(\hat{\boldsymbol{\theta}}, \boldsymbol{z}) = -\frac{nd}{2}\ln(\mathrm{tr}(\boldsymbol{W}(\boldsymbol{z}))) + \mathrm{cste}.$$

Ainsi, minimiser le critère des k-means sur la partition $\boldsymbol{z}$ revient à maximiser la vraisemblance classifiante sur $(\boldsymbol{\theta}, \boldsymbol{z})$ avec le modèle $[p\lambda I]$. A l'optimum, on a

$$\mathrm{CL}_{[p\lambda I]}(\hat{\boldsymbol{\theta}}_{CL}, \hat{\boldsymbol{z}}_{CL}) = -\frac{nd}{2}\ln(\mathrm{tr}(\boldsymbol{W}(\hat{\boldsymbol{z}}_{CL}))) + \mathrm{cste}. \tag{2.18}$$

D'autres exemples de rapprochement entre le critère de vraisemblance classifiante et divers critères classiques sont rapportés dans le tableau 2.1.

2.3.2 Algorithmes maximisant la vraisemblance classifiante

Nous présentons deux algorithmes utilisés pour maximiser la vraisemblance classifiante : CEM et CAEM. La vraisemblance classifiante étant une généralisation de critères classiques, ces deux algorithmes peuvent aussi s'interpréter

Critère d'inertie	Origine	Modèle
$\mathrm{tr}(\boldsymbol{W})$	critère des k-means Ward (1963) [89]	$[p\lambda I]$
$\lvert\boldsymbol{W}\rvert$	Friedman et Rubin (1967) [49]	$[p\lambda C]$
$\sum_{k=1}^{K} n_k \ln \left\lvert\frac{\boldsymbol{W}_k}{n_k}\right\rvert$	Scott et Symons (1971) [82]	$[p\lambda_k C_k]$

TAB. 2.1 – *Liens entre la vraisemblance classifiante et des critères classiques d'origine non probabiliste.*

comme une généralisation des algorithmes associés à ces critères. C'est le cas, par exemple, de l'algorithme des k-means. Les algorithmes CEM et CAEM sont donc plus courants que l'algorithme EM lors d'une démarche de classification.

L'algorithme CEM

Présentation L'algorithme CEM (*Classification* EM) est une version classification de l'algorithme EM proposée par Celeux et Govaert [36].

La procédure part d'une partition initiale des données $\boldsymbol{z}^\circ$ (on commence alors par l'étape M) ou bien d'un vecteur de paramètre initial $\boldsymbol{\theta}^\circ$. Ensuite, les étapes de la m^{e} itération sont :

1. **Étape E (estimation) :** estimation des probabilités t_{ik}^m pour chaque $\boldsymbol{x}_i$ sachant $\boldsymbol{\theta}^{m-1}$;

2. **Étape C (classification) :** estimation de tous les z_{ik}^m (donc $\boldsymbol{z}^m$) par MAP des t_{ik}^m. Une partition $\boldsymbol{c}^m$ est déduite ($\boldsymbol{c}^m = \boldsymbol{z}^m$) ;

3. **Étape M (maximisation) :** maximisation de $\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{c}^m)$ sachant $\boldsymbol{c}^m$ pour trouver $\boldsymbol{\theta}^m$ optimum (voir EM).

L'algorithme s'arrête lorsque la stationnarité de la vraisemblance classifiante est atteinte. Un autre critère d'arrêt peut aussi être la stabilité de la partition ($\boldsymbol{z}^{m+1} = \boldsymbol{z}^m$).

Croissance du critère Chaque itération fait croître le critère CL et la stationnarité est atteinte en un nombre fini d'itérations.

À l'itération m, les étapes E et C maximisent $\mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z})$ sachant $\boldsymbol{\theta}^{m-1}$ pour trouver une partition $\boldsymbol{z}^m$. En effet, la partition $\boldsymbol{z}^m$ retenue est définie par

$$z_{ik}^m = \begin{cases} 1 & \text{si} \quad k = \arg\max_k t_k(\boldsymbol{x}_i|\boldsymbol{\theta}^{m-1}) \\ 0 & \text{sinon} \end{cases} .$$

On démontre alors que $\boldsymbol{z}^m$ maximise $\mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z})$ (propriété $(*)$) par les équivalences suivantes

$$\begin{aligned}
(*) \quad &\Leftrightarrow \quad \mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z}^m) = \max_z \mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z}) \\
(*) \quad &\Leftrightarrow \quad \forall \boldsymbol{z}, \quad \mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z}^m) \geq \mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z}) \\
(*) \quad &\Leftrightarrow \quad \forall \boldsymbol{z}, \quad \sum_{i=1}^{n} \sum_{k=1}^{K} (z_{ik}^m - z_{ik}) \ln(p_k^{m-1} \phi_k(\boldsymbol{x}_i | \boldsymbol{a}_k^{m-1})) \geq 0 \\
(*) \quad &\Leftrightarrow \quad \forall \boldsymbol{z}, \quad \sum_{i=1}^{n} \sum_{k=1}^{K} (z_{ik}^m - z_{ik}) \ln(t_k(\boldsymbol{x}_i | \boldsymbol{a}_k^{m-1})) \geq 0 \\
(*) \quad &\Leftrightarrow \quad \forall \boldsymbol{z}, \quad \sum_{i=1}^{n} \{\max_k \ln(t_k(\boldsymbol{x}_i | \boldsymbol{a}_k^{m-1})) - \ln(t_{z_i}(\boldsymbol{x}_i | \boldsymbol{a}_k^{m-1}))\} \geq 0
\end{aligned}$$

D'où

$$\mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z}^m) \geq \mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z}^{m-1}).$$

L'étape M maximise ensuite $\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{z}^m)$ pour trouver $\boldsymbol{\theta}^m$:

$$\mathrm{CL}(\boldsymbol{\theta}^m, \boldsymbol{z}^m) \geq \mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z}^m).$$

Les deux dernières inégalités impliquent finalement la croissance du critère

$$\mathrm{CL}(\boldsymbol{\theta}^m, \boldsymbol{z}^m) \geq \mathrm{CL}(\boldsymbol{\theta}^{m-1}, \boldsymbol{z}^{m-1}).$$

Propriétés de convergence Comme EM, CEM converge aussi vers un maximum local. Il peut produire des estimateurs biaisés des paramètres du mélange (voir Marriott [67] et Bryant [28]). Ce biais peut être très sévère parfois mais reste tolérable si les classes sont bien séparées et si les proportions ne sont pas trop extrêmes. Ces résultats ont été établis par les études de Bryant et Williamson [31], Bryant [28], Ganesalingam [51] et Celeux et Govaert [37]. Le biais résulte de la «partition dure» réalisée itérativement à l'étape C.

Si les proportions sont très différentes, CEM a tendance aussi à vider les classes peu peuplées. Mais, cet algorithme donne des résultats satisfaisants d'un point de vue classification si les classes sont bien séparées et si les proportions sont peu différentes. Notons aussi que la convergence de CEM est généralement beaucoup plus rapide que celle de EM : une douzaine d'itérations parfois contre plusieurs centaines pour EM.

Stratégies d'utilisation Pour réduire la dépendance de l'algorithme CEM vis à vis de la partition initiale $\boldsymbol{z}^\circ$ ou des paramètres initiaux $\boldsymbol{\theta}^\circ$, on peut encore faire plusieurs essais avec des partitions initiales aléatoires ou avec des paramètres initiaux aléatoires (comme pour EM), et ne retenir que l'initialisation donnant le critère le meilleur parmi tous ceux obtenus. Cette solution est moins coûteuse en temps que pour EM, vu la plus grande vitesse de convergence de CEM. Remarquons que le tirage d'une partition aléatoire n'est pas vraiment conseillé, car cela revient à choisir des centres presques égaux au centre de l'échantillon $\mathcal{X}$.

Toujours pour éviter la dépendance de la position initiale, Celeux et Govert (1992) [36] proposent d'utiliser l'algorithme SEM (voir section 2.2.2) pour une douzaine d'itérations, suivi de CEM à partir de la position fournissant la plus grande valeur de CL parmi les itérations de SEM. L'algorithme CAEM (voir section 2.3.2) a aussi été développé dans ce but.

Rapprochement avec EM On a vu que l'algorithme EM maximisait le critère $\mathrm{L}(\boldsymbol{\theta}, \boldsymbol{c})$ (relation 2.6) sur le couple $(\boldsymbol{\theta}, \boldsymbol{c})$ avec $\boldsymbol{c}$ partition floue et que ceci était équivalent à maximiser la vraisemblance $\mathrm{L}(\boldsymbol{\theta})$ sur $\boldsymbol{\theta}$ uniquement. On peut alors voir CEM comme un algorithme maximisant aussi le critère $\mathrm{L}(\boldsymbol{\theta}, \boldsymbol{c})$ sur le couple $(\boldsymbol{\theta}, \boldsymbol{c})$ mais avec *la contrainte* que $\boldsymbol{c}$ soit une partition : $c_{ik} \in \{0, 1\}$.

Comme corrolaire de cette remarque, EM et CEM doivent être équivalents dans le cas de classes vraiment bien séparées. D'après la relation de Hathaway 2.6, on a

$$\mathrm{L}(\boldsymbol{\theta}) = \mathrm{C}(\boldsymbol{\theta}, \boldsymbol{t}(\boldsymbol{\theta})) + \mathrm{E}(\boldsymbol{t}(\boldsymbol{\theta})).$$

Notons $\hat{\boldsymbol{\theta}}_L$ l'estimateur de $\boldsymbol{\theta}$ par maximum de vraisemblance et $\hat{\boldsymbol{\theta}}_{CL}$ et $\hat{\boldsymbol{z}}_{CL}$ les estimateurs respectifs de $\boldsymbol{\theta}$ et $\boldsymbol{z}$ par maximum de vraisemblance classifiante. Si les classes sont vraiment bien séparées, EM doit fournir une partition proche de la partition de CEM : $\boldsymbol{t}(\hat{\boldsymbol{\theta}}_L) \approx \hat{\boldsymbol{z}}_{CL}$, et alors $\mathrm{E}(\boldsymbol{t}(\hat{\boldsymbol{\theta}}_L)) \approx 0$. Ainsi

$$\begin{aligned}
\hat{\boldsymbol{\theta}}_L &= \arg\max_{\theta} \mathrm{C}(\boldsymbol{\theta}, \boldsymbol{t}(\boldsymbol{\theta})) \\
&\approx \arg\max_{\theta} \mathrm{C}(\boldsymbol{\theta}, \hat{\boldsymbol{z}}_{CL}) \\
&= \arg\max_{\theta} \mathrm{CL}(\boldsymbol{\theta}, \hat{\boldsymbol{z}}_{CL}) \\
&= \hat{\boldsymbol{\theta}}_{CL}.
\end{aligned}$$

Il y a donc égalité des paramètres. On a aussi l'égalité des critères :

$$\begin{aligned}
\mathrm{L}(\hat{\boldsymbol{\theta}}_L) &\approx \mathrm{L}(\hat{\boldsymbol{\theta}}_{CL}) \\
&\approx \mathrm{C}(\hat{\boldsymbol{\theta}}_{CL}, t(\hat{\boldsymbol{\theta}}_{CL})) \\
&\approx \mathrm{C}(\hat{\boldsymbol{\theta}}_{CL}, \hat{\boldsymbol{z}}_{CL}) \\
&= \mathrm{CL}(\hat{\boldsymbol{\theta}}_{CL}, \hat{\boldsymbol{z}}_{CL}).
\end{aligned}$$

Comparaison avec EM Comparons par des simulations de Monte-Carlo le comportement de l'estimateur du paramètre du mélange obtenu par EM et CEM et l'estimateur de la partition obtenu par ces deux algorithmes.

Considérons des données provenant de deux classes normales bivariées avec des proportions égales ($p_1 = p_2 = 1/2$) et avec des matrices de variances sphériques égales ($\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \boldsymbol{I}$). Nous fixons le centre de la première classe à $\boldsymbol{\mu}_1 = (0, 0)'$

et celui de la deuxième à $\boldsymbol{\mu}_2 = (t, 0)'$. Le paramètre t règle la séparation des deux classes: nous prenons $t = 1.0$ pour une séparation faible (cela correspond à une probabilité d'erreur $\mathrm{P}_e(u_\theta) \approx 0.30$), $t = 2.0$ pour une séparation moyenne ($\mathrm{P}_e(u_\theta) \approx 0.15$) et $t = 3.3$ pour une séparation forte ($\mathrm{P}_e(u_\theta) \approx 0.05$). Pour chaque jeu de paramètre associé aux trois séparations, nous générons 30 échantillons de taille 20, 30 échantillons de taille 200 et 30 échantillons de taille 2000. Pour chaque échantillon, nous lançons EM et CEM initialisés avec les vrais paramètres et utilisant le modèle linéaire $[p\lambda C]$. Nous calculons donc $\hat{\boldsymbol{\theta}}_L$, $\hat{\boldsymbol{z}}_L$ (par MAP), $\hat{\boldsymbol{\theta}}_{CL}$ et $\hat{\boldsymbol{z}}_{CL}$, puis nous estimons les divergences de Kullback-Leibler (équation 1.10) $\mathrm{KL}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_L)$ et $\mathrm{KL}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}}_{CL})$ et les erreurs de classement (équation 1.11) erreur$(\hat{\boldsymbol{z}}_L)$ et erreur$(\hat{\boldsymbol{z}}_{CL})$. Nous moyennons alors ces résultats sur chacun des 30 échantillons et nous les consignons dans le tableau 2.2.

n	séparation	$\hat{\mathrm{KL}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$		erreur$(\hat{\boldsymbol{z}})$	
		EM	CEM	EM	CEM
20	faible	0.2770	0.2771	0.3383	0.3217
	moyenne	0.4916	0.3699	0.2050	0.1700
	forte	0.4108	0.3132	0.0983	0.0667
200	faible	0.0209	0.0822	0.3342	0.3188
	moyenne	0.0187	0.0425	0.1638	0.1587
	forte	0.0172	0.0209	0.0530	0.0500
2000	faible	0.0014	0.0454	0.3112	0.3113
	moyenne	0.0017	0.0246	0.1620	0.1619
	forte	0.0017	0.0059	0.0509	0.0510

TAB. 2.2 – *Comportement de $\hat{\boldsymbol{\theta}}$ et $\hat{\boldsymbol{z}}$ avec les algorithmes EM et CEM.*

Pour peu de données ($n = 20$) l'algorithme CEM est meilleur que EM pour l'estimation du paramètre du mélange et de la partition. L'explication résulte de la seule différence existant entre les deux algorithmes: l'étape de classification C. Cette étape permet d'obtenir une partition et traduit le fait que l'on pense qu'il y a bien présence d'une partition. On peut voir cette croyance comme une «sorte» *d'information a priori.* Celle-ci permettrait alors de rendre l'estimation de $(\boldsymbol{\theta}, \boldsymbol{z})$ plus robuste pour de petits échantillons. Pour $n = 200$, CEM reste meilleur pour estimer la partition, bien que cette domination soit moins marquée. Pour $n = 2000$, il y a très peu de différence d'estimation de la partition entre EM et CEM. On peut penser que l'information a priori introduite par l'étape C devient négligeable par rapport à l'information apportée par l'échantillon. Concernant l'estimation du paramètre du mélange, pour $n = 200, 2000$ et des classes peu séparées, on remarque le fort biais introduit par CEM.

L'algorithme CAEM

Présentation L'algorithme CAEM (*Classification Annealing* EM) de Celeux et Govaert [36] est une version de type *recuit simulé* de CEM. Son objectif est encore de maximiser la vraisemblance classifiante tout en s'affranchissant cette fois de la dépendance des paramètres initiaux. Comme l'algorithme SAEM (voir section 2.2.2), CAEM est aussi fondé sur SEM (voir section 2.2.2) mais, pour obtenir des estimateurs ponctuels des paramètres du mélange, la variance des affectations aléatoires introduites à l'étape C (étape de classification *stochastique*) tendent vers 0 quand le nombre d'itérations tend vers l'infini. On utilise une séquence de températures réelles positives $(\gamma^m, m \geq 0)$ décroissant vers 0 quand $m \to \infty$ à partir de $\gamma^\circ = 1$.

La procédure part d'une partition initiale des données $\boldsymbol{z}^\circ$ (on commence alors par l'étape M) ou bien d'un vecteur de paramètre initial $\boldsymbol{\theta}^\circ$, avec $\gamma^\circ = 1$. Ensuite, les étapes de la m^{e} itération sont :

1. **Étape E (estimation) :** estimation des probabilités t_{ik}^m pour chaque $\boldsymbol{x}_i$ sachant $\boldsymbol{\theta}^{m-1}$;

2. **Étape A (annealing) :** calcul des quantités $r_{ik}^m = \frac{(t_{ik}^m)^{\frac{1}{\gamma^{m-1}}}}{\sum_{k=1}^K (t_{ik}^m)^{\frac{1}{\gamma^{m-1}}}}$. Une nouvelle température γ^m est calculée ;

3. **Étape C (classification stochastique) :** tirage pour chaque $\boldsymbol{x}_i$ de la variable $\boldsymbol{z}_i^m = (z_{ik}^m, k = 1, \ldots, K)$ selon une loi multinomiale d'ordre un et de paramètre $(r_{ik}^m, k = 1, \ldots, K)$. On définit une partition $\boldsymbol{c}^m$ à partir des $\boldsymbol{z}_i^m$;

4. **Étape M (maximisation) :** maximisation de $\text{CL}(\boldsymbol{\theta}, \boldsymbol{c}^m)$ sachant $\boldsymbol{c}^m$ pour trouver $\boldsymbol{\theta}^m$ optimum (voir EM).

Les critères d'arrêt sont les mêmes que l'algorithme CEM. L'algorithme CAEM commence donc par l'algorithme *stochastique* SEM pour tendre vers l'algorithme *déterministe* CEM quand le nombre d'itérations tend vers l'infini. On remarquera que l'étape C de CAEM est en fait identique à l'étape S de SEM : il s'agit d'une classification stochastique.

Propriétés de convergence À notre connaissance, il n'a pas encore été prouvé que n'importe quelle séquence de CAEM convergeait presque sûrement vers un optimum local de la vraisemblance classifiante (contrairement à SAEM pour la vraisemblance). Malgré tout, CAEM semble souvent éviter des solutions suboptimales de CEM. Il a néanmoins l'inconvénient de nécessiter un nombre important d'itérations pour espérer obtenir un bon résultat.

Comme SAEM, le choix de la suite (γ^m) est important pour le comportement de l'algorithme et il est recommandé d'avoir une vitesse de convergence vers 0 assez lente. La règle choisie par les auteurs de CAEM est géométrique, $\gamma^{m+1} = a\gamma^m$ avec $0.9 \leq a < 1$. Le coefficient linéaire a est ainsi choisi pour bien obtenir une vitesse de convergence assez lente.

Quant à l'utilisation de CAEM, Celeux et Govaert recommendent son emploi plutôt pour de petits échantillons en préférant SEM dans le cas de plus grands.

2.4 Conclusion

Nous avons vu qu'il existait des points communs entre l'algorithme EM et les quatre variantes SEM, SAEM, CEM et CAEM. Les étapes E (expectation) et M (maximization) sont la base de tous ces algorithmes et les seuls changements reposent sur l'ajout d'étapes propres à ceux-ci : étape C de classification pour CEM, étape S de classification stochastique pour SEM, etc. Pour cette raison, nous avons développé un logiciel dans le langage Splus, le logiciel XEM, qui regroupe ces cinq algorithmes dans un *seul programme* en tirant partie des points communs des algorithmes. De plus, ce programme permet différentes initialisations des algorithmes (partition fixée, partition aléatoire, etc). Il permet aussi de lancer plusieurs algorithmes à la suite (partir de SEM puis utiliser la partition trouvée pour lancer CEM par exemple). Ce logiciel est présenté dans l'annexe A.

Ainsi nous disposons d'un outil d'estimation qui tient compte d'informations a priori de l'utilisateur que sont le modèle gaussien (en discrimination et en classification automatique) ou le nombre de classes (en classification automatique uniquement). Le chapitre qui suit propose de faire un choix entre plusieurs a priori de l'utilisateur. Il regroupe des outils classiques de choix de modèles et en présente de nouveaux.

Chapitre 3

Choix d'un modèle

3.1 Introduction

Jusqu'à présent, l'automatisation des démarches de classification et de classement n'a été que partielle lorsque le modèle sous-jacent est le modèle de mélange gaussien : en discrimination, il reste nécessaire de choisir le modèle gaussien et en classification automatique, il faut de plus choisir le nombre de classes. Supposant que l'utilisateur hésite entre plusieurs modèles, ce chapitre présente des outils classiques et des méthodes originales de choix de l'un d'entre eux. Pour trouver le nombre de classes en classification automatique, nous proposons d'introduire une notion de précision sur les données pour détecter un coude dans la courbe du maximum de vraisemblance. Une autre approche très intéressante pour trouver le nombre de classes ou le modèle gaussien en classification automatique, consiste à pénaliser la vraisemblance par la classifiabilité des données : la simplicité ainsi que la riche interprétation de ces critères (on peut établir un lien avec la vraisemblance classifiante et le critère des k-means) semblent indiquer une approche prometteuse et originale. En discrimination, nous proposons simplement d'adapter le critère classique de validation croisée au choix du modèle gaussien.

Après avoir rappelé la position du problème à la section 3.2 (définition rigoureuse d'un modèle par exemple), dans la section 3.3 nous étudions l'objectif de choix de modèle dans un contexte exclusivement mélange, c'est-à-dire sans référence ni à la classification automatique, ni à la discrimination. Dans cette situation, nous décrivons la théorie classique des tests d'hypothèses. Nous remarquons qu'elle n'est pas facilement applicable au choix de plusieurs modèles et qu'elle est même à proscrire pour choisir le nombre de classes bien que certains auteurs aient néanmoins proposé des adaptations. Nous préférons à ces approches l'utilisation de critères numériques, souvent appelés *critères d'information*. Certains, comme BIC ou AWE, sont établis dans le contexte de la théorie bayésienne, d'autres s'appuient sur des justifications de statistique classique (AIC, AIC3, ICOMP). Nous voyons les propriétés asymptotiques de certains d'entre eux. Nous proposons aussi une méthode originale pour trouver le nombre de classes. Il s'agit de détecter un coude dans la courbe du maximum de vraisemblance grâce à une hypothèse de précision sur les données.

La section 3.4 considère le choix de modèles dans le cadre de la classification automatique cette fois. Après avoir précisé que «l'approche mélange» était encore utilisable, nous étudions le problème de choix d'un modèle avec une «optique classification» : un modèle adapté doit conduire à des données bien classifiables. Nous rappelons des critères mesurant la classifiabilité des données (PC, MIR, NEC) et nous en proposons deux nouveaux (E et LP). Ces critères ont néanmoins l'inconvénient d'être rarement utilisables dans le cas monoclasse, c'est pourquoi nous proposons des critères qui pénalisent la vraisemblance par une mesure de la classifiabilité des données (C, CL et CLM). Nous voyons leur bon comportement sur quelques exemples et leur riche interprétation.

Enfin, la section 3.5 étudie le choix de modèles dans le cadre de la discrimination. L'approche mélange est toujours utilisable, mais nous présentons aussi la méthode de validation croisée, méthode qui s'inscrit plus profondément dans une «approche discrimination». Connue pour nécessiter le calcul des règles de classement autant de fois qu'il y a de points dans l'échantillon, processus coûteux, nous établissons alors des formules de mise à jour de ces règles pour certains des 28 modèles gaussiens.

3.2 Position du problème

Cette section introduit les concepts de base nécessaires pour présenter des méthodes de choix de modèles : définition d'un modèle, intérêt du choix et présentation de l'ensemble des modèles que nous considérons.

3.2.1 Définition d'un modèle

Dans le contexte d'un mélange gaussien, un modèle statistique de distribution

$$f(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{k=1}^{K_{sup}} p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k),$$

où K_{sup} est le nombre maximum de classes possible dans le mélange, est caractérisé par le paramètre $\boldsymbol{\theta}$ qui appartient à un espace $\Theta \subseteq \mathbb{R}^q$. On appellera *modèle* $\mathcal{M}$ la famille de modèles

$$\mathcal{M} = \{f(\boldsymbol{x}|\boldsymbol{\theta}) : \boldsymbol{\theta} \in \Theta\}.$$

Exemple Exprimons un modèle à K classes ($K \leq K_{sup}$) et avec contraintes $[p\lambda C]$. L'espace Θ associé est décrit dans le tableau 3.1. Nous posons par convention que les classes vides $K+1$ à K_{sup} ont même centre (placé à l'origine) et même matrice de variance (égale à l'identité). Bien sûr on pourrait prendre tout autre choix. Cette convention permet d'établir une relation d'inclusion entre les espaces Θ de certains modèles comme les modèles $[p_k\lambda C]$ à K et K' classes ($K \leq K' \leq K_{sup}$).

Paramètres	Classes	
	$k = 1, \ldots, K$	$k = K+1, \ldots, K_{sup}$
Proportions	$p_k = \frac{1}{K}$	$p_k = 0$
Centres	$\boldsymbol{\mu}_k \in \mathbb{R}^d$	$\boldsymbol{\mu}_k = \mathbf{0}$
Matrices de variance	$\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$	$\boldsymbol{\Sigma}_k = \boldsymbol{I}$

TAB. 3.1 – *Espace des paramètres associé au modèle $[p\lambda C]$ à K classes.*

Adoptant cette convention, le paramètre associé à un modèle s'écrit alors :

$$\boldsymbol{\theta} = (p_1, \ldots, p_K, 0, \ldots, 0, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_K, (\mathbf{0}, \boldsymbol{I}), \ldots, (\mathbf{0}, \boldsymbol{I}))$$

mais, pour simplifier, nous continuerons d'employer la notation

$$\boldsymbol{\theta} = (p_1, \ldots, p_K, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_K).$$

3.2.2 Nécessité de choisir un modèle

Une fois posée la définition précédente, nous précisons l'utilité de choisir les modèles gaussiens et le nombre de classes.

modèle gaussien

L'intérêt de choisir le modèle gaussien en classification automatique lorsque le nombre de classes est connu et en discrimination à déjà été vu à la section 1.4 du chapitre 1. En classification automatique, il s'agit d'estimer au mieux la partition $\boldsymbol{z}$ et en discrimination il s'agit d'estimer au mieux la règle de classement u. Voici l'effet que peuvent avoir de mauvais modèles :

- **modèle trop complexe** : on peut constater un phénomène de surparamétrisation. Le tableau 1.2 du chapitre 1 montre bien le manque de robustesse des estimateurs (partition et règle de classement) quand on utilise un modèle quadratique : l'erreur de classement erreur($\hat{\boldsymbol{z}}$) et la probabilité d'erreur de classement $\mathrm{P}_e(u)$ sont moins bons avec le modèle quadratique qu'avec un modèle plus simple ;

- **modèle trop simple** : il est évident qu'un modèle trop simple permettra rarement d'obtenir des estimateurs pertinents. Par exemple sur la figure 1.8 du chapitre 1, on voit clairement le problème si on utilise un modèle linéaire alors qu'en réalité il est quadratique : deux classes séparées par une frontière quadratique seront mal séparées par une frontière linéaire. Il ne suffit donc pas d'introduire un a priori sur les paramètres au travers des modèles gaussiens, encore faut-il s'assurer que cet a priori s'approche de la réalité.

Nombre de classes

Le choix du nombre de classes est tout aussi important. La classification automatique est une méthode d'analyse des données donc de réduction des données : on réduit notre ensemble de données (composé de plusieurs milliers d'éléments par exemple) à un résumé de seulement quelques classes (2 ou 3 par exemple). Ces classes doivent faire apparaître une structure sous-jacente aux données et ainsi permettre de faciliter leur interprétation. Chaque classe doit avoir une signification pour l'expert du domaine. Typiquement, deux situations se présentent :

- **trop de classes** : cette situation peut entraîner une grande confusion car certaines classes sont «artificielles», c'est-à-dire qu'elles ne représentent aucune réalité du domaine concerné ;

- **pas assez de classes** : cet autre cas peut cacher des aspects importants présents dans les données. Par exemple, on peut découper un ensemble de patients en deux groupes : les patients sains et malades. Mais il peut être plus intéressant pour le médecin d'utiliser une structure en trois classes faisant ressortir les patients sains, malades et à risque.

3.2.3 Modèles en compétition

La classification automatique et la discrimination ont pour objectif commun de rechercher l'automatisation d'une démarche : celle de la classification dans le premier cas et celle du classement dans le second. On a vu aux chapitres 1 et 2 plusieurs façons d'automatiser ces démarches lorsque le modèle $\mathcal{M}$ était connu. Ce modèle correspondait à un a priori de l'utilisateur. Mais l'utilisateur peut hésiter entre plusieurs modèles s'il ne dispose pas d'une information a priori précise sur un modèle. Le processus d'automatisation devra alors intégrer aussi le choix d'un modèle parmi ceux qui sont disponibles.

Nous sommes donc dans la situation où un ensemble Δ de plusieurs modèles sont en compétition. En discrimination le nombre de classes est connu et seul le modèle gaussien doit être choisi. Par contre, en classification automatique, cet ensemble de modèles peut correspondre à un choix entre plusieurs nombres de classes ou entre plusieurs modèles gaussiens ou enfin aux deux à la fois. Le tableau 3.2 donne des exemples de modèles en compétition.

Modèles en compétition		Exemple d'un ensemble Δ de modèles en compétition
Nombres de classes	Modèles gaussiens	
		$\{[p\lambda I]_{K=1}\}$ (un seul modèle)
	$\times$	$\{[p\lambda I]_{K=1}, [p\lambda_k C_k]_{K=1}\}$
$\times$		$\{[p\lambda I]_{K=1}, \ldots, [p\lambda I]_{K=K_{sup}}\}$
$\times$	$\times$	$\{[p\lambda I]_{K=1}, [p\lambda_k C_k]_{K=1}, \ldots, [p\lambda I]_{K=K_{sup}}, [p\lambda_k C_k]_{K=K_{sup}}\}$

TAB. 3.2 – *Exemples de modèles en compétition.*

3.3 Approche mélange

Une première approche pour choisir un modèle est de se placer dans un contexte uniquement mélange, c'est-à-dire sans faire aucune référence ni à la classification automatique, ni à la discrimination. La philosophie de cette approche mélange est de retenir le modèle qui fournit une densité mélange f qui représente au mieux les données $\mathcal{X}$. Les moyens utilisés sont variés : tests d'hypothèses classiques, tests d'hypothèses bayésiens, critères numériques de type AIC pénalisant la complexité du modèle, détection d'un coude du maximum de vraisemblance. Nous passons ainsi en revue bon nombre de méthodes. Nous proposons de plus une méthode originale pour trouver le nombre de classes : nous introduisons une hypothèse de précision sur les données pour détecter un coude dans la courbe du maximum de vraisemblance. L'utilisateur définit la précision des données qui sont étudiées et ensuite cette précision sert à trouver un coude dans la vraisemblance. La valeur de la précision est, à notre avis, bien plus facile à établir qu'une définition directe du coude de la vraisemblance.

3.3.1 Tests d'hypothèses

Principe des tests

La théorie dite «classique» pour choisir des hypothèses peut s'appliquer au choix de nos modèles. Le principe général est de tester une hypothèse H_0 (l'hypothèse *nulle*) contre une hypothèse H_1 (l'hypothèse *alternative*) avec

$$\begin{array}{l} \mathrm{H}_0 : \boldsymbol{\theta} \in \Theta_0 \\ \mathrm{H}_1 : \boldsymbol{\theta} \in \Theta_1 \end{array}.$$

H_0 et H_1 correspondent respectivement aux modèles $\mathcal{M}_0$ et $\mathcal{M}_1$ vrais. En posant $\Theta = \Theta_0 \cup \Theta_1$, le rapport des vraisemblances maximales s'écrit :

$$r(\mathcal{X}) = \frac{\max_{\theta \in \Theta_0} f(\mathcal{X}|\boldsymbol{\theta})}{\max_{\theta \in \Theta} f(\mathcal{X}|\boldsymbol{\theta})}.$$

Intuitivement, des petites valeurs pour $r(\mathcal{X})$ suggèrent de rejeter H_0 et, au contraire, des valeurs de $r(\mathcal{X})$ «proches» de 1, d'accepter H_0. Mais la distribution du rapport $r(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n)$ est souvent impossible à obtenir, sauf dans le *cas asymptotique* en supposant certaines *conditions de régularité* (par exemple $\boldsymbol{\theta}$ ne doit pas être sur la frontière de l'espace Θ). Il est alors prouvé (voir Bernardo et Smith p.487, 1994 [10] ou Saporta p.327, 1990 [80] pour un bref rappel) que $-2\ln r(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n)$ a une distribution limite sous H_0 qui est un $\chi^2_{q-q_0}$, q (respectivement q_0) indiquant le nombre de paramètres libres dans Θ (resp. Θ_0).

Le test du rapport des vraisemblances maximales, ou test LRT (*Likelihood Ratio Test*), comme tous les tests de la théorie classique, fixe l'erreur de première espèce α (probabilité de choisir H_1 alors que H_0 vraie) appelée aussi degré du test. La règle d'acceptation du test est alors définie par

$$\begin{cases} \text{accepter } \mathrm{H}_0 \text{ si} & -2\ln r(\mathcal{X}) < \chi^2_{q-q_0,1-\alpha} \\ \text{accepter } \mathrm{H}_1 \text{ si} & -2\ln r(\mathcal{X}) > \chi^2_{q-q_0,1-\alpha}, \end{cases}$$

où $\chi^2_{k,1-\alpha}$ est le fractile d'ordre $1-\alpha$ de la loi de χ^2 à k degrés de liberté.

Un défaut bien connu de ce test est qu'il accepte facilement H_0 pour des petites tailles d'échantillon. Au contraire, quand la taille de l'échantillon devient grande, il favorise excessivement l'hypothèse H_1. Dans ce dernier cas, les utilisateurs préfèrent généralement utiliser le critère bayésien asymptotique BIC que nous décrivons plus loin.

Inconvénient des tests successifs

Le LRT permet de tester entre une hypothèse nulle et alternative, c'est-à-dire de faire un choix entre seulement deux modèles $\mathcal{M}_0$ et $\mathcal{M}_1$. Pour choisir entre trois modèles $\mathcal{M}_0$, $\mathcal{M}_1$ et $\mathcal{M}_2$, il peut être nécessaire de faire deux tests successifs. La procédure des tests successifs s'énonce :

1. classer les modèles par ordre décroissant de préférence. On part généralement du modèle le plus simple pour aller vers le plus complexe (en nombre de paramètres) car on suppose préférer la parcimonie. On obtient l'ordre $\mathcal{M}_0$, $\mathcal{M}_1$ et $\mathcal{M}_2$, $\mathcal{M}_0$ étant le plus simple des trois ;

2. tester ($\mathrm{H}_0 : \mathcal{M}_0$) contre ($\mathrm{H}_1 : \mathcal{M}_1$) à un degré α. Si H_0 est conservée alors la procédure s'arrête et le modèle $\mathcal{M}_0$ est retenu. Sinon passer à l'étape suivante ;

3. tester ($\mathrm{H}_0 : \mathcal{M}_1$) contre ($\mathrm{H}_1 : \mathcal{M}_2$) à un degré α. Si H_0 est conservée alors la procédure s'arrête et le modèle $\mathcal{M}_1$ est retenu. Sinon la procédure s'arrête et le modèle $\mathcal{M}_2$ est retenu.

L'inconvénient de cette procédure, proposée par ailleurs par Orlov pour le nombre de classes, est de perdre le degré global du test : quel est le degré du test de ($\mathrm{H}_0 : \mathcal{M}_0$) contre ($\mathrm{H}_1 : \mathcal{M}_2$) ? C'est pour cette raison que nous préférons dans la suite utiliser des critères numériques qui permettent de choisir entre de nombreux modèles.

Problèmes pour le nombre de classes

Plaçons nous ici dans le cas simple où l'échantillon est réel ($x_i \in \mathbb{R}, 1 \leq i \leq n$) et testons les deux hypothèses

$$\begin{cases} \mathrm{H}_0 : 1 \text{ classe} \\ \mathrm{H}_1 : 2 \text{ classes.} \end{cases}$$

La loi mélange est donc définie par la densité

$$f(x|\boldsymbol{\theta}) = p\phi_1(x|\mu_1, \sigma_1^2) + (1-p)\phi_2(x|\mu_2, \sigma_2^2).$$

Il se trouve que l'espace des paramètres Θ_0 qui définit H_0 peut s'écrire de deux manières différentes et que chacune conduit à un nombre de degrés de liberté différent dans la statistique du χ^2. On a en effet :

$$\text{ou} \begin{cases} \mathrm{H}_0 : p = 1 & \text{donc 1 seul degré de liberté} \\ \mathrm{H}_0 : \mu_1 = \mu_2 \text{ et } \sigma_1^2 = \sigma_2^2 & \text{donc 2 degrés de liberté.} \end{cases}$$

De plus Aitkin et Rubin [1] ont montré d'autres difficultés interdisant l'usage du LRT :

- si H_0 est définie par $p = 1$, alors l'estimateur du maximum de vraisemblance $\hat{p}$ ne peut, sous H_0, suivre asymptotiquement une loi normale centrée sur 1 puisque $p \leq 1$. Le paramètre p étant donc sur la frontière de l'espace des paramètres, la démonstration conduisant à la distribution du χ^2 ne tient pas ;
- si H_0 est définie par $\mu_1 = \mu_2$ et $\sigma_1^2 = \sigma_2^2$, alors, sous H_0, la vraisemblance est constante pour toute valeur de p et donc $\hat{p}$ ne peut suivre asymptotiquement une loi normale de plein rang.

On peut donc s'attendre à ce que $-2\ln r(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n)$ ne suive pas asymptotiquement une distribution du χ^2 et donc qu'un test basé sur cette hypothèse soit décevant comme en témoignent les simulations de Everitt [46].

Diverses procédures ont été envisagées pour contourner les difficultés rencontrées par le LRT pour le nombre de classes. Nous les détaillons en nous plaçant dans la situation d'un test de K classes (H_0) contre K' classes (H_1) avec $K < K'$. Il s'agit de la conjecture de Wolfe, de l'approche d'Aitkin et Rubin et enfin du Bootstrap paramétrique de McLachlan.

Conjecture de Wolfe

À partir d'études de simulations avec mélanges normaux, Wolfe [91] [92] *conjectura* une nouvelle distribution asymptotique pour le LRT. Elle s'énonce

$$-2\ln r(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n)\frac{(n-1-d-\frac{1}{2}K')}{n} \sim \chi^2_{2d(K'-K)}. \tag{3.1}$$

Pour les modèles de mélange gaussiens correspondant à des matrices de variance et des proportions identiques pour chaque classe, il est facile de montrer que la différence du nombre de degrés de liberté entre un modèle à K' classes et un modèle à K classes ($K' > K$) est égale à $d(K' - K)$ (voir par exemple les degrés de liberté des modèles $[p\lambda I]$, $[p\lambda B]$ ou $[p\lambda C]$ du tableau 1.3 au chapitre 1). Wolfe doubla donc artificiellement le nombre de degrés de liberté, dans le cas où les matrices de variance et les proportions sont égales.

Des résultats de simulations encourageants sont obtenus par Everitt [46], avec des matrices de variance égales et un nombre de données suffisamment grand par rapport à l'espace d'observation ($n \gg d$). Par contre, cette approximation fut mise en défaut par MacLachlan [68] pour des matrices de variance différentes. Cet auteur montre que le test a une puissance assez faible.

Approche d'Aitkin et Rubin

Pour éviter de tester sur la frontière de l'espace des paramètres, Aitkin et Rubin [1] utilisent une distribution a priori $\pi(\boldsymbol{p})$ ($\boldsymbol{p} = (p_1, \ldots, p_K)$) sur les proportions. Ils maximisent alors par l'algorithme EM une nouvelle vraisemblance calculée non plus sur $\boldsymbol{\theta} = (\boldsymbol{p}, \boldsymbol{a})$ mais seulement sur $\boldsymbol{a}$, avec $\boldsymbol{a} = (\boldsymbol{a}_k)_{1 \leq k \leq K}$. Elle s'écrit

$$f(\mathcal{X}|\boldsymbol{a}) = \int f(\mathcal{X}|\boldsymbol{\theta})\pi(\boldsymbol{p})d\boldsymbol{p}.$$

L'étape E de l'algorithme EM (itération m) revient à calculer les probabilités a posteriori d'appartenance par la formule

$$t_{ik}^m = \int \left[\frac{p_k \phi_k(\boldsymbol{x}_i|\boldsymbol{a}_k^{m-1})}{f(\mathcal{X}|\boldsymbol{a}^{m-1})}\right] \pi(\boldsymbol{p}|\boldsymbol{x}_i, \boldsymbol{a}^{m-1})d\boldsymbol{p}$$

avec $\pi(\boldsymbol{p}|\mathcal{X}, \boldsymbol{a}^{m-1})$ loi a posteriori de $\boldsymbol{p}$ sachant $\mathcal{X}$ et $\boldsymbol{a}^{m-1}$. Elle est donnée par $\pi(\boldsymbol{p}|\mathcal{X}, \boldsymbol{a}^{m-1}) \propto f(\mathcal{X}|\boldsymbol{p}, \boldsymbol{a}^{m-1})\pi(\boldsymbol{p})$. On remarque que l'étape E nécessite dans ce cas $K - 1$ intégrations.

C'est la nouvelle vraisemblance $f(\mathcal{X}|\boldsymbol{a})$ qui est alors utilisée dans le LRT en utilisant de nouveau l'approximation asymptotique du χ^2. Dans le cas d'un test entre une et deux classes, les hypothèses s'énoncent :

$$\begin{cases} \mathrm{H}_0 : \boldsymbol{a}_1 = \boldsymbol{a}_2 \\ \mathrm{H}_1 : \boldsymbol{a}_1 \neq \boldsymbol{a}_2. \end{cases}$$

L'idée semble intéressante car on ne teste plus sur les frontières de l'espace des paramètres. Pourtant, cette distribution asymptotique n'est pas toujours un χ^2 car la matrice d'information de Fisher a asymptotiquement des valeurs propres négatives avec une probabilité non nulle sous H_0 (Quinn et al. [73]).

Bootstrap paramétrique

Enfin McLachlan [68] propose un test du rapport de vraisemblance s'appuyant sur un bootstrap paramétrique. Ce test est assez fiable mais plutôt coûteux en temps de calcul. On estime tout d'abord les paramètres $\boldsymbol{\theta}_0$ et $\boldsymbol{\theta}_1$ par maximum de vraisemblance sous respectivement H_0 et H_1. On tire ensuite R échantillons $\mathcal{X}_1, \ldots, \mathcal{X}_R$ de taille n sous H_0, c'est-à-dire selon la loi de densité $f(\boldsymbol{x}|\hat{\boldsymbol{\theta}}_0)$. Pour chacun de ces échantillons, les paramètres du mélange sont estimés sous H_0 et sous H_1 et la statistique du rapport des vraisemblances est calculée. Disposant ainsi de R valeurs $-2\ln r(\mathcal{X}_1), \ldots, -2\ln r(\mathcal{X}_R)$, on a une estimation de la vraie distribution de $-2\ln r(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n)$ sous H_0.

Le test au seuil α se fait en rejetant H_0 si la valeur de la statistique $-2\ln r(\mathcal{X})$ calculée pour l'échantillon original est plus grande que la j^{e} plus petite valeur de l'ensemble $-2\ln r(\mathcal{X}_1), \ldots, -2\ln r(\mathcal{X}_R)$. On calcule j par

$$j = (R+1)(1-\alpha).$$

Pour une estimation acceptable, il faut généralement $R \approx 100$.

Comparaison des procédures

Soromenho [84] compare les procédures d'Orlov, de Wolfe, d'Aitkin et Rubin et de McLachlan. En se plaçant dans les cas univariés et bivariés, elle constate que la procédure de McLachlan est la plus performante.

Prenant un échantillon test (en l'occurence celui qui se trouve à la figure 3.2 quelques pages plus loin), nous appliquons le test classique, le test de Wolfe et le test de McLachlan ($R = 100$) entre l'hypothèse H_0 de deux classes et l'hypothèse H_1 de trois classes, en utilisant le modèle $[p\lambda_k C_k]$. La distribution de $-2\ln(r)$ sous H_0 pour chacun des trois cas est représentée sur la figure 3.1. On remarque alors une grande différence entre les trois distributions. Néanmoins, les trois tests conduisent tous à l'acceptation de trois classes pour des seuils α classiques ($\alpha = 0.01$ par exemple). En effet, on a $-2(\mathrm{L}(2) - \mathrm{L}(3)) = 54.6$ alors que la limite de rejet de H_0 pour les trois tests et avec un seuil égal à 0.01 est inférieure à 20.

3.3.2 Approche bayésienne

L'approche bayésienne consiste à retenir *le modèle le plus probable a posteriori* si le coût 0–1 est utilisé. Le calcul de cette probabilité, ou de manière équivalente le calcul de *la vraisemblance intégrée*, nécessite d'intégrer la vraisemblance sur l'espace Θ du paramètre du mélange. Lorsque l'estimation de cette intégrale est délicate (c'est le cas si la dimension de Θ est grande), l'estimation non plus d'une vraisemblance intégrée mais du rapport entre deux vraisemblances intégrées est plus robuste en général : c'est *le facteur de Bayes*. Le facteur de Bayes par sa

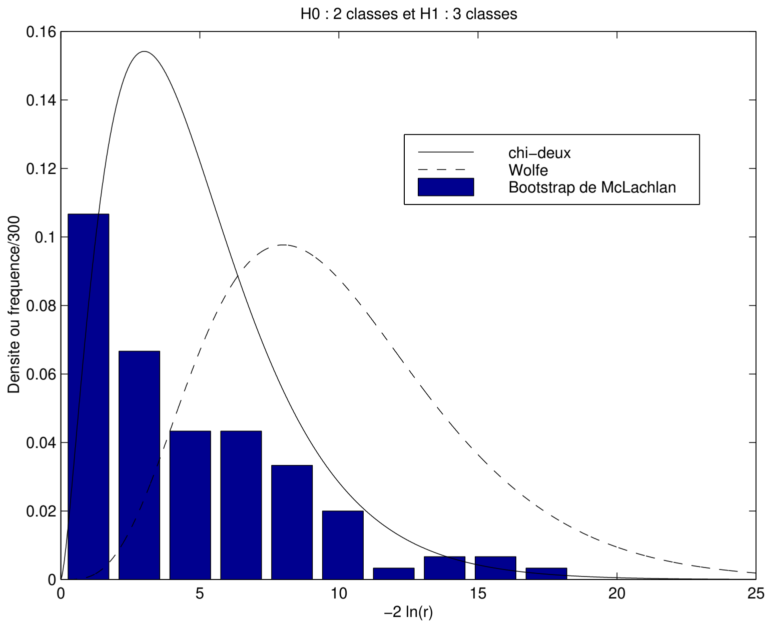


FIG. 3.1 – *Trois approximations différentes pour la distribution du rapport des vraisemblances maximales.*

forme ressemble au rapport des vraisemblances maximales et Jeffreys [59] [60] a fait une certaine analogie avec le LRT dans son travail sur l'inférence scientifique.

Nous appuyant, entre autres, sur l'article de Kass et Raftery [63], nous présentons l'usage des facteurs de Bayes et quelques méthodes permettant de les calculer (critère BIC et critère AWE). Nous présentons aussi la méthode de Laplace-Metropolis, méthode récente qui permet d'estimer directement la vraisemblance intégrée et donc de ne pas utiliser le facteur de Bayes.

Modèle le plus probable a posteriori

Si, pour chaque modèle $\mathcal{M}$ de l'ensemble Δ, nous connaissons une densité de probabilité $\mathrm{P}(\mathcal{X}|\mathcal{M})$ et une probabilité a priori $\mathrm{P}(\mathcal{M})$, le théorème de Bayes permet facilement de calculer la probabilité a posteriori de ce modèle $\mathcal{M}$:

$$\mathrm{P}(\mathcal{M}|\mathcal{X}) = \frac{\mathrm{P}(\mathcal{X}|\mathcal{M})\mathrm{P}(\mathcal{M})}{\sum_{\mathcal{M}'\in\Delta}\mathrm{P}(\mathcal{X}|\mathcal{M}')\mathrm{P}(\mathcal{M}')}. \tag{3.2}$$

Le terme $\mathrm{P}(\mathcal{X}|\mathcal{M})$ se calcule par l'intégration suivante :

$$\mathrm{P}(\mathcal{X}|\mathcal{M}) = \int_\Theta f(\mathcal{X}|\boldsymbol{\theta},\mathcal{M})\pi(\boldsymbol{\theta}|\mathcal{M})d\boldsymbol{\theta}, \tag{3.3}$$

où $f(\mathcal{X}|\boldsymbol{\theta},\mathcal{M}_k)$ est la densité mélange sous le modèle $\mathcal{M}$ et $\pi(\boldsymbol{\theta}|\mathcal{M})$ est la loi a priori de $\boldsymbol{\theta}$ sous ce même modèle. Le terme $\mathrm{P}(\mathcal{X}|\mathcal{M})$ est généralement appelé *probabilité marginale* de $\mathcal{X}$, *vraisemblance marginale* ou encore *vraisemblance intégrée*. Le choix d'un modèle peut ensuite se faire dans le cadre de la théorie de la décision en choisissant le coût 0–1 :

$$\mathcal{L}(\mathcal{M}'|\mathcal{M}) = \begin{cases} 0 & \text{si } \mathcal{M} = \mathcal{M}' \\ 1 & \text{sinon.} \end{cases}$$

Dans ce cas, on choisit le modèle $\hat{\mathcal{M}}$ qui minimise le risque a posteriori

$$\mathcal{R}(\hat{\mathcal{M}}|\mathcal{X}) = \sum_{\mathcal{M}\in\Delta} \mathcal{L}(\hat{\mathcal{M}}|\mathcal{M})\mathrm{P}(\mathcal{M}|\mathcal{X}).$$

Cela revient à choisir le modèle le plus probable a posteriori, donc celui qui maximise $\mathrm{P}(\mathcal{M}|\mathcal{X})$.

Notant $\mathcal{M}_0$ un des modèles en compétition de Δ, on peut exprimer $\mathrm{P}(\mathcal{M}|\mathcal{X})$ de l'équation 3.2 par

$$\begin{aligned} \mathrm{P}(\mathcal{M}|\mathcal{X}) &= \frac{B_0\mathrm{P}(\mathcal{M})}{\sum_{\mathcal{M}'\in\Delta} B_0'\mathrm{P}(\mathcal{M}')} \\ &\propto B_0\mathrm{P}(\mathcal{M}), \end{aligned}$$

avec

$$B_0 = \frac{\mathrm{P}(\mathcal{X}|\mathcal{M})}{\mathrm{P}(\mathcal{X}|\mathcal{M}_0)}.$$

Le terme B_0 est appelé *facteur de Bayes*, rapport entre les vraisemblances intégrées des deux modèles $\mathcal{M}$ et $\mathcal{M}_0$. De plus, il est courant de n'avoir aucun a priori sur les modèles et l'on pose alors toutes les probabilités a priori égales :

$$\forall\mathcal{M} \in \Delta, \qquad \mathrm{P}(\mathcal{M}) = \frac{1}{\mathrm{card}(\Delta)}.$$

Ceci entraîne que le calcul de la probabilité a posteriori d'un modèle se ramène soit au calcul du facteur de Bayes B_0, soit de manière équivalente au calcul de la vraisemblance intégrée $\mathrm{P}(\mathcal{X}|\mathcal{M})$:

$$\begin{aligned} \mathrm{P}(\mathcal{M}|\mathcal{X}) &\propto B_0 \\ &\propto \mathrm{P}(\mathcal{X}|\mathcal{M}). \end{aligned}$$

À ce stade, il peut bien sûr paraître artificiel de faire une différence entre le calcul du facteur de Bayes B_0 et le calcul de la vraisemblance intégrée $\mathrm{P}(\mathcal{X}|\mathcal{M})$. Pourtant, il existe une raison importante qui est la qualité de l'estimation. Il est en effet plus robuste d'estimer le rapport des vraisemblances intégrées du facteur de Bayes (sans estimer aucune vraisemblance intégrée) que d'estimer la vraisemblance intégrée. Les estimations du facteur de Bayes conduisent alors aux critères BIC et AWE que nous allons décrire. Néanmoins, une méthode récente, dite de Metropolis-Hastings, permet d'estimer correctement la vraisemblance intégrée, donc de s'affranchir de la voie détournée du facteur de Bayes. Nous présentons également cette méthode. Mais tout d'abord nous présentons une analogie entre les facteurs de Bayes et les tests d'hypothèses.

Relation entre facteur de Bayes et tests d'hypothèses

Dans la théorie classique (ou fréquentiste), on a deux hypothèses : l'hypothèse nulle et l'hypothèse alternative. Nous posons que l'hypothèse nulle (c'est-à-dire celle à laquelle on croit le plus) correspond à supposer le modèle $\mathcal{M}_0$ vrai, tandis que l'hypothèse alternative (le «challenger») correspond à supposer le modèle $\mathcal{M}$ vrai. On établit donc une *dissymétrie* entre les modèles.

Plus B_0 (donc son logarithme $2\ln B_0$) sera grand, plus l'évidence de $\mathcal{M}$ contre $\mathcal{M}_0$ sera forte. Le terme $2\ln B_0$ est souvent pour cela appelé le *poids de l'évidence* (*weight of evidence*). Contrairement au test LRT fréquentiste, il n'existe généralement pas de distribution asymptotique connue et cette notion d'évidence n'est ici que qualitative. Kass et Raftery [63], adaptant le travail de Jeffreys [60] sur le choix entre plusieurs théories scientifiques, fournissent alors une échelle toute subjective (et fortement dépendante du contexte) pour quantifier l'évidence de $\mathcal{M}$ contre $\mathcal{M}_0$ (tableau 3.3).

$2\ln B_0$	B_0	évidence contre $\mathcal{M}_0$
$0 \to 2$	$1 \to 3$	peu convaincante
$2 \to 5$	$3 \to 12$	certaine
$5 \to 10$	$12 \to 150$	forte
> 10	> 150	décisive

TAB. 3.3 – *Échelle d'évidence subjective de $\mathcal{M}$ contre $\mathcal{M}_0$.*

Approximations du facteur de Bayes : critères BIC et AWE

Nous présentons les critères BIC et AWE, tous deux approximations du facteur de Bayes B_0. Ces critères nous permettent d'éviter de définir les lois a priori.

Il est en effet possible d'éviter de fixer les distributions a priori $\pi(\boldsymbol{\theta}|\mathcal{M})$ ($\mathcal{M} \in \Delta$), pour calculer le facteur de Bayes B_0. On utilise le critère de Schwarz [81]

$$S = 2\mathrm{L}(\mathcal{M}) - 2\mathrm{L}(\mathcal{M}_0) - (q(\mathcal{M}) - q(\mathcal{M}_0))\ln(n), \tag{3.4}$$

où $\mathrm{L}(\mathcal{M})$ est le maximum de log-vraisemblance avec le modèle $\mathcal{M}$ et $q(\mathcal{M})$ le nombre de paramètres libres dans ce modèle. Le terme S est une *grossière approximation* de $2\ln B_0$ car, sous certaines conditions, on a asymptotiquement

$$\frac{S - 2\ln B_0}{2\ln B_0} \to 0.$$

On appelle généralement BIC, pour *Bayes Information Criterion*, la quantité $-S$. Cette approximation peut suffire à donner un ordre de grandeur de $2\ln B_0$ et donc permet d'utiliser la table d'échelle de grandeur 3.3. Si plus de deux modèles sont en compétition, on calcule le facteur de Bayes entre chacun de ces modèles et le modèle de référence $\mathcal{M}_0$ puis on retient le modèle qui produit le plus grand facteur de Bayes. Cette stratégie est donc indépendante du modèle de référence $\mathcal{M}_0$ (voir l'équation 3.4) et revient simplement à choisir le modèle qui minimise le critère

$$\mathrm{BIC}(\mathcal{M}) = -2\mathrm{L}(\mathcal{M}) + q(\mathcal{M})\ln(n). \tag{3.5}$$

Banfield et Raftery [7] ont donné une autre approximation du facteur de Bayes B_0, approximation nous affranchissant de nouveau de la définition de la loi a priori. Contrairement au critère BIC, ils se limitent au choix du nombre de classes (modèle gaussien connu). La vraisemblance intégrée $\mathrm{P}(\mathcal{X}|K)$ correspondant au modèle à K classes se décompose par rapport à la partition sous la forme

$$\mathrm{P}(\mathcal{X}|K) = \sum_{(\boldsymbol{z})} \int_{\Theta_K} f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta}, K)\pi(\boldsymbol{\theta}|K)d\boldsymbol{\theta},$$

et fait apparaître une structure hiérarchique, avec $f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta}, K)$ la vraisemblance classifiante de $(\boldsymbol{\theta}, \boldsymbol{z})$ pour K classes (équation 1.8 du chapitre 1) et $\pi(\boldsymbol{\theta}|K)$ loi a priori de $\boldsymbol{\theta}$ pour ce nombre de classes. Se plaçant dans un cadre de classification hiérarchique (donc en utilisant le fait que K classes résultent de l'agglomération de deux classes parmi $K+1$), imposant que les proportions des classes sont égales ($p_1 = \cdots = p_K = 1/K$) et utilisant des approximations asymptotiques (dont celle de Wolfe [91] [92]), ils obtiennent finalement l'approximation asymptotique suivante :

$$-2\ln B_0 \approx -2\mathrm{CL}(K) + 2\mathrm{CL}(1) + 2(q(K) - q(1))(\frac{3}{2} + \ln(n)),$$

où B_0 est le facteur de Bayes de K classes contre un modèle $\mathcal{M}_0$ à *une seule classe* et $\mathrm{CL}(K)$ le maximum de log-vraisemblance classifiante pour K classes. De même que pour le critère BIC, on abandonne la constante due au modèle $\mathcal{M}_0$ (le modèle monoclasse comme on l'a dit) pour obtenir le critère AWE (*Approximate Weight of Evidence*)

$$\mathrm{AWE}(K) = -2\mathrm{CL}(K) + 2q(K)(\frac{3}{2} + \ln(n)). \tag{3.6}$$

On retient le nombre de classes qui minimise ce terme.

Approximation de la vraisemblance intégrée

Le calcul direct de la vraisemblance intégrée (ou du facteur de Bayes après calcul, cette fois, des vraisemblances intégrées) nécessite généralement de définir les lois a priori. Le premier objectif est bien sûr de choisir des distributions représentant au mieux l'information disponible. Evidemment, prendre les critères BIC ou AWE ne nécessite aucune information a priori mais ces critères ne sont valables que pour des échantillons «suffisamment grands» et tout le problème est de connaître ce seuil d'acceptabilité. Différentes méthodes de choix de lois a priori ont été déjà précisées dans la partie consacrée à l'estimation au chapitre 2 (section 2.2.3). Ces lois sont souvent choisies par convenance (facilité de calcul par les lois a priori conjuguées) car l'effet de ces dernières est faible si l'échantillon est assez grand. Mais contrairement à l'estimation bayésienne ponctuelle, le calcul du facteur de Bayes ou de la vraisemblance intégrée (outre BIC et AWE) sont plus sensibles au choix des lois a priori sur les paramètres (Kass et Greenhouse, 1989 [62], Kass, 1993 [61]). L'usage des lois a priori conjuguées ainsi que de lois non informatives doit se faire avec prudence. En général, les lois impropres standards utilisées en estimation sont ici inapplicables. Ainsi, le calcul du facteur de Bayes ou de la vraisemblance intégrée doit généralement s'accompagner, par prudence, d'une *étude de sensibilité* de leur calcul au choix de la loi a priori. Par exemple, on peut perturber les hyperparamètres puis recalculer les termes d'intérêt.

Une fois résolu le problème de la loi a priori, l'estimation directe de la vraisemblance intégée (3.3) est peu évidente en général, à cause des intégrations souvent multiples : le calcul analytique est presque toujours impossible et l'usage de méthodes classiques d'approximations numériques donne des résultats assez instables (Kass et Raftery [63]). Une idée est alors d'utiliser un échantillon $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m$ approximativement distribué suivant la loi a posteriori $\pi(\boldsymbol{\theta}|\mathcal{X}, \mathcal{M})$. Dans notre contexte mélange, nous avons vu qu'un tel échantillon pouvait s'obtenir par une méthode MCMC comme l'échantillonneur de Gibbs. Disposant d'un échantillon obtenu par cette méthode, une première voie est alors d'estimer la vraisemblance intégrée à l'aide de l'approximation de Monte-Carlo suivante (Newton et Raftery [71]) :

$$\frac{1}{\hat{\mathrm{P}}(\mathcal{X}|\mathcal{M})} = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{f(\mathcal{X}|\boldsymbol{\theta}_i \mathcal{M})}.$$

On a $\hat{\mathrm{P}}(\mathcal{X}|\mathcal{M}) \to \mathrm{P}(\mathcal{X}|\mathcal{M})$ presque sûrement quand $n \to \infty$. Cet estimateur, bien que facile à calculer, est assez instable. Néanmoins il peut être assez précis pour la table d'échelle de référence de $2 \ln B_0$ (tableau 3.3). Pour augmenter la précision de l'estimateur, Lewis et Raftery [65] et Raftery [74] suggérent ce qu'ils appellent l'estimateur de *Laplace-Metropolis* de la vraisemblance intégrée, estimateur qui utilise l'approximation de Laplace. Sous certaines conditions de régularité, portant par exemple sur la fonction à intégrer, on peut faire une approximation quadratique (décomposition en série de Taylor à l'ordre deux) de $\ln(f(\mathcal{X}|\boldsymbol{\theta}, \mathcal{M})\pi(\boldsymbol{\theta}|\mathcal{M}))$ en son mode $\hat{\boldsymbol{\theta}}$. On obtient alors l'approximation de Laplace

$$\hat{\mathrm{P}}(\mathcal{X}|\mathcal{M}) = (2\pi)^{q/2} |\hat{\boldsymbol{\Psi}}|^{1/2} f(\mathcal{X}|\hat{\boldsymbol{\theta}}, \mathcal{M}) \pi(\hat{\boldsymbol{\theta}}|\mathcal{M}), \qquad (3.7)$$

avec q nombre de paramètres libres dans $\mathcal{M}$, $\hat{\boldsymbol{\Psi}}$ opposée de l'inverse de la matrice hessienne de $\ln(f(\mathcal{X}|\boldsymbol{\theta}, \mathcal{M})\pi(\boldsymbol{\theta}|\mathcal{M}))$ en $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$. La méthode Laplace-Metropolis consiste ensuite à simplement utiliser la suite $(\boldsymbol{\theta}_m)$ pour estimer le mode $\hat{\boldsymbol{\theta}}$ de $\ln(f(\mathcal{X}|\boldsymbol{\theta}, \mathcal{M})\pi(\boldsymbol{\theta}|\mathcal{M}))$, ainsi que pour estimer $|\hat{\boldsymbol{\Psi}}|$.

À la suite de l'estimation bayésienne des paramètres d'un mélange de lois normales, Bensmail et al. [9] effectuent un choix de modèle en contexte bayésien sur des données réelles et sur des données simulées : choix du nombre de classes et choix de la paramétrisation de la matrice de variance parmi 8 possibles (voir la section 2.2.3). Le calcul des vraisemblances intégrées se fait par cette méthode de Laplace-Metropolis, donc utilise directement les sorties de l'algorithme MCMC employé pour l'estimation. Les lois a priori choisies sont donc celles de l'estimation, c'est-à-dire des lois conjuguées. Une étude de robustesse accompagne l'étude en faisant varier les hyperparamètres. Ces auteurs présentent cette démarche comme une extension du critère AWE. D'une part, ils ne se limitent plus ni au seul nombre de classes, ni à une approche classification hiérarchique, ni non plus à des proportions de mélange égales. D'autre part, leur approche est beaucoup plus précise que l'approximation grossière de AWE.

Lien avec l'approche MDL

Rissanen p.59 [78] propose un critère de choix de modèle appelé *complexité stochastique* (*stochastic complexity*). Elle est définie, dans le cas continu, par

$$SC(\mathcal{X}|\mathcal{M}) = -\ln \int_{\Theta} f(\mathcal{X}|\boldsymbol{\theta}, \mathcal{M})\pi(\boldsymbol{\theta}|\mathcal{M})d\boldsymbol{\theta},$$

et le modèle choisi est celui qui minimise la valeur de SC. On peut remarquer que $SC(\mathcal{X}|\mathcal{M}) = -\ln \mathrm{P}(\mathcal{X}|\mathcal{M})$ et donc que SC s'apparente alors à la vraisemblance intégrée bayésienne. Dans le cas asymptotique, Rissanen montre que $SC(\mathcal{X}|\mathcal{M}) \approx \mathrm{MDL}(\mathcal{M})$ avec $\mathrm{MDL}(\mathcal{M}) = -\mathrm{L}(\mathcal{M}) + q(\mathcal{M})\ln(n)$ (relation 2.16). On retrouve de nouveau un lien entre les bayésiens et les partisans de la théorie du codage car

$$2\mathrm{MDL}(\mathcal{M}) = \mathrm{BIC}(\mathcal{M}).$$

Dans une situation non asymptotique, Bryant [29] propose plutôt d'utiliser MDL de l'équation 2.15 (version non asymptotique) à la place de SC, car minimiser MDL apporte en même temps un estimateur de $\boldsymbol{\theta}$. Il s'appuie aussi sur le fait qu'asymptotiquement, ils sont de toute façon équivalents. Donc l'estimation et le choix de modèles par la méthode MDL ne font qu'un. Le critère HMDL (section 2.2.4), extension hiérarchique de MDL, peut aussi être utilisé et l'extension aux données avec labels est directe.

3.3.3 Pénalisation de la vraisemblance par la complexité

Position du problème

La vraisemblance peut s'interpréter comme une mesure de la qualité de l'ajustement de la densité aux données. Ainsi, il peut être tentant d'utiliser le maximum de vraisemblance comme un critère, c'est-à-dire de choisir le modèle $\mathcal{M}$ qui maximise la vraisemblance parmi un ensemble de modèles. On aurait ainsi le modèle le plus «proche des données». Mais le critère du maximum de vraisemblance a le défaut de choisir systématiquement les modèles les plus complexes, car plus le modèle est complexe plus il peut «s'adapter» aux données. On constate alors un *apprentissage par cœur*, propriété bien génante pour décrire les données ou pour en prévoir de nouvelles. Quel intérêt de dire qu'il y a autant de classes que de données et que chaque classe est réduite à un seul point dans l'espace? Illustrons cela en notant Θ_1 et Θ_2 les espaces respectifs de $\mathcal{M}_1$ et $\mathcal{M}_2$, et $\mathrm{L}(\mathcal{M}_1)$ et $\mathrm{L}(\mathcal{M}_2)$ les maximums de vraisemblance associés. Ces maximums de vraisemblance sont aussi bien considérés dans le cas des données $\mathcal{X}$ sans labels (vraisemblance 2.2) que dans le cas des données $(\mathcal{X}, \boldsymbol{z})$ avec labels (vraisemblance 2.3). Si les deux espaces Θ_1 et Θ_2 sont emboîtés. il est immédiat que

$$\Theta_1 \subset \Theta_2 \quad \Rightarrow \quad \mathrm{L}(\mathcal{M}_1) \leq \mathrm{L}(\mathcal{M}_2),$$

donc que le modèle $\mathcal{M}_2$ sera toujours choisi. Cette situation arrive par exemple dans les cas suivants :

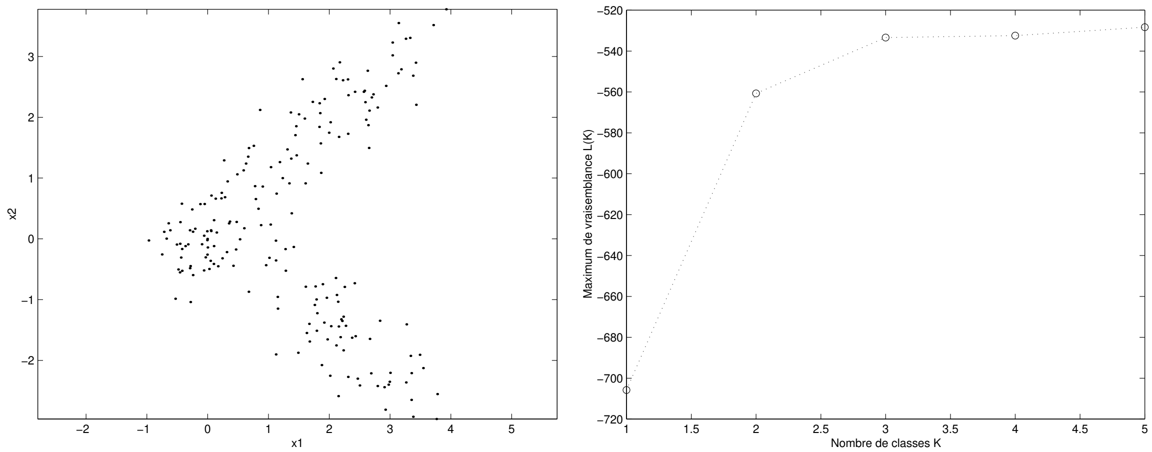
- **modèle gaussien** : choisir (à nombre de classes connu) entre $[p\lambda I]$ et $[p\lambda_k I]$;
- **nombre de classes** : choisir entre K et $K+1$ classes avec un modèle gaussien $[p_k\lambda I]$, ou de manière plus générale un modèle gaussien à *proportions libres*.

Par contre on ne peut théoriquement rien prédire si $\Theta_1 \not\subset \Theta_2$. Par exemple :

- **modèle gaussien** : choisir (à nombre de classes connu) entre $[p\lambda B_k]$ et $[p\lambda_k B]$;
- **nombre de classes** : choisir entre K et $K+1$ classes avec un modèle gaussien $[p\lambda I]$, ou de manière plus générale un modèle gaussien à *proportions contraintes*. Néanmoins, les expériences intensives de Cutler et Windham [41] montrent que le maximum de vraisemblance croît encore avec le nombre de classes. On constate d'ailleurs cette insuffisance du critère de vraisemblance par un exemple sur la figure 3.2. Un échantillon en dimension deux semble composé de deux ou trois classes mais le critère du maximum de vraisemblance en détecte cinq avec le modèle $[p\lambda_k C_k]$.

Pénalisation du maximum de vraisemblance

Puisque le critère de vraisemblance a tendance à sélectionner les modèles les plus complexes, des critères pénalisant cette vraisemblance par la complexité du modèle ont été proposés. Ils cherchent à réaliser un *compromis* entre la complexité



«visuellement» 2 ou 3 classes ... ... mais choix de 5 classes

FIG. 3.2 – *Le maximum de vraisemblance croît avec le nombre de classes.*

du modèle et la proximité modèle–données mesurée par la vraisemblance. Ces critères s'énoncent

$$\begin{array}{rcccc} \text{critère}(\mathcal{M}) & = & -2\mathrm{L}(\mathcal{M}) & + & g(\mathcal{M}) \\ & = & \left(\begin{array}{c} \text{proximité} \\ \text{modèle–données} \end{array} \right) & + & \left(\begin{array}{c} \text{complexité} \\ \text{du modèle} \end{array} \right) \end{array}$$

où $g(\mathcal{M})$ est une application de l'ensemble des modèles Δ dans $\mathbb{R}$ et mesure la complexité du modèle $\mathcal{M}$. Une fois définie une telle application, il suffit de retenir *le modèle minimisant ce critère.*

Une grande famille d'applications g répond à la forme linéaire

$$g(\mathcal{M}) = \tau q(\mathcal{M})$$

où τ est un coefficient réel et $q(\mathcal{M})$ est le nombre de paramètres libres dans le modèle $\mathcal{M}$. Plusieurs critères utilisent cette définition mais se différencient par les valeurs de τ. Il s'agit des critères AIC, AIC3 et des critères bayésiens BIC et AWE. Le terme τ *règle* le compromis entre vraisemblance et complexité. Si $\tau < \tau'$, le critère associé à τ' choisira des modèles avec moins de paramètres que celui associé à τ. Précisons d'emblée qu'aucun consensus n'existe sur la valeur de τ. Certains auteurs le qualifient même de «nombre magique» comme pour montrer le caractère souvent approximatif de son obtention. De plus, la valeur optimale (au sens du choix du bon modèle) peut ne pas exister, la complexité n'étant peut-être pas une fonction linéaire du nombre de paramètres. Néanmoins, la simplicité de ces critères et leur large usage incitent à s'y attarder.

Critère AIC

AIC (*Akaike Information Criterion*) fut le précurseur des critères de pénalisation de ce genre (Akaike [2] [3]). Il fut introduit bien plus tard dans un contexte de mélange par Bozdogan et Sclove [27]. Il s'énonce

$$\mathrm{AIC}(\mathcal{M}) = -2\mathrm{L}(\mathcal{M}) + 2q(\mathcal{M}), \tag{3.8}$$

donc $\tau_{\text{AIC}} = 2$. Akaike a donné deux principales justifications :

- la première justification (Akaike [2]), s'appuie sur un argument de prédiction. Utilisant un coût entropique entre la vraie distribution et la distribution avec paramètres estimés par maximum de vraisemblance (distance de Kullback-Leibler entre les deux distributions), $\mathrm{AIC}(\mathcal{M})$ est un estimateur du risque moyen de choisir l'estimateur du maximum de vraisemblance sous modèle $\mathcal{M}$. Il peut aussi être vu comme un estimateur de l'espérance moyenne de la log-vraisemblance. Les détails du développement ont été repris par Bozdogan et nous les avons reportés à l'annexe B.

- la seconde (Akaike (1983) [4]) est bayésienne. Selon Akaike, les comparaisons de modèles par AIC sont asymptotiquement équivalentes à celles fondées sur les facteurs de Bayes. Or, cela serait vrai seulement si l'information de la loi a priori était aussi importante que celle fournit par les données. En général, comme on l'a déjà signalé sur la théorie bayésienne, l'information a priori est faible comparée aux données.

Linhart et Zucchini (1986) [66] ont généralisé l'approche d'Akaike en remplaçant la distance de Kullback-Leibler entre la vraie distribution et la distribution estimée par d'autres distances. Ils montrent alors que $\tau_{\text{AIC}} = 2$ est arbitraire et dépend essentiellement de la distance. La valeur $\tau'_{\text{AIC}} = 4$ serait alors aussi valable selon eux.

Critère AIC3

Lors de l'obtention du critère AIC, Bozdogan utilise la distribution asymptotique du rapport classique de vraisemblance sous l'hypothèse alternative. Pour les mêmes raisons que celles évoquées à la section 3.3.1, cette distribution asymptotique ne peut être utilisée dans le cas de la recherche du nombre de classes. Pour résoudre ce problème, Bozdogan [21] [22] propose de doubler le nombre de degrés de liberté dans la distribution du χ^2 décentrée comme le fait Wolfe [91]. Il obtient alors une variante de AIC, nommée AIC3 avec $\tau_{\text{AIC3}} = 3$:

$$\mathrm{AIC3}(\mathcal{M}) = -2\mathrm{L}(\mathcal{M}) + 3q(\mathcal{M}). \tag{3.9}$$

Le détail de l'obtention de AIC3 est également reporté à l'annexe B.

Critères bayésiens

Les critères bayésiens BIC ou MDL ($\tau_{\text{BIC}} = \ln(n)$) et AWE ($\tau_{\text{AWE}} = 2(\frac{3}{2} + \ln(n))$) sont aussi des critères pénalisant linéairement le nombre de paramètres.

Notons que τ est dans ce cas une fonction de la taille n de l'échantillon. Le critère AWE pénalise cette fois le maximum de vraisemblance classifiante et non plus le maximum de vraisemblance classique.

Critère ICOMP : un exemple de pénalisation non linéaire

Bozdogan [24] [25] [26] propose d'utiliser une mesure de complexité non linéaire faisant intervenir la matrice d'information de Fisher $\boldsymbol{F}_n(\mathcal{M})$ et le nombre de paramètres. Il s'agit de la complexité de van Emden $g(\mathcal{M}) = q(\mathcal{M}) \ln\left(\frac{\mathrm{tr} F_n^{-1}(\mathcal{M})}{q(\mathcal{M})}\right)$ $- \ln|\boldsymbol{F}_n^{-1}(\mathcal{M})|$. Il obtient alors le critère ICOMP (*Informational Complexity Criterion*) :

$$\mathrm{ICOMP}(\mathcal{M}) = -2\mathrm{L}(\mathcal{M}) + q(\mathcal{M}) \ln\left(\frac{\mathrm{tr}\boldsymbol{F}_n^{-1}(\mathcal{M})}{q(\mathcal{M})}\right) - \ln|\boldsymbol{F}_n^{-1}(\mathcal{M})|. \qquad (3.10)$$

Remarquons que cette pénalisation porte en outre sur les interdépendances entre paramètres. En effet, si $\boldsymbol{F}_n^{-1}(\mathcal{M}) = \beta(\mathcal{M})\boldsymbol{I}$, $\boldsymbol{I}$ matrice identité $q(\mathcal{M}) \times q(\mathcal{M})$ avec $\beta(\mathcal{M}) \in \mathbb{R}$, c'est-à-dire que les paramètres sont indépendants et de même importance $\beta(\mathcal{M})$, alors on obtient après simplification $\mathrm{ICOMP}(\mathcal{M}) = -2\mathrm{L}(\mathcal{M})$. La pénalisation de $\mathrm{L}(\mathcal{M})$ n'intervient donc plus dans ce cas.

Le calcul de ICOMP est beaucoup moins évident que celui des critères précédents. Notons $\boldsymbol{F}(\mathcal{M}) = \boldsymbol{F}(\hat{\boldsymbol{\theta}}_L)$ la matrice d'information de Fisher pour un seul individu $\boldsymbol{X}$ et calculée à la valeur $\hat{\boldsymbol{\theta}}_L$ du maximum de vraisemblance sous les contraintes du modèle $\mathcal{M}$. Par définition on a

$$\boldsymbol{F}(\mathcal{M}) = \mathrm{E}_X\left[\frac{\partial}{\partial\boldsymbol{\theta}} \ln f(\boldsymbol{X}|\boldsymbol{\theta})\bigg|_{\hat{\boldsymbol{\theta}}_L} \cdot \frac{\partial}{\partial\boldsymbol{\theta}} \ln f(\boldsymbol{X}|\boldsymbol{\theta})\bigg|_{\hat{\boldsymbol{\theta}}_L}'\right].$$

Les n vecteurs aléatoires $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ étant i.i.d., il est bien connu que la matrice de Fisher pour l'ensemble des n individus est donnée par $\boldsymbol{F}_n(\mathcal{M}) = n\boldsymbol{F}(\mathcal{M})$. Il est aisé de montrer (simple calcul) qu'utiliser l'une ou l'autre des matrices $\boldsymbol{F}_n(\mathcal{M})$ ou $\boldsymbol{F}(\mathcal{M})$ est absolument équivalent dans le terme de complexité $g(\mathcal{M})$. Nous utilisons alors simplement $\boldsymbol{F}(\mathcal{M})$, terme que nous estimons par une méthode de Monte-Carlo à la manière de Cutler et Windham [41]

$$\hat{\boldsymbol{F}}(\mathcal{M}) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\partial}{\partial\boldsymbol{\theta}} \ln f(\boldsymbol{x}_i|\boldsymbol{\theta})\bigg|_{\hat{\boldsymbol{\theta}}_L} \cdot \frac{\partial}{\partial\boldsymbol{\theta}} \ln f(\boldsymbol{x}_i|\boldsymbol{\theta})\bigg|_{\hat{\boldsymbol{\theta}}_L}'\right).$$

La matrice $\boldsymbol{F}$ dépend de la paramétrisation des matrices de variance $\boldsymbol{\Sigma}_k$ ainsi que des contraintes sur les proportions (égales ou libres). Nous reportons alors dans l'annexe C les compléments nécessaires au calcul de $\hat{\boldsymbol{F}}$ pour chacun des 28 modèles. Bozdogan [26] a établi de son côté des formes algébriques explicites de ICOMP pour certains des modèles les plus courants (linéaire sphérique, linéaire diagonal, linéaire général et cas quadratique). Il suppose que tous les $\boldsymbol{x}_i$ sont classifiés dans une classe et que cette appartenance est connue. Ainsi il évite la difficulté du calcul de $\boldsymbol{F}$ en supprimant le problème de mélange. Cette approximation, pour être valide, requiert implicitement des classes bien séparées. Voici l'expression qu'il obtient dans le cas linéaire général $[p\lambda C]$ avec K classes :

$$\begin{aligned}
\mathrm{ICOMP}_{[p\lambda C]}(K) &= -2\mathrm{L}_{[p\lambda C]}(K) \\
&+(Kd+d(d+1)/2)\ln\left[\frac{K\{1/\hat{p}_k\mathrm{tr}(\hat{\boldsymbol{\Sigma}})+1/2\mathrm{tr}(\hat{\boldsymbol{\Sigma}}^2)+1/2(\mathrm{tr}\hat{\boldsymbol{\Sigma}})^2+\sum_{j=1}^d(\sigma_{jj})^2\}}{Kd+d(d+1)/2}\right] \\
&-\left\{(d+2)K\ln|\hat{\boldsymbol{\Sigma}}|-d\sum_{k=1}^K\ln(n\hat{p}_k)\right\}-Kd\ln(2n). \qquad (3.11)
\end{aligned}$$

Consistance de AIC et de BIC

L'étude théorique du comportement de certains des critères précédents, comme la consistance, est intéressante. Un critère est dit *consistant en dimension* s'il choisit le vrai modèle avec probabilité un quand la taille de l'échantillon tend vers l'infini. Posons les deux hypothèses suivantes

$$\text{contre}\left\{\begin{array}{ll}\mathrm{H}_0: & \text{modèle } \mathcal{M}_0 \text{ vrai}\\ \mathrm{H}_1: & \text{modèle } \mathcal{M}_1 \text{ vrai.}\end{array}\right.$$

Notons Θ_0 (resp. Θ_1), q_0 (resp. q_1), L_0 (resp. L_1), l'espace des paramètres, le nombre de paramètres libres et le maximum de vraisemblance sous $\mathcal{M}_0$ (resp. $\mathcal{M}_1$).

Supposons tout d'abord que $\mathcal{M}_1$ est plus complexe que $\mathcal{M}_0$, c'est-à-dire $\Theta_0 \subseteq \Theta_1$ (donc $q_0 \leq q_1$). La probabilité de choisir H_1 alors que H_0 vraie (erreur de première espèce) s'écrit pour AIC :

$$\begin{aligned}
\mathrm{P}(\mathrm{AIC}(\mathcal{M}_1) < \mathrm{AIC}(\mathcal{M}_0)) &= \mathrm{P}(-2(\mathrm{L}_0-\mathrm{L}_1) > 2(q_1-q_0)) \\
&\underset{n\to\infty}{\longrightarrow} \mathrm{P}(\chi^2_{q_1-q_0} > 2(q_1-q_0)) \qquad (3.12)\\
&> 0.
\end{aligned}$$

De la même manière pour BIC, en posant $\mu = \mathrm{E}[\chi^2_{q_1-q_0}] = q_1-q_0$ et $\sigma^2 = \mathrm{Var}[\chi^2_{q_1-q_0}] = 2(q_1-q_0)$, on a

$$\begin{aligned}
\mathrm{P}(\mathrm{BIC}(\mathcal{M}_1) < \mathrm{BIC}(\mathcal{M}_0)) &= \mathrm{P}(-2(\mathrm{L}_0-\mathrm{L}_1) > (q_1-q_0)\ln(n)) \\
&\underset{n\to\infty}{\longrightarrow} \mathrm{P}(\chi^2_{q_1-q_0} > (q_1-q_0)\ln(n)) \qquad (3.13)\\
&= \mathrm{P}(\chi^2_{q_1-q_0}-\mu > (q_1-q_0)\ln(n)-\mu) \\
&\leq \mathrm{P}(|\chi^2_{q_1-q_0}-\mu| > (q_1-q_0)\ln(n)-\mu) \\
&\leq \frac{\sigma^2}{((q_1-q_0)\ln(n)-\mu)^2} \quad \text{(inégalité de Chebyshev)} \\
&\underset{n\to\infty}{\longrightarrow} 0.
\end{aligned}$$

Donc les critère AIC et AIC3 (même démonstration que pour AIC) ont une probabilité positive de surestimer le vrai modèle : ils sont inconsistants en dimension. Par contre, BIC ne surestime pas asymptotiquement la dimension du modèle grâce à la pénalisation du maximum de vraisemblance qui dépend de la taille de l'échantillon.

Supposons maintenant que $\mathcal{M}_0$ est plus complexe que $\mathcal{M}_1$ ($\Theta_1 \subset \Theta_0$ donc $q_1 < q_0$). Notons $\chi^2_{q_0-q_1}(\delta_n)$ la distribution du chi-deux avec paramètre de décentrage $\delta_n = n(\boldsymbol{\theta}_0 - \boldsymbol{\theta}_1)'\boldsymbol{F}(\boldsymbol{\theta}_0)(\boldsymbol{\theta}_0 - \boldsymbol{\theta}_1)$, où $\boldsymbol{F}(\boldsymbol{\theta}_0)$ est la matrice de Fisher en $\boldsymbol{\theta}_0$ pour un individu, $\boldsymbol{\theta}_0$ étant le *vrai* paramètre et $\boldsymbol{\theta}_1$ étant son projeté sur le sous-espace Θ_1. Notons aussi $\mu_n = \mathrm{E}[\chi^2_{q_0-q_1}(\delta_n)] = q_0 - q_1 + \delta_n$ et $\sigma^2_n = \mathrm{Var}[\chi^2_{q_0-q_1}(\delta_n)] = 2(q_0 - q_1 + \delta_n)$. La probabilité de première espèce associée à BIC s'écrit

$$\begin{aligned}
\mathrm{P}(\mathrm{BIC}(\mathcal{M}_1) < \mathrm{BIC}(\mathcal{M}_0)) &= \mathrm{P}(-2(\mathrm{L}_1 - \mathrm{L}_0) > (q_0 - q_1)\ln(n)) \\
&\underset{n\to\infty}{\longrightarrow} \mathrm{P}(\chi^2_{q_0-q_1}(\delta_n) > (q_0 - q_1)\ln(n)) \qquad (3.14) \\
&= \mathrm{P}(\mu_n - \chi^2_{q_0-q_1} > \mu_n - (q_0 - q_1)\ln(n)) \\
&\leq \mathrm{P}(|\chi^2_{q_0-q_1}(\delta_n) - \mu_n| > \mu_n - (q_0 - q_1)\ln(n)) \\
&\leq \frac{\sigma^2_n}{(\mu_n - (q_0 - q_1)\ln(n))^2} \quad \text{(inégalité de Chebyshev)} \\
&\underset{n\to\infty}{\longrightarrow} 0.
\end{aligned}$$

Ainsi, asymptotiquement, BIC (on le démontrerait de même pour AIC ou AIC3), ne sous-estime pas la vraie dimension. BIC, ne surestimant ni ne sous-estimant (asymptotiquement) la dimension, est donc consistant, contrairement à AIC ou AIC3.

Remarquons néanmoins que cette consistance de BIC et cette inconsistance de AIC ne sont pas démontrées dans le cas particulier du choix du nombre de classes. En effet, les conditions assurant la distribution du χ^2 et qui sont utilisées aux relations 3.12 à 3.14 ne sont pas valables dans ce cas (voir la discussion à la section 3.3.1).

Résultats pour le nombre de classes

Au travers de simulations de Monte-Carlo intensives, Cutler et Windham [41] comparent les performances des critères AIC, AIC3, BIC et ICOMP pour la détection du nombre de classes K. Il ressort que AIC surestime fortement K, AIC3 le surestime encore mais beaucoup moins et BIC a par contre tendance à le sous-estimer. Ces résultats sont cohérents avec les propriétés de consistance précédentes et aussi avec le fait que, de AIC à BIC, on pénalise de plus en plus la complexité : pour $n > 20$ on a $\tau_{\mathrm{AIC}} < \tau_{\mathrm{AIC3}} < \tau_{\mathrm{BIC}}$. Le critère ICOMP semble meilleur que ces trois critères. De son côté, AWE sous-estime fortement le nombre de classes.

Nous avons repris l'échantillon test de la figure 3.2 pour illustrer le comportement de l'ensemble des critères. Utilisant le modèle $[p\lambda_k C_k]$, on remarque l'effet de la différence de pénalisation entre les critères AIC, AIC3, BIC et ICOMP. Le critère AWE est le seul à détecter deux classes, tous les autres critères en sélectionnant trois.

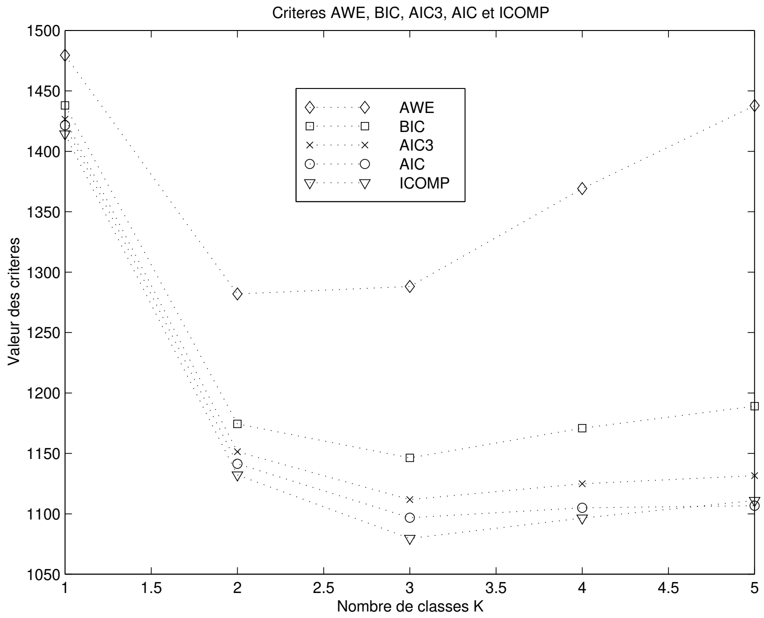


FIG. 3.3 – *Différence de comportement entre les critères AIC, AIC3, BIC, AWE et ICOMP.*

3.3.4 Détection d'un coude de la vraisemblance

Nous avons signalé à la section 3.3.3 que la vraisemblance ne pouvait être utilisée seule comme critère pour choisir le nombre de classes. En effet, ce critère aurait tendance à toujours surestimer le nombre de classes car il croît systématiquement avec lui. Néanmoins, plusieurs auteurs, dont Cutler et windham [41], ont remarqué que le maximum de vraisemblance avait une croissance moins forte dès que le «bon» nombre de classes était dépassé. On parle dans ce cas d'un coude (*elbow*) de la vraisemblance. Ce coude est par exemple bien visibible sur la figure 3.2 lorsque l'on passe de trois à quatre classes. Le côté «visuel» de ce coude est subjectif et l'objectif de cette section est d'automatiser la détection de celui-ci en introduisant une notion de précision sur les données.

Le critère EL de Cutler et Windham

Pour trouver le coude de la vraisemblance, Cutler et Windham [41] ont proposé le critère EL (*Elbow Likelihood*). Pour choisir entre 1 à K_{sup} classes, la procédure associée à ce critère s'énonce ($\mathrm{L}(K)$ désigne le maximum de log-vraisemblance pour K classes) :

- **initialisation** : partir de $K = 1$;
- **itération** :
  - si $K = K_{sup}$ ou $\mathrm{L}(K+1) \leq \mathrm{L}(K) + 0.01\mathrm{L}(1)$ alors $\hat{K} = K$ et arrêt ;
  - sinon incrémenter K de un et itérer.

Le seuil $\mathrm{L}(K+1) \leq \mathrm{L}(K) + 0.01\mathrm{L}(1)$ semble déterminé empiriquement et les expériences numériques de leur article montrent clairement une surestimation du nombre de classes. Nous proposons alors de transformer la définition de ce seuil en une connaissance que l'on pourrait avoir a priori sur la précision des données.

L'hypothèse de précision sur les données

Soit un n-échantillon de $\mathbb{R}^d$ $\mathcal{Y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$ qui soit une réalisation de n variables aléatoires $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$ i.i.d. de densité de mélange gaussien $f(\boldsymbol{y}|\boldsymbol{\theta})$. Supposons que ces n valeurs ne soient pas directement observables et qu'il soit nécessaire d'utiliser un instrument pour les mesurer. Par exemple, $y_i \in \mathbb{R}$ représente le temps de course du i^{e} coureur ($1 \leq i \leq n$) et on utilise un chronomètre pour mesurer y_i. L'instrument fournit alors un ensemble de mesures $\mathcal{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$, chaque $\boldsymbol{x}_i$ étant la mesure associée à la valeur $\boldsymbol{y}_i$. Idéalement, on a $\boldsymbol{x}_i = \boldsymbol{y}_i$ ($1 \leq i \leq n$), mais l'instrument peut fournir une mesure $\boldsymbol{x}_i$ différente de la valeur $\boldsymbol{y}_i$ à cause d'une *précision* propre à cet appareil. Nous faisons l'hypothèse qu'à une mesure $\boldsymbol{x}$ donnée, la vraie valeur $\boldsymbol{y}$ associée soit une réalisation d'une variable aléatoire $\boldsymbol{Y}$ de densité $\pi_\gamma(\boldsymbol{y}|\boldsymbol{x})$ avec

$$\pi_\gamma(\boldsymbol{y}|\boldsymbol{x}) = \prod_{j=1}^{d} \mathcal{U}_{[x^j - \gamma^j, x^j + \gamma^j]}(\boldsymbol{y}^j)$$

où $\boldsymbol{y}^j$ est la j^{e} composante de $\boldsymbol{y}$ ($1 \leq j \leq d$), $\boldsymbol{x}^j$ est la j^{e} composante de $\boldsymbol{x}$, $\boldsymbol{\gamma}^j$ est la j^{e} composante de $\boldsymbol{\gamma} = (\boldsymbol{\gamma}^1, \ldots, \boldsymbol{\gamma}^d)$ et $\mathcal{U}_{[x^j - \gamma^j, x^j + \gamma^j]}(\boldsymbol{y}^j)$ est la densité uniforme sur l'intervalle $[x^j - \gamma^j, x^j + \gamma^j]$. La précision de l'instrument de mesure est donc définie par la loi π conditionnelle à la valeur $\boldsymbol{x}$ et de paramètre γ. Par exemple, un chronomètre avec une précision au 10^{e} de seconde revient à prendre $\gamma = 0.05$. Ainsi, une mesure $x = 9.8$ s indique que la valeur y est comprise entre 9.75 s et 9.85 s, chaque valeur de cet intervalle étant equiprobable. Nous supposons dans la suite que cette précision est une caractéristique *connue* de l'instrument de mesure.

L'estimation du paramètre du mélange est délicate dans le contexte de précision que nous venons d'énoncer car on ne connaît pas l'échantillon $\mathcal{Y}$ et seules les valeurs $\mathcal{X}$ et la précision de l'instrument sont disponibles. Nous proposons alors d'estimer ce paramètre $\boldsymbol{\theta}$ en moyennant la log-vraisemblance sur tous les échantillons $\mathcal{Y}$ possibles. Notant

$$\mathrm{L}(\boldsymbol{\theta}, \mathcal{Y}) = \ln(f(\mathcal{Y}|\boldsymbol{\theta}))$$

la log-vraisemblance de $\boldsymbol{\theta}$ obtenue avec l'échantillon $\mathcal{Y}$, la moyenne de cette vraisemblance sur tous les $\mathcal{Y}$ s'écrit

$$\begin{aligned} \mathrm{L}_\gamma(\boldsymbol{\theta}) &= \mathrm{E}^{\pi_\gamma}[\mathrm{L}(\boldsymbol{\theta}, (\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n))|\mathcal{X}] \\ &= \int_{\mathbb{R}^{dn}} \mathrm{L}(\boldsymbol{\theta}, \mathcal{Y})\ \pi_\gamma(\mathcal{Y}|\mathcal{X}) d\mathcal{Y}, \end{aligned}$$

avec la loi jointe

$$\pi_\gamma(\mathcal{Y}|\mathcal{X}) = \prod_{i=1}^{n} \pi_\gamma(\boldsymbol{y}_i|\boldsymbol{x}_i).$$

On peut bien écrire la loi jointe comme un produit car tous les $\boldsymbol{y}_i$ $(1 \leq i \leq n)$ sont indépendants entre eux. Nous retenons finalement l'estimateur $\hat{\boldsymbol{\theta}}$ qui maximise la log-vraisemblance moyenne $\mathrm{L}_\gamma(\boldsymbol{\theta})$.

Une fois $\hat{\boldsymbol{\theta}}$ connu, on cherche l'échantillon $\mathcal{Y}^+$ qui maximise la vraisemblance $\mathrm{L}(\hat{\boldsymbol{\theta}}, \mathcal{Y})$. Pour K classes, nous notons $\mathrm{L}_\gamma^+(K) = \max_\mathcal{Y}(\hat{\boldsymbol{\theta}}, \mathcal{Y})$. De même, on a $\mathrm{L}_\gamma^-(K) = \min_\mathcal{Y}(\hat{\boldsymbol{\theta}}, \mathcal{Y})$. Ainsi, $\mathrm{L}_\gamma^+(K)$ et $\mathrm{L}_\gamma^-(K)$ sont respectivement les vraisemblances maximales et minimales possibles avec $\hat{\boldsymbol{\theta}}$ fixé parmi tous les échantillons admissibles $\mathcal{Y}$. Nous proposons ensuite la procédure suivante pour trouver le nombre de classes :

- **initialisation** : partir de $K = 1$;
- **itération** :
  - si $K = K_{sup}$ ou $\mathrm{L}_\gamma^-(K+1) \leq \mathrm{L}_\gamma^+(K)$ alors $\hat{K} = K$ et arrêt ;
  - sinon incrémenter K de un et itérer.

Cette procédure ressemble à celle utilisée par le critère EL mais le seuil est déterminé différemment. Cette procédure consiste simplement à préférer K classes à $K+1$ classes si la vraisemblance maximale avec K classes est supérieure à la vraisemblance minimale avec $K+1$ classes. En d'autres termes, on retient K classes si, grâce à la précision, on peut «mieux décrire» les données avec K classes que la moins bonne des descriptions avec $K+1$. On peut voir une illustration de ce principe sur la figure 3.4.

Cette méthode se comprend bien mais présente deux défauts importants :

- trouver le minimum ou le maximum est difficile et donc $\mathrm{L}_\gamma^+(K)$ et $\mathrm{L}_\gamma^-(K)$ sont estimés de manière peu robuste ;

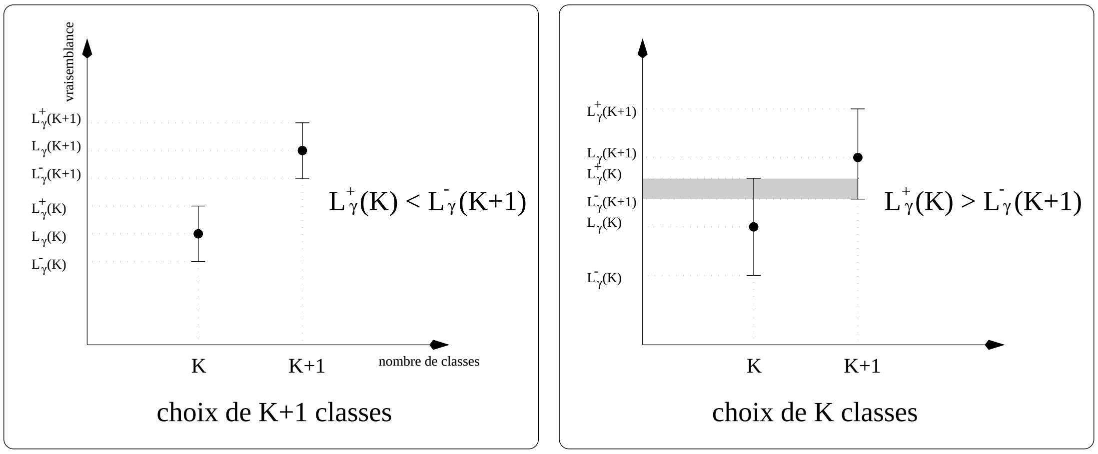


FIG. 3.4 – *Choix entre K et $K+1$ classes par le nouveau seuil de EL.*

– une telle procédure exclut bien sûr toute précision qui n'utilise pas une loi uniforme (ou au moins une loi à support borné). Une loi normale, par exemple, ferait toujours choisir le cas monoclasse.

Pour remédier à ces deux problèmes, nous proposons de définir $\mathrm{L}^+_\gamma(K)$ et $\mathrm{L}^-_\gamma(K)$ de façon différente en conservant la même philosophie. Plutôt que de prendre l'échantillon $\mathcal{Y}$ qui maximise $\mathrm{L}(\hat{\boldsymbol{\theta}}, \mathcal{Y})$, nous définissons $\mathrm{L}^+_\gamma(K)$ par la moyenne des vraisemblances $\mathrm{L}(\hat{\boldsymbol{\theta}}, \mathcal{Y})$ sur tous les échantillons $\mathcal{Y}$ dont la vraisemblance est supérieure à $\mathrm{L}_\gamma(\hat{\boldsymbol{\theta}})$, la vraisemblance moyenne maximale sur tous les $\mathcal{Y}$ possibles. Cette log-vraisemblance s'écrit :

$$\mathrm{L}^+_\gamma(\hat{\boldsymbol{\theta}}) = \mathrm{E}^{\pi^+_\gamma}[\mathrm{L}(\boldsymbol{\theta}, (\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n))|\mathcal{X}],$$

avec la densité jointe

$$\pi^+_\gamma(\mathcal{Y}|\mathcal{X}) \propto \pi_\gamma(\mathcal{Y}|\mathcal{X}) \ \cdot\ \mathbb{1}_{\{\mathrm{L}(\hat{\theta},\mathcal{Y}) \geq \mathrm{L}_\gamma(\hat{\theta})\}}(\mathcal{Y}).$$

De manière symétrique, notons

$$\mathrm{L}^-_\gamma(\hat{\boldsymbol{\theta}}) = \mathrm{E}^{\pi^-_\gamma}[\mathrm{L}(\boldsymbol{\theta}, (\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n))|\mathcal{X}],$$

la log-vraisemblance moyenne sur tous les échantillons $\mathcal{Y}$ qui permettent une vraisemblance $\mathrm{L}(\hat{\boldsymbol{\theta}}, \mathcal{Y})$ inférieure à la log-vraisemblance moyenne $\mathrm{L}_\gamma(\hat{\boldsymbol{\theta}})$. Le terme $\pi^-_\gamma(\mathcal{Y}|\mathcal{X})$ désigne la densité jointe

$$\pi^-_\gamma(\mathcal{Y}|\mathcal{X}) \propto \pi_\gamma(\mathcal{Y}|\mathcal{X}) \ \cdot\ \mathbb{1}_{\{\mathrm{L}(\hat{\theta},\mathcal{Y}) \leq \mathrm{L}_\gamma(\hat{\theta})\}}(\mathcal{Y}).$$

Nous considérons ensuite la même procédure de choix du coude.

Par ce changement, nous gommons les défauts précédents : meilleure robustesse pour calculer une moyenne qu'un extremum, possibilité d'utiliser une loi π_γ à support non borné comme la loi normale.

Choix de la précision

Il est important de préciser de nouveau que la précision $\boldsymbol{\gamma}$ est une caractéristique connue par l'utilisateur. Sa détermination est facile à partir des caractéristiques de l'instrument de mesure. De plus le terme $\boldsymbol{\gamma}$ a une signification très concrète donc est compréhensible par chacun sans rentrer dans le détail de la théorie.

Mais il faut rester conscient que la valeur de $\boldsymbol{\gamma}$ a une incidence directe sur la détection du seuil de la vraisemblance. Par exemple, si $\|\boldsymbol{\gamma}\| = 0$, l'instrument de mesure est parfait et détecte le nombre maximum de classes car on retrouve la vraisemblance classique. Au contraire, si $\|\boldsymbol{\gamma}\|$ est «très grand», l'instrument de mesure est très imprécis et on détectera systématiquement le cas monoclasse.

Mise en œuvre de la méthode

La détection d'un coude de la vraisemblance par la méthode précédente nécessite de trouver l'estimateur $\hat{\boldsymbol{\theta}}$ qui maximise la log-vraisemblance moyenne $\mathrm{L}_\gamma(\boldsymbol{\theta})$, puis de calculer les log-vraisemblances $\mathrm{L}_\gamma^+(\hat{\boldsymbol{\theta}})$ et $\mathrm{L}_\gamma^-(\hat{\boldsymbol{\theta}})$. Nous présentons alors les moyens employés pour résoudre ces problèmes.

Nous proposons d'utiliser une méthode de Monte-Carlo pour estimer le paramètre $\boldsymbol{\theta}$ qui maximise la log-vraisemblance moyenne $\mathrm{L}_\gamma(\boldsymbol{\theta})$. Nous générons m échantillons $\mathcal{Y}_1, \ldots, \mathcal{Y}_m$ i.i.d. suivant la loi $\pi_\gamma(\mathcal{Y}|\mathcal{X})$, puis nous estimons la log-vraisemblance moyenne par :

$$\hat{\mathrm{L}}_\gamma(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m} \mathrm{L}(\boldsymbol{\theta}, \mathcal{Y}_i).$$

Puisque $\mathrm{L}(\boldsymbol{\theta}, \mathcal{Y}) = \ln(f(\mathcal{Y}|\boldsymbol{\theta}))$, cet estimateur s'écrit aussi

$$\begin{aligned}\hat{\mathrm{L}}_\gamma(\boldsymbol{\theta}) &= \frac{1}{m}\ln(\prod_{i=1}^{m} f(\mathcal{Y}_i|\boldsymbol{\theta})) \\ &= \frac{1}{m}\mathrm{L}(\boldsymbol{\theta}, (\mathcal{Y}_1, \ldots, \mathcal{Y}_m)).\end{aligned}$$

Pour trouver $\hat{\boldsymbol{\theta}}$ qui maximise $\hat{\mathrm{L}}_\gamma(\boldsymbol{\theta})$, il suffit donc de maximiser la vraisemblance $\mathrm{L}(\boldsymbol{\theta}, (\mathcal{Y}_1, \ldots, \mathcal{Y}_m))$, log-vraisemblance s'appuyant sur l'échantillon $(\mathcal{Y}_1, \ldots, \mathcal{Y}_m)$, échantillon avec $n \times m$ éléments.

Pour ce faire on peut utiliser l'algorithme EM de la façon suivante. Si $\|\boldsymbol{\gamma}\|$ est «petit» ($\|\cdot\|$ étant une norme quelconque), cette hypothèse étant souvent raisonnable, on peut faire tourner plusieurs fois EM avec l'échantillon $\mathcal{X}$ en partant d'initialisations aléatoires. Pour trouver $\hat{\boldsymbol{\theta}}$, on fait ensuite tourner EM une seule fois avec l'échantillon $(\mathcal{Y}_1, \ldots, \mathcal{Y}_m)$ en l'initialisant avec le paramètre optimum obtenu précédemment.

Ayant obtenu $\hat{\boldsymbol{\theta}}$ par la méthode précédente, on génère m^+ échantillons $\mathcal{Y}_1^+$ à $\mathcal{Y}_{m^+}^+$ de manière i.i.d. à partir de la loi $\pi_\gamma^+(\mathcal{Y}|\mathcal{X})$, puis on estime $\mathrm{L}_\gamma^+(\hat{\boldsymbol{\theta}})$ par une méthode de Monte-Carlo :

$$\hat{\mathrm{L}}_\gamma^+(\hat{\boldsymbol{\theta}}) = \frac{1}{m^+}\sum_{i=1}^{m^+} \mathrm{L}(\hat{\boldsymbol{\theta}}, \mathcal{Y}_i^+).$$

De la même façon, on génère m^- échantillons $\mathcal{Y}_1^-, \ldots, \mathcal{Y}_{m^-}^-$ de manière i.i.d. à partir de la loi $\pi_\gamma^-(\mathcal{Y}|\mathcal{X})$ puis on estime $\mathrm{L}_\gamma^-(\hat{\boldsymbol{\theta}})$ par

$$\hat{\mathrm{L}}_\gamma^-(\hat{\boldsymbol{\theta}}) = \frac{1}{m^-}\sum_{i=1}^{m^-} \mathrm{L}(\hat{\boldsymbol{\theta}}, \mathcal{Y}_i^-).$$

En pratique, on partitionne simplement l'ensemble des échantillons $\mathcal{Y}_1, \ldots, \mathcal{Y}_m$ qui proviennent de manière i.i.d. de $\pi_\gamma(\mathcal{Y}|\mathcal{X})$ en deux sous-ensembles. Le premier sous-ensemble regroupe les échantillons $\mathcal{Y}_i$ $(1 \leq i \leq m)$ tels que $\hat{\mathrm{L}}(\hat{\boldsymbol{\theta}}, \mathcal{Y}_i) \geq \hat{\mathrm{L}}_\gamma(\hat{\boldsymbol{\theta}})$. On obtient alors m^+ échantillons $\mathcal{Y}_1^+, \ldots, \mathcal{Y}_{m^+}^+$ i.i.d de loi $\pi_\gamma^+(\mathcal{Y}|\mathcal{X})$. Le deuxième sous-ensemble regroupe les $m^- = m - m^+$ échantillons restant. Ils correspondent aux échantillons $\mathcal{Y}_1^-, \ldots, \mathcal{Y}_{m^-}^-$ i.i.d. de loi $\pi_\gamma^-(\mathcal{Y}|\mathcal{X})$.

Stabilité des estimateurs

Nous nous proposons d'étudier la stabilité des estimateurs $\hat{\mathrm{L}}_\gamma^+(\hat{\boldsymbol{\theta}})$ et $\hat{\mathrm{L}}_\gamma^-(\hat{\boldsymbol{\theta}})$ obtenus par les méthodes précédentes. Il est important que ces deux estimateurs n'aient pas trop de variance car ils servent directement à la détection du coude de la vraisemblance.

Pour ce faire, prenons l'échantillon test de la figure 3.2 et posons une précision arbitraire $\boldsymbol{\gamma}^1 = \boldsymbol{\gamma}^2 = 0.05$. Pour tous les nombres de classes de 1 à 5 avec le modèle $[p\lambda_k C_k]$, nous calculons 10 fois l'estimateur $\hat{\boldsymbol{\theta}}$ et les estimateurs $\hat{\mathrm{L}}_\gamma^+(\hat{\boldsymbol{\theta}})$ et $\hat{\mathrm{L}}_\gamma^-(\hat{\boldsymbol{\theta}})$ (en prenant $m = 100$). Les moyennes et les écart-types de ces deux derniers estimateurs sont reportés sur le tableau 3.4. Il ressort que les écart-types sont très faibles, ce qui indique une bonne stabilité de ceux-ci.

Estimateurs		K				
		1	2	3	4	5
$\hat{\mathrm{L}}_\gamma^+(\hat{\boldsymbol{\theta}})$	moyenne	-705.54	-560.38	-533.06	-532.23	-528.08
	écart-type	0.05	0.09	0.15	0.05	0.13
$\hat{\mathrm{L}}_\gamma^-(\hat{\boldsymbol{\theta}})$	moyenne	-706.22	-562.10	-535.01	-534.04	-530.03
	écart-type	0.04	0.09	0.16	0.14	0.14

TAB. 3.4 – *Moyennes et écart-types des estimateurs $\hat{L}_\gamma^+(\hat{\boldsymbol{\theta}})$ et $\hat{L}_\gamma^-(\hat{\boldsymbol{\theta}})$.*

Remarquons aussi que cette précision détecte le coude de la vraisemblance à trois classes (figure 3.5). Il faut néanmoins être prudent car la précision est arbitraire sur ce jeu de données (l'échantillon test est un jeu de données simulé).

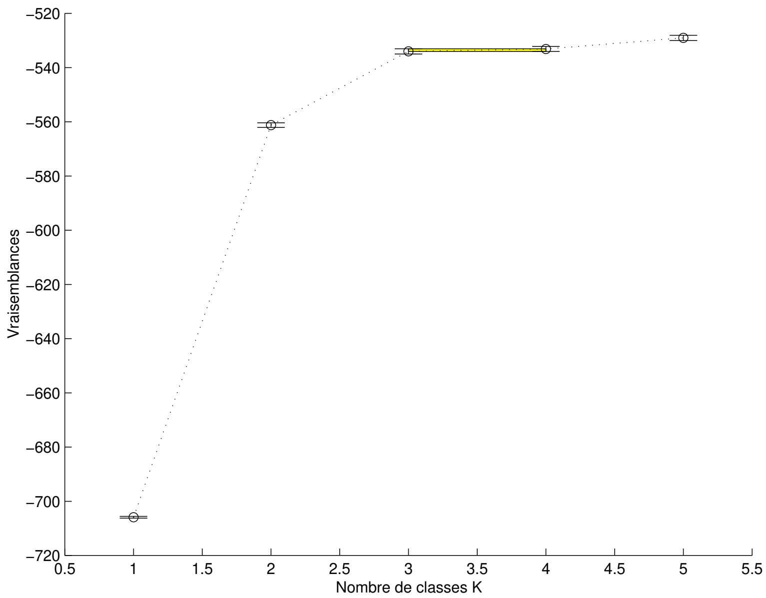


FIG. 3.5 – *Détection de trois classes dans l'échantillon test de la figure 3.2 avec une précision* $\boldsymbol{\gamma}^1 = \boldsymbol{\gamma}^2 = 0.05$ *et avec le modèle* $[p\lambda_k C_k]$.

3.4 Critères en classification automatique

Cette section présente différents critères pour choisir un modèle en classification automatique utilisant un modèle gaussien de mélange. Tout d'abord l'approche mélange est utilisable dans ce contexte. Ensuite deux ensembles de critères plus spécifiques à la classification automatique sont présentés. Le premier ensemble regroupe les critères mesurant la classifiabilité : ces critères favorisent les modèles produisant des classes bien séparées mais ils ont le défaut de ne pouvoir choisir entre une et plusieurs classes. Le second ensemble regroupe des critères qui pénalisent la vraisemblance par un critère de classifiabilité. Ce type de critères est très intéressant à plusieurs titres :

– il évite les problèmes liés aux critères de classifiabilité car on peut comparer une et plusieurs classes ;

– on peut retrouver la vraisemblance classifiante donc interpréter celle-ci comme une pénalisation de la vraisemblance par un terme de classifiabilité ;

– on peut aussi interpréter la vraisemblance classifiante ainsi retrouvée comme une pénalisation de certains critères classiques en classification, comme le critère des k-means.

3.4.1 Utilisation de l'approche mélange

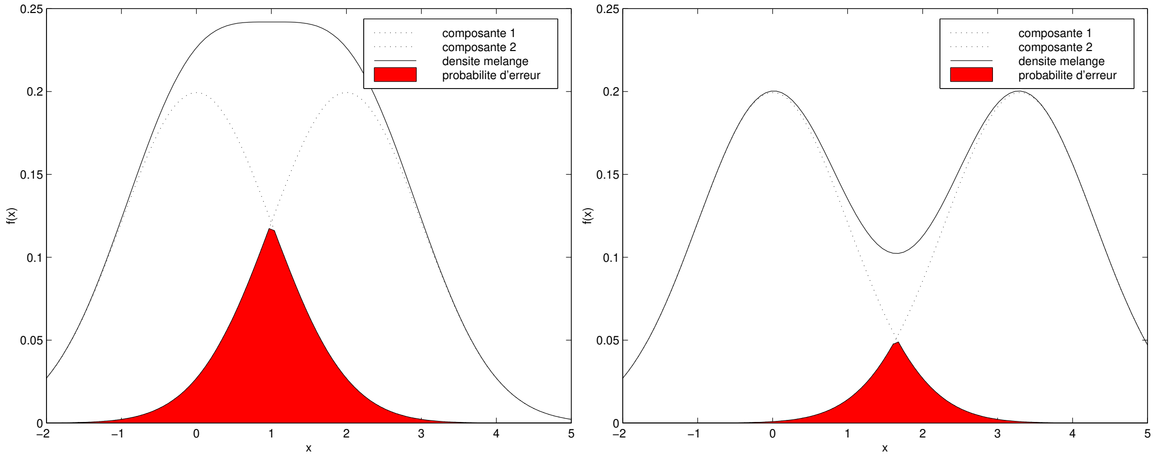
La classification automatique gaussienne s'appuie sur l'hypothèse de mélange gaussien et les méthodes et les critères de la section 3.3 sont donc utilisables. On peut ainsi employer les tests d'hypothèses, l'approche bayésienne avec les critères BIC et AWE et les autres critères pénalisant la complexité que sont AIC, AIC3 et ICOMP. Néanmoins, ces méthodes générales ne prennent pas en considération la spécificité de la classification automatique qui est d'obtenir une partition. Les parties suivantes présentent des critères plus adaptés à ce cas particulier : les critères classifiabilité et les critères pénalisant la vraisemblance par la classifiabilité.

3.4.2 Critères de classifiabilité

Principe

Les critères de classifiabilité ont pour objectif de retenir le modèle qui produit les classes les plus séparées. En effet, des classes bien séparées conduisent à un partitionnement facile des données : les probabilités t_{ik} d'appartenance aux classes a posteriori sont toutes proches soit de 0, soit de 1. On parle alors de données bien *classifiables*. Cette approche a surtout été utilisée pour la recherche du nombre de classes. En effet, certains auteurs ont considéré le problème du nombre de classes dans un modèle de mélange gaussien avec ce point de vue classification. En d'autres termes, ils considèrent qu'une classe a lieu d'exister uniquement si elle est «suffisamment» séparée des autres. Si deux classes ont des paramètres assez proches, la probabilité d'erreur de classement théorique $\mathbf{P}_e$ entre ces deux classes est forte (figure 3.6). Cela signifie qu'il est difficile de décider avec certitude de l'appartenance des individus à une de ces deux classes. Si les individus d'une première classe peuvent aussi bien appartenir à une seconde classe, il est naturel de se poser la question de l'utilité de cette deuxième classe. L'expert du domaine relatif aux données, utilisateur final du résultat de classification, aura certainement du mal à interpréter de manière différente des classes aussi semblables.

Les critères que nous présentons, au contraire des critères de l'approche mélange en général, ont peu de fondement théorique : ils s'appuient sur des notions qui peuvent sembler de «bon sens» mais qui restent néanmoins subjectives. Un défaut important qu'ils présentent tous est une difficulté (voire une impossibilité) de comparer un modèle à une seule classe et un modèle à au moins deux classes. En effet le cas monoclasse est souvent «absorbant» c'est-à-dire qu'il optimise d'office le critère (critères PC, MIR, E et LP), ou bien conduit à un critère incalculable théoriquement (critère NEC). Le critère E n'avait jamais, à ce jour,



15 % d'erreur 5 % d'erreur

FIG. 3.6 – *Illustration de classes peu séparées et bien séparées.*

été utilisé seul pour trouver le nombre de classes, tandis que le critère LP est totalement nouveau.

Critère PC

Une mesure pour trouver le nombre de classes dans ce contexte de mélange a été proposée par Bezdek [11] sous le nom de PC (*Partition Coefficient*). Il est à maximiser et s'énonce

$$\mathrm{PC}(K) = \sum_{i=1}^{n} \sum_{k=1}^{K} t_{ik}^2,$$

où les t_{ik} (probabilités d'appartenance a posteriori aux classes) sont obtenues au point $\hat{\boldsymbol{\theta}}_L$ du maximum de vraisemblance. Il a l'inconvénient de ne pouvoir tester $K = 1$ contre $K > 1$ car $\mathrm{PC}(K=1) = 1 \geq \mathrm{PC}(K > 1)$. Ce critère a tendance en pratique à sous-estimer le nombre de classes. Utilisant l'échantillon test de la figure 3.2 et le modèle $[p\lambda_k C_k]$, la figure 3.7 donne un exemple d'utilisation du critère PC.

Critère MIR et variantes

MIR (*Minimum Information Ratio*) a été introduit par Windham et Cutler [90]. Il correspond à la plus petite valeur propre de la matrice $\boldsymbol{F}_C^{-1}(\hat{\boldsymbol{\theta}}_L)\boldsymbol{F}(\hat{\boldsymbol{\theta}}_L)$, où $\boldsymbol{F}(\hat{\boldsymbol{\theta}}_L)$ est la matrice d'information de Fisher pour une seule observation en $\hat{\boldsymbol{\theta}}_L$ et $\boldsymbol{F}_C(\hat{\boldsymbol{\theta}}_L)$ est la matrice d'information de Fisher pour une seule observation en

$\hat{\boldsymbol{\theta}}_L$ pour l'échantillon classifié. Ces matrices sont définies par

$$\begin{aligned}
\boldsymbol{F}(\hat{\boldsymbol{\theta}}_L) &= -\mathrm{E}_X \left[\left. \frac{\partial^2 \ln(f(\boldsymbol{X}|\boldsymbol{\theta}))}{\partial \boldsymbol{\theta}^2} \right|_{\hat{\boldsymbol{\theta}}_L} \right] \\
\boldsymbol{F}_C(\hat{\boldsymbol{\theta}}_L) &= -\mathrm{E}_X \left[\sum_{k=1}^{K} t_k(\boldsymbol{X}|\hat{\boldsymbol{\theta}}_L) \left. \frac{\partial^2 \ln(p_k \phi_k(\boldsymbol{X}|\boldsymbol{a}_k))}{\partial \boldsymbol{\theta}^2} \right|_{\hat{\boldsymbol{\theta}}_L} \right].
\end{aligned}$$

Le quotient $\boldsymbol{F}_C^{-1}(\hat{\boldsymbol{\theta}}_L)\boldsymbol{F}(\hat{\boldsymbol{\theta}}_L)$ des matrices d'information mesure la proportion d'information sur $\boldsymbol{\theta}$ disponible sans connaître les classes. Dans le contexte de mélange, il peut s'interpréter comme la capacité des données à distinguer les différentes densités. Le terme MIR prend des valeurs comprises entre 0 et 1. Un MIR élevé suggère une bonne classification et, au contraire, un moindre MIR est un signe de faible classification.

Si l'algorithme EM est utilisé pour obtenir les estimateurs des paramètres, alors MIR peut être estimé en utilisant le taux de convergence de EM. Donc, si $\boldsymbol{\theta}^m$ $(m > 0)$ est un vecteur de paramètres estimés à une m^{e} itération de EM, on a pour m assez grand et une norme $\|\cdot\|$ convenable

$$\widehat{\mathrm{MIR}}(K) = 1 - \frac{\|\boldsymbol{\theta}^{m+1} - \boldsymbol{\theta}^m\|}{\|\boldsymbol{\theta}^m - \boldsymbol{\theta}^{m-1}\|}. \tag{3.15}$$

Comme le critère PC, MIR ne peut distinguer entre $K = 1$ et $K > 1$ puisque que $\mathrm{MIR}(K = 1) = 1 \geq \mathrm{MIR}(K > 1)$. De plus, MIR produit des difficultés numériques dans le calcul du quotient de l'équation 3.15 qui, à la convergence de EM, devient proche de la forme indéterminée $\frac{0}{0}$.

À l'origine, MIR faisait partie d'une procédure nommée MIREV, E désignant la phase d'estimation de MIR que l'on vient de voir et qui donne comme résultat un nombre $\hat{K}$ de classes, V concernant une nouvelle phase dite de validation. Cette dernière est composée d'un bootstrap pour estimer d'une certaine manière une mesure de confiance sur le $\hat{K}$ obtenu à la phase E. À chaque nouvel échantillon $\mathcal{X}_r$ du bootstrap $(1 \leq r \leq R)$, on recalcule MIR_r pour obtenir un nombre de classes $\hat{K}_r$ par la même méthode qu'à la phase E. La mesure de confiance pour $\hat{K}$ est alors le rapport $\frac{\text{nombre de fois où } \hat{K}_r = \hat{K}}{R}$.

De nombreuse variantes de MIR existent : $\mathrm{AMIR}(K) = \mathrm{MIR}(K)(\mathrm{L}(K) - \mathrm{L}(1))$ (*Adjusted MIR*) où $\mathrm{L}(K)$ est le maximum de vraisemblance pour K classes, $\mathrm{ANC}(K) = (K-1)\mathrm{MIR}(K)$ (*Adjusted Number of Components*), WID (*Within Component Discrepancy*). On se reportera à Cutler et Windham [41].

MIR et les critères dérivés ont tendance à sous-estimer le nombre de classes et se montrent en général peu convaincants (Cutler et Windham [41]). La figure 3.7 illustre le comportement sur un échantillon test de MIR, AMIR et ANC avec le modèle $[p\lambda_k C_k]$: les critères MIR et AMIR détectent deux classes, tandis que ANC en sélectionne trois.

Critère NEC

Plus récemment, Celeux et Soromenho [40] ont proposé le critère NEC (*Normalized Entropy Criterion*) dérivé de la relation d'Hathaway [58] (équation 2.6). Il s'énonce

$$\mathrm{NEC}(K) = \frac{\mathrm{E}(K)}{\mathrm{L}(K) - \mathrm{L}(1)} \tag{3.16}$$

avec $\mathrm{L}(K)$ le maximum de vraisemblance pour K classes et $\mathrm{E}(K)$ l'entropie définie par l'équation 2.8 avec la partition floue obtenue par l'estimateur du maximum de vraisemblance.

L'entropie $\mathrm{E}(K)$ mesure une certaine qualité de la classification des données en K classes. La différence $\mathrm{L}(K) - \mathrm{L}(1)$ mesure la qualité de description de ces mêmes données sous K classes par rapport à une seule classe. NEC est alors censé réaliser un *compromis* entre ces deux mesures. Plus faible sera NEC, meilleure sera ce compromis. Remarquons que NEC(1) est incalculable en l'état car c'est une forme indéterminée ($\mathrm{E}(1) = 0$, $\mathrm{L}(1) - \mathrm{L}(1) = 0$). Pour résoudre ce problème, Celeux et Soromenho ont proposé la procédure empirique de choix suivante dans le cas de mélange multivarié gaussien :

1. estimer les paramètres du mélange par maximum de vraisemblance pour différentes valeurs de K ($2 \leq K \leq K_{sup}$) et choisir K^* minimisant $\mathrm{NEC}(K)$;

2. pour décider $K = 1$ ou $K = K^*$, estimer les paramètres d'un mélange de K^* classes gaussiennes de centres égaux $\boldsymbol{\mu}_1 = \cdots = \boldsymbol{\mu}_{K^*} = \bar{\boldsymbol{x}}$ ($\bar{\boldsymbol{x}}$ moyenne empirique de $\mathcal{X}$) et de proportions égales $p_1 = \cdots = p_{K^*} = \frac{1}{K^*}$. Les paramètres libres à estimer sont donc seulement les matrices de variance. On obtient ensuite une entropie $\tilde{\mathrm{E}}(1)$ et un logarithme de la vraisemblance $\tilde{\mathrm{L}}(1)$ et on pose alors

$$\mathrm{NEC}(1) = \frac{\tilde{\mathrm{E}}(1)}{\tilde{\mathrm{L}}(1) - \mathrm{L}(1)}.$$

Le choix monoclasse contre K^* classes se fait si $\mathrm{NEC}(1) \leq \mathrm{NEC}(K^*)$.

Au vu des résultats de simulations (Celeux et Soromenho [40]), NEC semble donner de bons résultats car de l'ordre du critère ICOMP de Bozdogan. Utilisant l'échantillon test de la figure 3.2 avec le modèle $[p\lambda_k C_k]$, la figure 3.7 donne un exemple d'utilisation du critère NEC. On remarque que la procédure de Celeux et Soromenho pour calculer NEC dans le cas monoclasse semble adaptée dans ce cas précis.

Critère d'entropie et critère LP

On peut aussi songer à utiliser le critère d'entropie seul

$$\mathrm{E}(K) = \mathrm{E}(\boldsymbol{t}) = -\sum_{i=1}^{n}\sum_{k=1}^{K} t_{ik}\ln(t_{ik}),$$

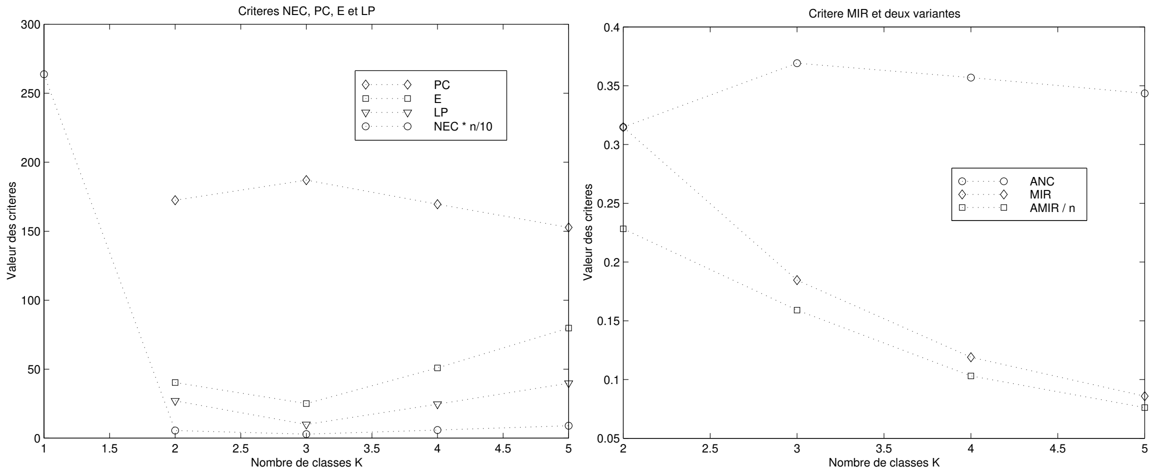


FIG. 3.7 – *Exemple d'utilisation des critères PC, E, LP, NEC, MIR et variantes de MIR (AMIR et ANC).*

la partition floue $\boldsymbol{t} = \{t_{ik}\}$ étant obtenue à l'optimum de la vraisemblance. On retient alors le nombre K de classes qui minimise $\mathrm{E}(K)$. Malheureusement, comme l'a vu précédemment avec le critère NEC, on a $\mathrm{E}(1) = 0$ et $\mathrm{E}(K) \geq 0$, donc on ne peut comparer le cas monoclasses aux autres nombres de classes.

Un critère du même genre est le critère que nous appelons LP pour *Logarithme de la Probabilité de la Partition.* À notre connaissance, ce critère est nouveau. Il est donné par

$$\mathrm{LP}(K) = \mathrm{LP}(\boldsymbol{t}, \boldsymbol{z}) = -\sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik}\ln(t_{ik}), \tag{3.17}$$

où $\boldsymbol{t} = \{t_{ik}\}$ est la partition floue obtenue en maximisant la vraisemblance et $\boldsymbol{z} = \{z_{ik}\}$ est la partition obtenue par MAP de $\boldsymbol{t}$. Ce terme LP correspond à l'opposé du logarithme de la probabilité de la partition a posteriori (d'où son nom). Cette probabilité s'écrit en effet

$$\mathrm{P}(\boldsymbol{z}|\mathcal{X}, \boldsymbol{\theta}) = \prod_{i=1}^{n}\prod_{k=1}^{K} t_{ik}^{z_{ik}}.$$

Le critère LP a le même défaut que le critètre d'entropie car $\mathrm{LP}(1) = 0$ et $\mathrm{LP}(K) \geq 0$. Utilisant l'échantillon test de la figure 3.2 et le modèle $[p\lambda_k C_k]$, la figure 3.7 donne un exemple d'utilisation des critères E et LP. On remarque que la courbe de LP est au dessous de celle de E (nous reviendrons sur cette propriété dans une section suivante), mais que les deux courbes ont néanmoins des formes très semblables.

Les deux critères E et LP ont des formes assez similaires et nous proposons de montrer qu'ils peuvent s'interpréter tous deux au sens de la divergence de Kullback-Leibler. Notant $\boldsymbol{c} = \{c_{ik}\}$ et $\boldsymbol{t} = \{t_{ik}\}$ deux partitions floues, les vecteurs $\boldsymbol{c}_i = (c_{i1}, \ldots, c_{iK})$ et $\boldsymbol{t}_i = (t_{i1}, \ldots, t_{iK})$ $(1 \leq i \leq n)$ correspondent à des distributions de probabilités discrètes et la divergence de Kullback-Leibler entre ces deux distributions s'écrit

$$\mathrm{KL}(\boldsymbol{c}_i, \boldsymbol{t}_i) = \sum_{k=1}^{K} \ln\left[\frac{c_{ik}}{t_{ik}}\right] c_{ik}.$$

Nous définissons la divergence de Kullback-Leibler entre les partitions floues $\boldsymbol{c}$ et $\boldsymbol{t}$ comme la somme des divergences entre tous les n $\boldsymbol{c}_i$ et $\boldsymbol{t}_i$:

$$\mathrm{KL}(\boldsymbol{c}, \boldsymbol{t}) = \sum_{i=1}^{n} \mathrm{KL}(\boldsymbol{c}_i, \boldsymbol{t}_i) = \sum_{i=1}^{n}\sum_{k=1}^{K} \ln\left[\frac{c_{ik}}{t_{ik}}\right] c_{ik}.$$

Il est ensuite direct de montrer que

$$\begin{cases} \mathrm{LP}(\boldsymbol{t}, \boldsymbol{z}) = \mathrm{KL}(\boldsymbol{z}, \boldsymbol{t}) \\ \forall \boldsymbol{z} \text{ partition}, \quad \mathrm{E}(\boldsymbol{t}) = \mathrm{KL}(\boldsymbol{z}, \tilde{\boldsymbol{t}}) - \mathrm{KL}(\boldsymbol{t}, \tilde{\boldsymbol{t}}) = \text{cste} - \mathrm{KL}(\boldsymbol{t}, \tilde{\boldsymbol{t}}) \end{cases}$$

avec $\tilde{\boldsymbol{t}}$ la matrice de partition floue indiquant l'indécision dans le cas de K classes, c'est-à-dire

$$\tilde{\boldsymbol{t}} = \{\tilde{t}_{ik}\} \quad / \quad \forall\, 1 \leq i \leq n,\ \forall\, 1 \leq k \leq K, \quad \tilde{t}_{ik} = \frac{1}{K}.$$

On déduit donc que (voir une illustration sur la figure 3.8) :

- $\mathrm{LP}(\boldsymbol{t}, \boldsymbol{z})$ mesure un écart entre la partition floue $\boldsymbol{t}$ et la partition $\boldsymbol{z}$;
- $\mathrm{E}(\boldsymbol{t})$ mesure la différence entre deux écarts : l'ecart entre la partition floue d'indécision $\tilde{\boldsymbol{t}}$ et toute partition $\boldsymbol{z}$ (c'est un terme constant qui est en fait indépendant de $\boldsymbol{z}$) et l'écart entre la partition floue $\boldsymbol{t}$ et la partition floue d'indécision $\tilde{\boldsymbol{t}}$.

Utilisation des critères pour le modèle gaussien

Nous avons présenté les critères précédents pour le choix du nombre de classes car la plupart d'entre eux n'avaient été proposés que dans ce cas (PC, MIR, NEC). Néanmoins, ils sont facilement utilisables pour le choix du modèle gaussien. Notant $\hat{\boldsymbol{\theta}}_L$ l'estimateur du maximum de vraisemblance sous le modèle gaussien $\mathcal{M}$, on peut poser

$$\mathrm{E}(\mathcal{M}) = \mathrm{E}(\boldsymbol{t}(\hat{\boldsymbol{\theta}}_L)), \quad \mathrm{LP}(\mathcal{M}) = \mathrm{LP}(\boldsymbol{t}(\hat{\boldsymbol{\theta}}_L)), \quad \mathrm{PC}(\mathcal{M}) = \mathrm{PC}(\boldsymbol{t}(\hat{\boldsymbol{\theta}}_L)), \quad \mathrm{MIR}(\mathcal{M}) = \mathrm{MIR}(\hat{\boldsymbol{\theta}}_L).$$

Nous proposons aussi très empiriquement la version suivante de NEC pour trouver le modèle gaussien $\mathcal{M}$

$$\mathrm{NEC}(\mathcal{M}) = \frac{\mathrm{E}(\mathcal{M})}{\mathrm{L}(\mathcal{M}) - \mathrm{L}_1(\mathcal{M})}, \tag{3.18}$$

où $\mathrm{L}_1(\mathcal{M})$ est le maximum de log-vraisemblance pour une seule classe avec le modèle $\mathcal{M}$. L'étude de certains de ces critères pour trouver le modèle gaussien se fera en détail au chapitre suivant.

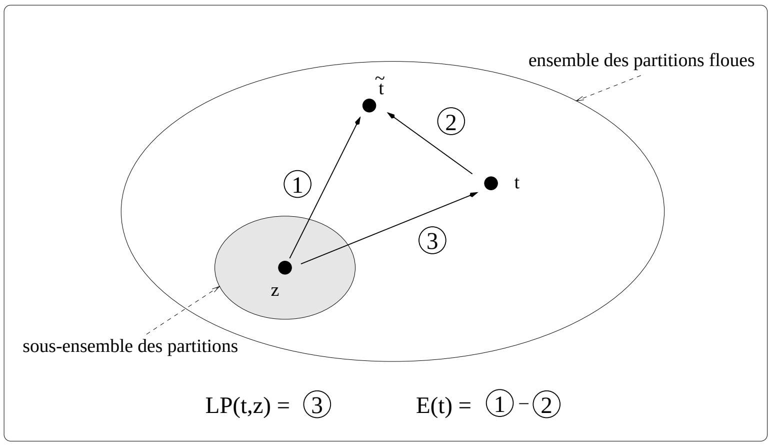


FIG. 3.8 – *Interprétation des critères E et LP au sens de la divergence de Kullback-Leibler.*

3.4.3 Pénalisation de la vraisemblance par la classifiablité

Principe

L'approche mélange et les critères classifiabilité ne nous contentent pas pleinement car les critères mélange ne tiennent pas compte de l'objectif de la classification automatique tandis que les critères classifiabilité favorisent excessivement l'aspect classification au détriment de l'aspect mélange (comme conséquence, on ne peut choisir entre une et plusieurs classes). Notre idée consiste à faire un *compromis* entre entre les deux points de vue de mélange et de classifiabilité. Nous proposons de traduire ce compromis sous la forme d'une pénalisation de la log-vraisemblance $\mathrm{L}(\mathcal{M})$ par un terme $g(\mathcal{M})$ mesurant la classifiabilité des données, c'est-à-dire de prendre un critère de la forme

$$\begin{array}{rcccc} \text{critère}(\mathcal{M}) & = & \mathrm{L}(\mathcal{M}) & - & g(\mathcal{M}) \\ & = & \left(\begin{array}{c} \scriptstyle\text{proximité} \\ \scriptstyle\text{modèle–données} \end{array} \right) & - & \left(\begin{array}{c} \scriptstyle\text{classifiabilité} \\ \scriptstyle\text{des données par le modèle} \end{array} \right) \end{array}$$

On retient le modèle $\mathcal{M}$ associé à la plus grande valeur du critère. À ce stade, deux stratégies d'utilisation de ce critère peuvent être envisagées :

- **Stratégie 1** : maximiser la vraisemblance, donc $\mathrm{L}(\mathcal{M})$ désigne le maximum de log-vraisemblance sous $\mathcal{M}$, et ensuite calculer $g(\mathcal{M})$ avec les estimateurs du maximum de vraisemblance pour enfin pénaliser $\mathrm{L}(\mathcal{M})$ et ainsi obtenir le critère ;

- **Stratégie 2** : maximiser le critère dans sa globalité pour chaque modèle $\mathcal{M}$. On optimise donc directement le compromis.

On remarque que la forme des critères proposés ressemble fortement à celle des critères de l'approche mélange qui pénalisent la complexité. Les deux stratégies précédentes étaient souvent équivalentes en mélange, car minimiser le critère AIC en globalité ou maximiser la vraisemblance pour ensuite calculer AIC étaient deux démarches strictement identiques (ce n'était pas le cas pour ICOMP). La différence avec les critères de l'approche mélange réside bien sûr dans le terme de pénalisation $g(\mathcal{M})$. Ce terme reste à définir, c'est pourquoi nous faisons deux propositions à son sujet : le terme d'entropie E et le terme LP. Comme on va le voir, ce choix est motivé par des interprétations intéressantes des critères qui en résultent. Il est aussi imaginable de pénaliser la vraisemblance par d'autres mesures de classifiablité telles que PC, MIR ou NEC. Ces derniers choix n'ont pas été retenus.

Pénalisation par l'entropie

La relation de Hathaway (2.6) énoncée au chapitre 2 s'écrit

$$\mathrm{C}(\boldsymbol{\theta}) = \mathrm{L}(\boldsymbol{\theta}) - \mathrm{E}(\boldsymbol{\theta})$$

et nous incite alors à prendre comme terme pénalisant $g(\boldsymbol{\theta}) = \mathrm{E}(\boldsymbol{\theta})$. Utilisant la stratégie 1 définie ci-dessus, nous obtenons le critère :

$$\mathrm{C}(\mathcal{M}) = \mathrm{L}(\mathcal{M}) - \mathrm{E}(\mathcal{M}) \tag{3.19}$$

où $\mathrm{L}(\mathcal{M}) = \mathrm{L}(\hat{\boldsymbol{\theta}}_L)$ maximum de log-vraisemblance avec le modèle $\mathcal{M}$ et $\mathrm{E}(\mathcal{M}) = \mathrm{E}(\boldsymbol{t}(\hat{\boldsymbol{\theta}}_L))$. Un exemple d'utilisation de ce critère est donné sur la figure 3.9 avec l'échantillon test de la figure 3.2 et le modèle $[p\lambda_k C_k]$. La deuxième stratégie (maximiser le critère $\mathrm{C}(\boldsymbol{\theta})$ globalement) n'a pas été étudiée pour le moment.

Pénalisation conduisant à la vraisemblance classifiante

Nous avons constaté une relation entre la log-vraisemblance et la log-vraisemblance classifiante par l'intermédiaire du terme LP (équation 3.17). Cette relation, démontrée par un calcul direct, s'énonce (Biernacki et Govaert [13] [14] [17])

$$\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{z}) = \mathrm{L}(\boldsymbol{\theta}) - \mathrm{LP}(\boldsymbol{\theta}, \boldsymbol{z})$$

où $\mathrm{LP}(\boldsymbol{\theta}, \boldsymbol{z}) = \mathrm{LP}(\boldsymbol{t}(\boldsymbol{\theta}), \boldsymbol{z})$. Cela nous incite à choisir la pénalisation $g(\boldsymbol{\theta}, \boldsymbol{z}) = \mathrm{LP}(\boldsymbol{\theta}, \boldsymbol{z})$. Les deux stratégies énoncées précédemment sont facilement envisageables :

- **Stratégie 1** : maximiser la vraisemblance, donc $\mathrm{L}(\mathcal{M}) = \mathrm{L}(\hat{\boldsymbol{\theta}}_L)$, puis calculer $\mathrm{LP}(\mathcal{M}) = \mathrm{LP}(\hat{\boldsymbol{\theta}}_L, \hat{\boldsymbol{z}}_L)$ avec $\hat{\boldsymbol{z}}_L$ partition obtenue par MAP de $\boldsymbol{t}(\hat{\boldsymbol{\theta}}_L)$. On a le critère de log-vraisemblance classifiante que nous appelons CLM (*Classification Likelihood in Mixture*) :

$$\mathrm{CLM}(\mathcal{M}) = \mathrm{L}(\mathcal{M}) - \mathrm{LP}(\mathcal{M}) \; ; \tag{3.20}$$

– **Stratégie 2** : maximiser directement $\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{z})$ sur $\boldsymbol{\theta}$ et $\boldsymbol{z}$. Cela conduit donc au maximum de vraisemblance classifiante

$$\mathrm{CL}(\mathcal{M}) = \mathrm{CL}(\hat{\boldsymbol{\theta}}_{CL}, \hat{\boldsymbol{z}}_{CL})$$

avec $\hat{\boldsymbol{\theta}}_{CL}$ et $\hat{\boldsymbol{z}}_{CL}$ estimateurs respectifs de $\boldsymbol{\theta}$ et $\boldsymbol{z}$ par maximum de vraisemblance classifiante.

Des exemples d'utilisation de ces deux critères sont donnés sur la figure 3.9 avec l'échantillon test de la figure 3.2 et le modèle $[p\lambda_k C_k]$.

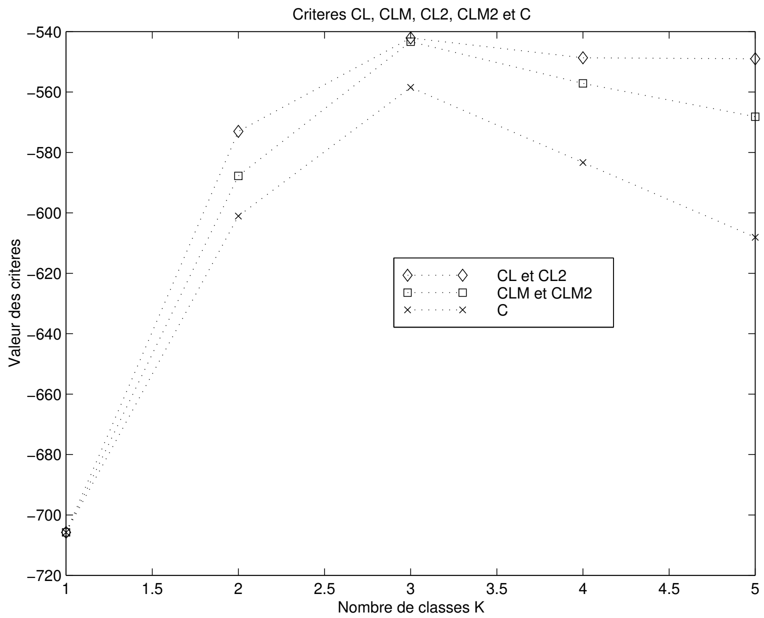


FIG. 3.9 – *Illustration du comportement des critères CL, CL2, CLM, CLM2 et C. On remarque l'influence des différentes pénalisations.*

Relation entre vraisemblance classifiante et critères classiques

Nous avions signalé à la section 2.3.1 du chapitre 2 que la maximisation du critère de vraisemblance classifiante était équivalente à l'optimisation de critères classiques de la classification automatique. Par exemple, la relation 2.18 du chapitre 2 montre que le maximum de log-vraisemblance classifiante correspond à une pénalisation d'une fonction du critère des k-means par une fonction du nombre de classes (ce terme était inclus dans la constante à l'équation 2.18). On a en effet

$$\mathrm{CL}_{[p\lambda I]}(K) = -\frac{nd}{2}\ln(\underbrace{\mathrm{tr}(\boldsymbol{W}(K))}_{\text{critère des } k\text{-means}}) - \underbrace{n\ln(K)}_{\alpha(K)} + \text{cste}$$

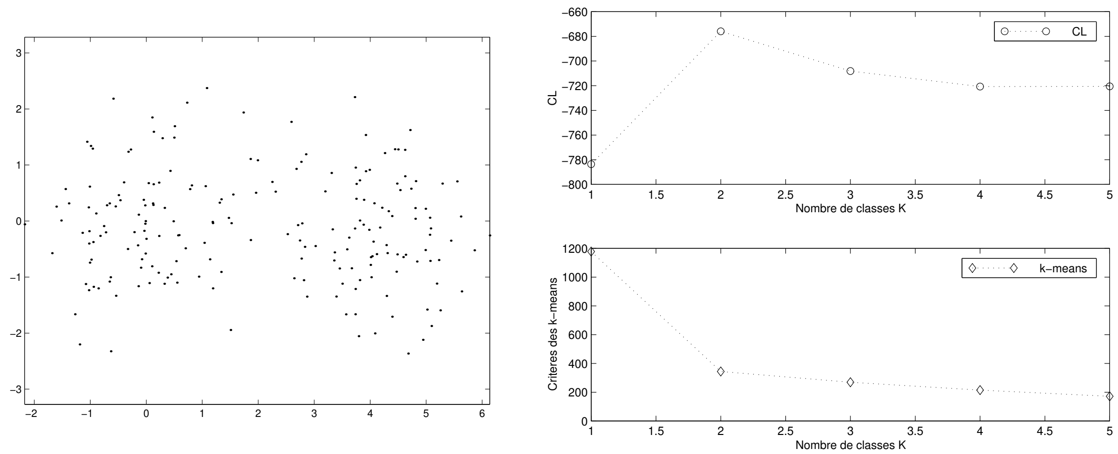
avec $\mathrm{CL}_{[p\lambda I]}(K) = \mathrm{CL}_{[p\lambda I]}(\hat{\boldsymbol{\theta}}_{CL}, \hat{\boldsymbol{z}}_{CL})$ maximum de log-vraisemblance classifiante pour le modèle $[p\lambda I]$ et K classes, $\boldsymbol{W}(K) = \boldsymbol{W}(\hat{\boldsymbol{z}}_{CL})$ et $\alpha(K)$ terme de pénalisation qui est fonction du nombre de classes.

Le critère des k-means est bien connu pour *systématiquement* surestimer le nombre de classes car il est toujours optimisé pour $K = n$ (on a en effet $\mathrm{tr}(\boldsymbol{W}(n)) = 0$). Hartigan [56] pp. 90-91 proposa alors une méthode empirique pour trouver un coude de ce critère. Il choisissait $K+1$ classes à la place de K classes si

$$\left[\frac{\mathrm{tr}(\boldsymbol{W}(K))}{\mathrm{tr}(\boldsymbol{W}(K+1))} - 1\right] \times (n - K - 1) \geq 10.$$

L'extension du critère des k-means apportée par la vraisemblance classifiante est intéressante car on peut supposer que la pénalisation $\alpha(K)$ sera suffisante pour enrayer la surestimation de ce critère (Biernacki et Govaert [17]). L'exemple de la figure 3.10 soutient cette hypothèse. On peut voir aussi le critère CL comme une extension pénalisée d'autres critères classiques en classification automatique (voir le tableau 2.1 du chapitre 2) :

$$\begin{aligned}
\mathrm{CL}_{[p\lambda C]}(K) &= -\frac{n}{2}\ln(|\boldsymbol{W}(K)|) - \alpha(K) + \mathrm{cste} \\
\mathrm{CL}_{[p\lambda_k C_k]}(K) &= -\frac{1}{2}\sum_{k=1}^{K} n_k \ln(|\boldsymbol{W}(K)|) - \alpha(K) + \mathrm{cste}.
\end{aligned}$$



«visuellement» deux classes sphériques — critère CL et critère des kmeans

FIG. 3.10 – *La vraisemblance classifiante avec le modèle $[p\lambda I]$ semble être une extension efficace du critère des k-means pour trouver le nombre de classes.*

Défaut prévisible de la vraisemblance classifiante

Malheureusement le critère de maximum de vraisemblance classifiante CL présente un défaut prévisible : si les espaces des modèles en compétition sont emboîtés, il choisit le plus complexe des modèles. Notant $\Theta(\mathcal{M})$ l'espace associé au modèle $\mathcal{M}$, on retrouve un défaut constaté lors de l'étude du maximum de vraisemblance

$$\Theta(\mathcal{M}) \subset \Theta(\mathcal{M}') \Rightarrow \mathrm{CL}(\mathcal{M}) \leq \mathrm{CL}(\mathcal{M}').$$

Cette propriété entraîne les conséquences suivantes :

- **modèle gaussien** : on ne peut choisir efficacement entre deux modèles gaussiens dont les espaces de paramètres sont inclus (par exemple $[p\lambda I]$ et $[p\lambda_k I]$) ;
- **Nombre de classes** :
  - on ne peut choisir efficacement entre deux nombres de classes dont les modèles gaussiens ont des *proportions libres* (par exemple $[p_k\lambda I]$) ;
  - **mais** rien n'empêche l'utilisation de CL pour sélectionner le nombre de classes lorsque les modèles gaussiens ont des *proportions égales* (par exemple $[p\lambda I]$).

Par contre le critère CLM ne souffre pas de cette croissance systématique, au moins théoriquement, car on ne maximise plus la vraisemblance classifiante mais on pénalise le maximum de vraisemblance par le terme LP.

Stratégies CL et CL2

Pour le nombre de classes, nous proposons deux stratégies *empiriques* pour résoudre le problème précédent : CL2 et CLM2. Elles correspondent respectivement aux critères CL et CLM mais rejettent systématiquement tous les nombres de classes où au moins une des classes à un cardinal inférieur au nombre total de paramètres libres. Cette méthode est discutable et on aurait pu prendre tout autre seuil de rejet de la partition. Par exemple, on aurait pu éliminer les nombres de classes dont une classe a une proportion inférieure à un certain seuil (par exemple, $p_k \leq \frac{0.1}{K}$). L'avantage de notre choix est d'avoir une méthode automatique (pas de seuil à fixer par l'utilisateur).

Relation entre les critères C, CL et CLM

On peut facilement établir des relations théoriques entre les critères C, CL et CLM. Tout d'abord, les critères C et CLM ne se différencient que par leur pénalisation $g(\mathcal{M})$. On a la relation d'inégalité

$$\mathrm{C}(\mathcal{M}) \leq \mathrm{CLM}(\mathcal{M}) \quad \text{car} \quad \mathrm{LP}(\mathcal{M}) \leq \mathrm{E}(\mathcal{M}).$$

Cette dernière inégalité peut se montrer en écrivant LP sous la forme $\mathrm{LP}(\mathcal{M}) = -\sum_{i=1}^{n}\sum_{k=1}^{K}\frac{1}{K}\ln(t_{i\nu_i})$ avec ν_i le numéro de la classe où le point $\boldsymbol{x}_i$ est affecté par la méthode du MAP. On a alors

$$\mathrm{E}(\mathcal{M}) - \mathrm{LP}(\mathcal{M}) = \sum_{i=1}^{n}\sum_{k=1}^{K}\underbrace{\Big(\frac{1}{K}\ln(t_{i\nu_i}) - t_{ik}\ln(t_{ik})\Big)}_{\geq 0} \geq 0$$

car $\forall\, 1 \leq i \leq n,\ \forall\, 1 \leq k \leq K,\ t_{i\nu_i} \geq \frac{1}{K}$ et $0 \leq t_{ik} \leq \frac{1}{K}$. On a aussi l'inégalité

$$\mathrm{CLM}(\mathcal{M}) \leq \mathrm{CL}(\mathcal{M})$$

car $\mathrm{CL}(\mathcal{M}) = \mathrm{CL}(\hat{\boldsymbol{\theta}}_{CL}, \hat{\boldsymbol{z}}_{CL}) = \max_{\theta,z}\mathrm{CL}(\boldsymbol{\theta}, \boldsymbol{z})$. Les valeurs relatives de ces critères se retrouvent sur l'exemple donné par la figure 3.9. On remarque aussi sur cette figure que la pénalisation apportée par le critère C semble être plus prononcée que celle produite par le critère CLM. On verra avec les expériences menées dans les deux chapitres suivants que, comme conséquence principale, C choisira des modèles plus simples que CLM. Ce dernier choisira également des modèles plus simples que CL.

Critère AWE

Remarquons que le critère AWE (équation 3.6), bien que critère de pénalisation de la complexité établi dans un contexte mélange, peut aussi être classé dans le groupe des critères pénalisant la vraisemblance par la classifiabilité. En effet, il utilise aussi le maximum de vraisemblance classifiante. Il est donc intéressant de constater que ce critère pénalise la vraisemblance par les deux types de pénalisation : la complexité du modèle *et* la classifiabilité que ce modèle induit sur les données.

3.5 Approches pour la discrimination

Cette section présente différents moyens pour choisir le modèle gaussien en discrimination. Tout d'abord, l'approche mélange peut généralement être employée car la discrimination gaussienne fait usage du modèle de mélange gaussien. Nous présentons aussi la méthode de la validation croisée, méthode s'inscrivant plus dans une «approche dicrimination» car tenant compte de l'objectif final de celle-ci : obtenir une règle de classement correcte. La validation croisée a l'inconvénient de nécessiter le calcul répétitif des règles de classement, c'est pourquoi nous présentons des formules de mise à jour de celles-ci pour certains modèles gaussiens.

3.5.1 Utilisation de l'approche mélange

La discrimination gaussienne s'appuie sur l'hypothèse de mélange gaussien, donc l'ensemble des méthodes décrites dans la section sur l'approche mélange peuvent être utilisées. Toutefois une différence importante existe entre les hypothèses faites dans le chapitre sur l'approche mélange et les hypothèses de la discrimination : les labels $\boldsymbol{z}$ sont maintenant supposés connus. L'utilisation des méthodes mélange mérite alors quelques justifications supplémentaires.

En discrimination, le rapport des vraisemblances s'écrit

$$r(\mathcal{X}, \boldsymbol{z}) = \frac{\max_{\theta \in \Theta_0} f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta})}{\max_{\theta \in \Theta} f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta})}.$$

La distribution asymptotique de $-2\ln(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n, Z_1, \ldots, Z_n)$ est encore un $\chi^2_{q-q_0}$ sous H_0, donc la théorie des tests d'hypothèses utilisée en mélange est encore valable.

La théorie bayésienne décrite pour l'approche mélange s'applique aussi dans le cas où le couple $(\mathcal{X}, \boldsymbol{z})$ est connu. Dans ce cas le facteur de Bayes s'écrit

$$B_0 = \frac{\mathrm{P}(\mathcal{X}, \boldsymbol{z}|\mathcal{M})}{\mathrm{P}(\mathcal{X}, \boldsymbol{z}|\mathcal{M}_0)}.$$

avec la vraisemblance intégée

$$\mathrm{P}(\mathcal{X}, \boldsymbol{z}|\mathcal{M}) = \int_{\Theta} f(\mathcal{X}, \boldsymbol{z}|\boldsymbol{\theta}, \mathcal{M})\pi(\boldsymbol{\theta}|\mathcal{M})d\boldsymbol{\theta}.$$

Le critère BIC est alors utilisable mais le terme $\mathrm{L}(\mathcal{M})$ qui le compose correspond maintenant au maximum de log-vraisemblance de la discrimination, c'est-à-dire la log-vraisemblance de l'équation 2.3 du chapitre 2.

Le critère AIC aussi est suffisamment général pour englober la recherche de modèles en discrimination. L'annexe B peut facilement s'étendre au cas des données avec labels. Dans cette situation les densités de $\boldsymbol{x}$ deviennent des densités jointes de $(\boldsymbol{x}, z)$ et les différentes espérances ne se font plus sur la variable $\boldsymbol{X}$ mais sur le couple $(\boldsymbol{X}, Z)$. Par exemple, le coût entropique B.1 s'écrit

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^*) = \sum_{k=1}^{K} \int \ln \left[\frac{f(\boldsymbol{x}, k|\boldsymbol{\theta}^*)}{f(\boldsymbol{x}, k|\boldsymbol{\theta})}\right] f(\boldsymbol{x}, k|\boldsymbol{\theta}^*) d\boldsymbol{x}.$$

Le critère AIC3 est lui aussi applicable en discrimination car il reprend le même principe que AIC. Remarquons néanmoins que sa justification théorique disparaît car il s'appuie sur les travaux de Wolfe, travaux uniquement applicables lors de la recherche du nombre de classes.

3.5.2 Critère de validation croisée

Principe de la validation croisée

Notons $\mathrm{P}_e(\mathcal{M})$ la probabilité d'erreur de classement $\mathrm{P}_e(u_{\hat{\theta}})$ (équation 1.7 du chapitre 1) avec $\hat{\boldsymbol{\theta}}$ estimateur de $\boldsymbol{\theta}$ sous les contraintes du modèle gaussien $\mathcal{M}$.

La règle $u_{\hat{\theta}}$ est, rappelons-le, la règle optimale associée au paramètre $\hat{\boldsymbol{\theta}}$. Si, pour tous les 28 modèles $\mathcal{M}$, $\mathrm{P}_e(\mathcal{M})$ était calculable, il suffirait alors de choisir le modèle minimisant cette erreur. Une telle approche prend donc en considération les problèmes de fond de la discrimination. Bien sûr, calculer $\mathrm{P}_e(\mathcal{M})$ nécessite la connaissance du paramètre $\boldsymbol{\theta}$ qui est inconnu. L'enjeu est alors d'estimer cette quantité $\mathrm{P}_e(\mathcal{M})$.

Reprenons pour cela l'expression de P_e donnée par l'équation 1.7. Utilisant la relation 1.6, on a

$$\begin{aligned}\mathrm{P}_e(\mathcal{M}) &= \int_{\mathbb{R}^d} \left(\sum_{k=1}^{K} \mathcal{L}(u_{\hat{\theta}}(\boldsymbol{x})|k) t_k(\boldsymbol{x}|\boldsymbol{\theta}) \right) f(\boldsymbol{x}|\boldsymbol{\theta}) d\boldsymbol{x} \\ &= \int_{\mathbb{R}^d} \sum_{k=1}^{K} \mathcal{L}(u_{\hat{\theta}}(\boldsymbol{x})|k) p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}) d\boldsymbol{x}.\end{aligned}$$

Cette quantité peut être estimée par une méthode de Monte-Carlo si l'on dispose d'un échantillon $(\mathcal{X}', \boldsymbol{z}')$ dont chaque élément $(\boldsymbol{x}'_i, z'_i)$ $(i = 1, \ldots, n')$ est une réalisation $i.i.d$ de la loi jointe $p_k\phi_k(\boldsymbol{x}|\boldsymbol{\theta})$ (rappelons que ce paramètre $\boldsymbol{\theta}$ est inconnu) :

$$\hat{\mathrm{P}}_e(\mathcal{M}) = \frac{1}{n'} \sum_{i=1}^{n'} \mathcal{L}(u_{\hat{\theta}}(\boldsymbol{x}'_i)|z'_i). \tag{3.21}$$

Il s'agit donc simplement du taux de nombre de points mal classés dans cet échantillon par $u_{\hat{\theta}}$. Ce taux correspond à l'erreur de classement de l'équation 1.11. Remarquons qu'on ne peut utiliser l'échantillon initial $(\mathcal{X}, \boldsymbol{z})$ pour estimer $\boldsymbol{\theta}$ et pour estimer *à la suite* $\mathrm{P}_e(\mathcal{M})$: cela produirait une estimation optimiste en moyenne car la règle de classement $u_{\hat{\theta}}$ est optimale pour l'échantillon $(\mathcal{X}, \boldsymbol{z})$ dont elle est issue. La quantité que l'on estimerait alors est souvent appelée *le taux d'erreur apparent*. Pour remédier à ce problème, une première technique, consiste à diviser l'ensemble $(\mathcal{X}, \boldsymbol{z})$ en deux sous-ensembles. Le premier sous-ensemble, dit *ensemble d'apprentissage*, sert à mettre au point le discriminateur $u_{\hat{\theta}}$, tandis que le second, *l'ensemble test*, est utilisé pour estimer $\mathrm{P}_e(\mathcal{M})$. Cette méthode nécessite un nombre de données suffisamment grand pour obtenir des résultats fiables.

La méthode de la validation croisée (ou «leave one out») fait un usage plus efficace de l'information disponible et ainsi permet d'être appliquée à de plus petits échantillons. Il est possible de recalculer par maximum de vraisemblance les paramètres $\hat{\boldsymbol{\theta}}^{(i)}$ du dicriminateur sur la base de l'échantillon complet $(\mathcal{X}, \boldsymbol{z})$ privé de l'individu $(\boldsymbol{x}_i, z_i)$ $(i = 1, \ldots, n)$. Ensuite la classe $u_{\hat{\theta}^{(i)}}(\boldsymbol{x}_i)$ peut être prédite. Si ce processus est réitéré pour toutes les n observations, $\mathrm{P}_e(\mathcal{M})$ est estimée par :

$$\mathrm{VC}(\mathcal{M}) = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(u_{\hat{\theta}^{(i)}}(\boldsymbol{x}_i)|z_i), \tag{3.22}$$

appelé critère de validation croisée. Stone [85] a montré que ce que critère était asymptotiquement équivalent au critère AIC (relation 3.8). L'inconvénient de cette seconde méthode est de nécessiter l'estimation de n discriminateurs $u_{\hat{\theta}(i)}$ ($i = 1, \ldots, n$), processus coûteux en temps de calcul. L'idée est alors d'établir des formules d'actualisation de $u_{\hat{\theta}(i)}$ à partir de $u_{\hat{\theta}}$, et ceci pour le plus grand nombre de modèles. L'établissement de toute règle $u_{\hat{\theta}}$ nécessite la connaissance des probabilités a posteriori $t_1(\boldsymbol{x}_i|\hat{\boldsymbol{\theta}}), \ldots, t_K(\boldsymbol{x}_i|\hat{\boldsymbol{\theta}})$, ou, de manière équivalente, des densités jointes $\hat{p}_1\phi_1(\boldsymbol{x}_i|\hat{\boldsymbol{a}}_1), \ldots, \hat{p}_K\phi_K(\boldsymbol{x}_i|\hat{\boldsymbol{a}}_K)$. Chacune de ces densités jointes s'exprime par les proportions $\hat{p}_k$, les centres $\hat{\boldsymbol{\mu}}_k$, les déterminants $|\hat{\boldsymbol{\Sigma}}_k|$ et inverses $\hat{\boldsymbol{\Sigma}}_k^{-1}$ des matrices de variance ($k = 1, \ldots, K$). Ainsi, seule l'actualisation de ces quantités suffira à mettre à jour la règle $u_{\hat{\theta}}$ en $u_{\hat{\theta}(i)}$. Le processus d'estimation de $\mathrm{P}_e(\mathcal{M})$ par validation croisée sera donc le suivant :

1. calculer $\hat{\boldsymbol{\theta}}$ par maximum de vraisemblance pour l'échantillon $(\mathcal{X}, \boldsymbol{z})$ complet sous la contrainte du modèle $\mathcal{M}$;

2. calculer alors $|\hat{\boldsymbol{\Sigma}}_k|$ et $\hat{\boldsymbol{\Sigma}}_k^{-1}$;

3. pour chacun des n échantillons distincts de taille $n-1$ et correspondant à $(\mathcal{X}, \boldsymbol{z})$ privé de $(\boldsymbol{x}_i, \boldsymbol{z}_i)$ ($i = 1, \ldots, n$), calculer $\hat{p}_k^{(i)}\phi_k(\boldsymbol{x}_i|\hat{\boldsymbol{a}}_k^{(i)})$ ($k = 1, \ldots, K$) par simple *actualisation* des $\hat{p}_k$, $\hat{\boldsymbol{\mu}}_k$, $|\hat{\boldsymbol{\Sigma}}_k|$ et $\hat{\boldsymbol{\Sigma}}_k^{-1}$, i.e. si possible sans recalcul par maximum de vraisemblance, ni extraction de déterminant ou d'inverse. On en déduit la classe $u_{\hat{\theta}(i)}$ de $\boldsymbol{x}_i$ par la méthode du MAP ;

4. calculer $\mathrm{VC}(\mathcal{M})$ par l'équation 3.22.

Pour établir les formules d'actualisation, nous nous plaçons dans un cadre légèrement plus général en supposant que soit on *ajoute* ($\epsilon = +1$), soit on *retire* ($\epsilon = -1$) le point $\boldsymbol{y}$ de la classe t ($1 \leq t \leq K$). Le point $\boldsymbol{y}$ est donc élément de l'ensemble $\{\mathcal{X}|z_i = t\}$. Dans ce contexte nous détaillons alors les formules d'actualisation des proportions $\hat{p}_k$, des centres $\hat{\boldsymbol{\mu}}_k$, des déterminants et inverses des matrices de variance $\hat{\boldsymbol{\Sigma}}_k$ de toutes les K classes (Celeux et al. [32], Biernacki et Govaert [16]). Remarquons par ailleurs que l'actualisation des centres et de l'inverse de $\hat{\boldsymbol{\Sigma}}_k$ a déjà été traitée pour le cas linéaire ou quadratique (voir Dubuisson [45] p. 174-178 par exemple).

Actualisation des proportions, des centres et des matrices d'inertie

Les formules d'actualisation des proportions dépendent des contraintes que l'on impose : proportions égales, proportions libres. Les formules d'actualisation des matrices d'inertie des classes $\boldsymbol{W}_k = \boldsymbol{W}_k(\boldsymbol{z}_k)$ (relation 2.5 avec $k = 1, \ldots, K$ et $\boldsymbol{z}_k$ k^{e} ligne de la matrice de partition $\boldsymbol{z}$), et de la matrice d'inertie intraclasse $\boldsymbol{W} = \boldsymbol{W}(\boldsymbol{z})$ (relation 2.4), ainsi que de leur inverse, de leur déterminant et de leur trace sont aussi à déterminer. Ces termes interviendront dans l'actualisation ultérieure des inverses et des déterminants des matrices de variance. Utilisant alors les résultats de l'annexe D, nous avons les formules consignées dans les tableaux 3.5 et 3.6.

Terme à actualiser	Formule d'actualisation	
	$k = t$	$k \neq t$
n_k	$n_t + \epsilon$	inchangé
$\hat{p}_k$ égales	inchangé	inchangé
$\hat{p}_k$ libres	$(n_t + \epsilon)/(n + \epsilon)$	$n_k/(n + \epsilon)$
$\hat{\boldsymbol{\mu}}_k$	$(n_t\hat{\boldsymbol{\mu}}_t + \epsilon\boldsymbol{y})/(n_t + \epsilon)$	inchangé
$\boldsymbol{W}_k$	$\boldsymbol{W}_t + \epsilon\boldsymbol{h}_t\boldsymbol{h}_t'$	inchangé
$\|\boldsymbol{W}_k\|$	$\|\boldsymbol{W}_t\|(1 + \epsilon\boldsymbol{h}_t'\boldsymbol{W}_t^{-1}\boldsymbol{h}_t)$	inchangé
$\boldsymbol{W}_k^{-1}$	$\boldsymbol{W}_t^{-1} - \{\epsilon(\boldsymbol{W}_t^{-1}\boldsymbol{h}_t)(\boldsymbol{W}_t^{-1}\boldsymbol{h}_t)'\}/(1 + \epsilon\boldsymbol{h}_t'\boldsymbol{W}_t^{-1}\boldsymbol{h}_t)$	inchangé
$\mathrm{tr}(\boldsymbol{W}_k)$	$\mathrm{tr}(\boldsymbol{W}_t) + \epsilon\boldsymbol{h}_t'\boldsymbol{h}_t$	inchangé

TAB. 3.5 – *Formules d'actualisation des populations, des proportions, des centres et des matrices d'inertie des classes ainsi que le déterminant, l'inverse et la trace de celles-ci. On ajoute ou on retire un point de la classe t. On a posé $\boldsymbol{h}_t = \omega_t(\boldsymbol{y} - \hat{\boldsymbol{\mu}}_t)$ et $\omega_t^2 = \frac{n_t}{n_t+\epsilon}$.*

Terme à actualiser	Formule d'actualisation
n	$n + \epsilon$
$\boldsymbol{W}$	$\boldsymbol{W} + \epsilon\boldsymbol{h}_t\boldsymbol{h}_t'$
$\|\boldsymbol{W}\|$	$\|\boldsymbol{W}\|(1 + \epsilon\boldsymbol{h}_t'\boldsymbol{W}^{-1}\boldsymbol{h}_t)$
$\boldsymbol{W}^{-1}$	$\boldsymbol{W}^{-1} - \{\epsilon(\boldsymbol{W}^{-1}\boldsymbol{h}_t)(\boldsymbol{W}^{-1}\boldsymbol{h}_t)'\}/(1 + \epsilon\boldsymbol{h}_t'\boldsymbol{W}^{-1}\boldsymbol{h}_t)$
$\mathrm{tr}(\boldsymbol{W})$	$\mathrm{tr}(\boldsymbol{W}) + \epsilon\boldsymbol{h}_t'\boldsymbol{h}_t$

TAB. 3.6 – *Formules d'actualisation de la taille de l'échantillon, de la matrice d'inertie intraclasse ainsi que du déterminant, de l'inverse et de la trace de celle-ci. On ajoute ou on retire un point de la classe t. On a posé $\boldsymbol{h}_t = \omega_t(\boldsymbol{y} - \hat{\boldsymbol{\mu}}_t)$ et $\omega_t^2 = \frac{n_t}{n_t+\epsilon}$.*

Actualisation du déterminant et de l'inverse des $\hat{\boldsymbol{\Sigma}}_k$

Le calcul du déterminant et de l'inverse des matrices de variance dépend du modèle considéré, parmi les 14 disponibles. Deux types de modèles existent : ceux permettant une forme explicite du calcul du déterminant et de l'inverse des matrices de variance, et ceux, au contraire, nécessitant l'emploi d'une procédure itérative (voir Celeux et Govaert [38]). Tous utilisent les matrices d'inertie des classes ou la matrice d'inertie intraclasse, voire leur déterminant, leur inverse ou leur trace.

Modèles avec calcul explicite Huit modèles appartiennent à cette catégorie : $[\lambda I]$, $[\lambda_k I]$, $[\lambda B]$, $[\lambda B_k]$, $[\lambda_k B_k]$, $[\lambda C]$, $[\lambda C_k]$ et $[\lambda_k C_k]$. Les formules de calcul sont disponibles dans le tableau 3.7 et proviennent de l'article de Celeux et Govaert [38].

Modèle	$\lvert\hat{\boldsymbol{\Sigma}}_k\rvert$		$\hat{\boldsymbol{\Sigma}}_k^{-1}$	
	$k = t$	$k \neq t$	$k = t$	$k \neq t$
$[\lambda_k I]$	$\left(\frac{\mathrm{tr}(\boldsymbol{W}_t)}{n_t d}\right)^d$	inchangé	$\frac{n_t d}{\mathrm{tr}(\boldsymbol{W}_t)}$	inchangé
$[\lambda_k B_k]$	$\frac{\lvert\mathrm{diag}(\boldsymbol{W}_t)\rvert}{n_t^d}$	inchangé	$n_t(\mathrm{diag}(\boldsymbol{W}_t))^{-1}$	inchangé
$[\lambda_k C_k]$	$\frac{\lvert\boldsymbol{W}_t\rvert}{n_t^d}$	inchangé	$n_t \boldsymbol{W}_t^{-1}$	inchangé
$[\lambda I]$	$\left(\frac{\mathrm{tr}(\boldsymbol{W})}{nd}\right)^d$		$\frac{nd}{\mathrm{tr}(\boldsymbol{W})}$	
$[\lambda B]$	$\frac{\lvert\mathrm{diag}(\boldsymbol{W})\rvert}{n^d}$		$n\,(\mathrm{diag}(\boldsymbol{W}))^{-1}$	
$[\lambda B_k]$	$\frac{(\sum_\ell \lvert\mathrm{diag}(\boldsymbol{W}_\ell)\rvert^{1/d})^d}{n^d}$		$n\left(\frac{\mathrm{diag}(\boldsymbol{W}_k)}{\sum_\ell \lvert\mathrm{diag}(\boldsymbol{W}_\ell)\rvert}\right)^{1/d}(\mathrm{diag}(\boldsymbol{W}_k))^{-1}$	
$[\lambda C]$	$\frac{\lvert\boldsymbol{W}\rvert}{n^d}$		$n\boldsymbol{W}^{-1}$	
$[\lambda C_k]$	$\left(\frac{\sum_\ell \lvert\boldsymbol{W}_\ell\rvert^{1/d}}{n^d}\right)^d$		$n\frac{\lvert\boldsymbol{W}_k\rvert^{1/d}}{\sum_\ell \lvert\boldsymbol{W}_\ell\rvert^{1/d}}\boldsymbol{W}_k^{-1}$	

TAB. 3.7 – *Formules de calcul du déterminant et de l'inverse des matrices de variance pour les huit modèles explicites. On s'appuiera sur les tableaux 3.5 et 3.6 pour accélérer le calcul. On ajoute ou on retire un point de la classe t.*

Modèles avec calcul itératif Les six modèles restant nécessitent une procédure itérative pour calculer $\hat{\boldsymbol{\Sigma}}_k$. Son déterminant et son inverse ne peuvent alors être actualisés sans un nouveau calcul de déterminant ou d'inverse.

Seul l'un d'entre eux, le modèle diagonal $[\lambda_k B]$, permet de calculer $\lvert\hat{\boldsymbol{\Sigma}}_k\rvert$ et $\hat{\boldsymbol{\Sigma}}_k^{-1}$ à partir de $\hat{\boldsymbol{\Sigma}}_k$ sans avoir à évaluer un déterminant ou une inverse trop coûteux en temps de calcul. Ceci est dû au caractère diagonal de la matrice de variance. Le calcul de $\lambda_k \boldsymbol{B}$ est donné dans Celeux et Govaert [38]. Partant d'une matrice $\boldsymbol{B}$ donnée, on itère les deux étapes suivantes jusqu'à convergence du critère $\sum_{k=1}^K \mathrm{tr}(\boldsymbol{W}_k \boldsymbol{B}^{-1}) + d\sum_{k-1}^K n_k \ln(\lambda_k)$:

$$\begin{aligned}
\lambda_k &= \frac{\boldsymbol{W}_k \boldsymbol{B}^{-1}}{dn_k} \quad (1 \leq k \leq K); \\
\boldsymbol{B} &= \frac{\mathrm{diag}\left(\sum_{k=1}^K \frac{1}{\lambda_k}\boldsymbol{W}_k\right)}{\left|\mathrm{diag}\left(\sum_{k=1}^K \frac{1}{\lambda_k}\boldsymbol{W}_k\right)\right|^{1/d}}.
\end{aligned}$$

Stratégie en situation d'égalité

Nous avons remarqué que plusieurs modèles peuvent fréquemment avoir la même erreur de validation croisée VC, en particulier lorsque la taille de l'ensemble d'apprentissage est faible. Suivant Bensmail et Celeux [8], nous choisissons alors deux stratégies d'utilisation du critère de validation croisée : VC^- et VC^+. Le critère VC^- choisit le modèle le plus simple en nombre de paramètres parmi tous les modèles à départager. En cas d'égalité du nombre de paramètres de surcroît, on retient de plus le modèle produisant le maximum de vraisemblance le plus fort. Au contraire de cette stratégie, VC^+ choisit le modèle qui fournit le maximum de vraisemblance le plus fort (c'est-à-dire en général le modèle le plus complexe) parmi ceux ayant même valeur du critère de validation croisée.

3.6 Conclusion

Ce chapitre nous a permis de rappeler bon nombre de méthodes et de critères classiques pour sélectionner une modèle. Certaines de ces approches ont une justification théorique bien établie, tandis que d'autres sont plus empiriques. Les points originaux qui ont été proposés peuvent se résumer ainsi :

- en mélange : introduction d'une notion de précision sur les données pour détecter un coude dans la courbe du maximum de vraisemblance. Cette méthode s'applique au choix du nombre de classes ;

- en classification automatique : proposition de critères pénalisant la vraisemblance par une mesure de la classifiabilité des données. Ces critères sont utilisables pour détecter le nombre de classes ainsi que pour trouver le modèle gaussien ;

- en discrimination : utilisation de la validation croisée pour retenir le modèle gaussien. Des formules d'actualisation pour accélérer le calcul accompagnent ce critère.

Les deux chapitres qui suivent ont pour objectif d'étudier le comportement de ces méthodes originales en comparant leurs performances à d'autres critères classiques que nous avons présentés :

- Le chapitre 4 étudie tout d'abord les caractéristiques d'un grand nombre de critères, dont ceux pénalisant la vraisemblance par la classifiabilité, quant au choix du modèle gaussien (le nombre de classes étant connu). Cette étude se fait en classification automatique. Puis, dans un contexte de discrimination, nous comparons de nombreux critères, dont celui de validation croisée, dans le cadre, de nouveau, de la détection du modèle gaussien ;

– le chapitre 5 compare de nombreux critères pour la détection du nombre de classes en classification (le modèle gaussien est connu). Nous verrons alors tout l'intérêt des critères pénalisant la vraisemblance par la complexité.

Chapitre 4

Sélection d'un modèle gaussien : étude comparative

4.1 Introduction

Ce chapitre propose de comparer certains des critères précédents pour la détection des modèles gaussiens. Cette comparaison a lieu dans deux situations distinctes : la classification automatique et la discrimination. De plus, nous adoptons une «optique classification» pour comparer les performances des critères : un critère est correct en classification automatique s'il choisit une bonne partition ; il est correct en discrimination s'il choisit une bonne règle de classement. Cela conduit à la définition d'un critère de performance relative pour comparer les critères en compétition. On constate alors la bonne tenue du critère AIC3 dans les deux cas, le critère de validation croisée étant aussi bien adapté en discrimination.

À la section 4.2 nous nous plaçons dans un cadre de classification automatique. Nous considérons les critères L, AIC, AIC3, BIC, ICOMP, la version modifiée de NEC, et les critères LP, E, CLM et C. Des simulations avec des données bivariées bidimensionnelles dont les structures sous-jacentes sont variées sont utilisées. La comparaison des critères ne s'effectue pas par le taux de sélection du «bon» modèle gaussien car nous préférons une mesure de performance directement dérivée du taux d'erreur de classement : un modèle est d'autant meilleur qu'il classe bien les données. Il ressort alors que la version retenue de NEC est inadaptée et qu'au contraire AIC3 est le meilleur des critères car il réalise un compromis plus efficace que AIC, BIC et ICOMP entre vraisemblance et complexité. Quant aux Les critères pénalisant la vraisemblance par une mesure de la classifiablilité, ils se révèlent assez médiocres. Ces comportements sont alors confirmés sur un jeu de données réelles.

Dans le cadre de la discrimination à la section 4.3, nous comparons les critères L, AIC, AIC3, BIC, AWE et NEC, ainsi que les critères de validation croisée VC^- et VC^+. Les simulations utilisent les mêmes structures de données qu'auparavant mais en se restreignant aux proportions égales et en considérant deux tailles d'échantillons : taille petite et taille moyenne. Utilisant encore une mesure de performance s'appuyant sur la notion de «classification» (le modèle le meilleur est celui qui minimise l'erreur de classement théorique), nous constatons une nette domination de VC^- pour de petits échantillons. Le critère AIC3 est le meilleur parmi les critères de pénalisation de la complexité. Ce critère AIC3 se distingue de nouveau avec une taille moyenne d'échantillon en devenant cette fois le critère le plus performant, mais dans ce cas l'ensemble des critères, dont ceux de validation croisée, sont assez bons. Nous concluons les essais en discrimination en générant des données provenant de classes non gaussiennes et en étudiant une situation en grande dimension. Il ressort que le critère de validation croisée VC^- semble le critère le plus robuste : ils résiste bien à un mauvais modèle et il a un bon comportement en grande dimension.

4.2 Expériences en classification automatique

Un grand nombre des critères présentés au chapitre 3 sont applicables au choix des 28 modèles gaussiens. Nous nous proposons d'étudier le comportement de certains de ces critères en classification automatique pour le choix de ces nouveaux modèles, ceci lorsque le nombre de classes est connu : les comportements que nous avions précédemment rapportés portaient essentiellement sur le choix du nombre de classes. Nous établissons ces résultats par le biais d'expériences numériques portant sur des données simulées et réelles (Biernacki et Govaert [15]).

Les critères que nous testons sont : AIC, AIC3, BIC, ICOMP, L, NEC, LP, E, CLM et C. Les critères AIC, AIC3, BIC, LP, E, CLM et C ne posent aucune difficulté de calcul. Le critère ICOMP est calculé par une méthode de Monte-Carlo présentée à la section 3.3.3 et utilise les résultats de l'annexe C. À notre connaissance, c'est le seul critère qui a déjà été utilisé pour choisir certains modèles gaussiens (Bozdogan [26]) : $[p\lambda I]$, $[p\lambda B]$, $[p\lambda C]$ et $[p\lambda_k C_k]$. Le critère L du maximum de log-vraisemblance est aussi considéré : tous les autres critères en sont finalement une extension et la comparaison au critère L permettra d'évaluer le gain que ces critères apportent. Enfin, le critère NEC que nous utilisons est la version modifiée (3.18) du critère NEC original.

4.2.1 Données simulées

Conditions d'expérience

Les jeux de données simulés sont bidimensionnels avec deux classes et 200 points. Ce nombre de classes correspond au plus petit nombre de classes qui permette d'avoir un mélange de distributions : nous pourrons donc utiliser des *contraintes interclasses* du genre $[p_k\lambda_k C_k]$. La dimension deux est la plus petite dimension faisant apparaître des *contraintes intraclasses* comme l'orientation ou la forme. Nous nous plaçons ainsi dans une situation très simple mais qui permet d'utiliser pleinement les caractéristiques des 28 modèles.

Nous définissons 14 couples différents de matrices de variance, chacun représentant une des 14 contraintes sur les matrices de variance. Les matrices $\boldsymbol{A}_k$ (matrice diagonale de terme a_k et $1/a_k$) et $\boldsymbol{D}_k$ (matrice de rotation d'angle α_k) sont définies explicitement à l'équation 1.13 du chapitre 1. Nous proposons aussi deux couples de valeurs pour les proportions p_1, p_2 pour représenter la contrainte des proportions égales ou libres. Le tableau 4.1 donne les valeurs des proportions p_k, des volumes λ_k, des formes a_k et des angles α_k que nous avons utilisées. Les valeurs choisies représentent ainsi chacune des contraintes des 28 modèles. Une vue globale des 14 couples de matrices de variance utilisés est disponible sur la figure 4.1.

proportions	
égales $[p]$	$p_1 = 0.5$, $p_2 = 0.5$
libres $[p_k]$	$p_1 = 0.3$, $p_2 = 0.7$

volumes	
égaux $[\lambda]$	$\lambda_1 = 1$, $\lambda_2 = 1$
libres $[\lambda_k]$	$\lambda_1 = 1$, $\lambda_2 = 3$

formes et orientations							
	$[I]$	$[B]$	$[B_k]$	$[C]$	$[C_k]$	$[DA_kD']$	$[D_kAD'_k]$
a_1	1	2	2	2	2	2	2
a_2	1	2	4	2	4	4	2
α_1	-	90	90	45	45	45	45
α_2	-	90	0	45	-45	45	-45

TAB. 4.1 – *Détail des proportions et des matrices de variance associées à chacun des 28 modèles. Les angles sont en degrés.*

Le centre de la première classe est fixé à $\boldsymbol{\mu}_1 = (0, 0)'$ et celui de la deuxième classe est défini par $\boldsymbol{\mu}_2 = (t, 0)'$. Pour chacune des 28 combinaisons proportions – matrices de variance, nous définissons 3 valeurs du paramètre t pour obtenir trois degrés de mélange : de l'ordre de 30 % d'erreur de classement $\mathrm{P}_e(u_\theta)$ (voir équation 1.7 du chapitre 1) pour un degré fort de mélange, 15 % pour un degré moyen et 5 % pour un degré faible. On peut voir un exemple des trois degrés de mélange sur la figure 4.2 (a)(b)(c). Une approximation de la valeur de t a été trouvée par la procédure dichotomique suivante :

- **initialisation** : pour des paramètres p_1, p_2, $\boldsymbol{\Sigma}_1$ et $\boldsymbol{\Sigma}_2$ donnés, pour un degré de mélange s voulu et pour $\varepsilon > 0$ «petit» donné, choisir un intervalle $[t^\circ_{inf}, t^\circ_{sup}]$ avec $t^\circ_{inf} = 0$ et t°_{sup} tel qu'il existe t dans cet intervalle permettant d'obtenir $\mathrm{P}_e(u_{\theta(t)}) = s$;

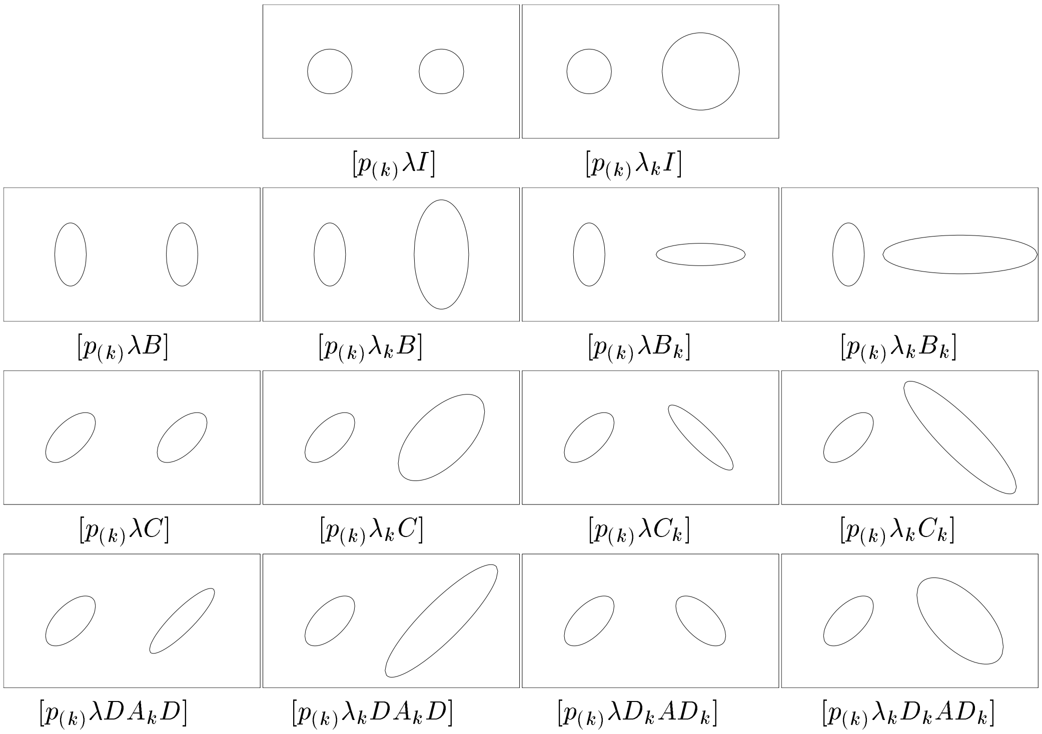


FIG. 4.1 – *Courbes isodensités des 14 couples de matrices de variance utilisés pour simuler les données. Chaque couple est associé à un modèle gaussien.*

- **itération** $(m > 0)$: estimer $\mathrm{P}_e(u_{\theta(t^m)})$ avec $t^m = \frac{t_{inf}^{m-1} + t_{sup}^{m-1}}{2}$ par $\hat{\mathrm{P}}_e(u_{\theta(t^m)})$ en générant un échantillon $(\mathcal{X}, \boldsymbol{z})$ de taille 10000 suivant la loi jointe $p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k(t^m))$ puis en utilisant la méthode de Monte-Carlo décrite en (3.21) au chapitre précédent. Ensuite
  - si $|\hat{\mathrm{P}}_e(u_{\theta(t^m)}) - s| \leq \varepsilon$, alors arrêt et $t \approx t^m$;
  - si $\hat{\mathrm{P}}_e(u_{\theta(t^m)}) \leq s$ alors $t_{inf}^m = t^m$;
  - si $\hat{\mathrm{P}}_e(u_{\theta(t^m)}) > s$ alors $t_{sup}^m = t^m$.

Certaines matrices de variance très différentes entre classes n'ont pu permettre un degré de mélange fort. C'est le cas de 11 couples proportions – matrices de variance (voir un exemple figure 4.2 (d)) associés aux modèles $[p\lambda B_k]$, $[p\lambda_k B_k]$, $[p\lambda C_k]$, $[p\lambda_k C_k]$, $[p\lambda_k D_k A D_k']$, $[p_k \lambda B_k]$, $[p_k \lambda_k B_k]$, $[p_k \lambda C_k]$, $[p_k \lambda_k C_k]$, $[p_k \lambda D_k A D_k']$, $[p_k \lambda_k D_k A D_k']$. Le nombre total de jeux de paramètres disponibles est alors de 17 modèles $\times$ 3 distances + 11 modèles $\times$ 2 distances = 73.

Pour chacun de ces 73 jeux de paramètres nous avons généré 30 échantillons de 200 points. Pour tous les échantillons, nous estimons les paramètres avec les 28 modèles gaussiens, ceci grâce à l'algorithme EM (section 2.2.2 du chapitre 2) initialisé avec les vrais paramètres. L'initialisation de EM avec des tirages

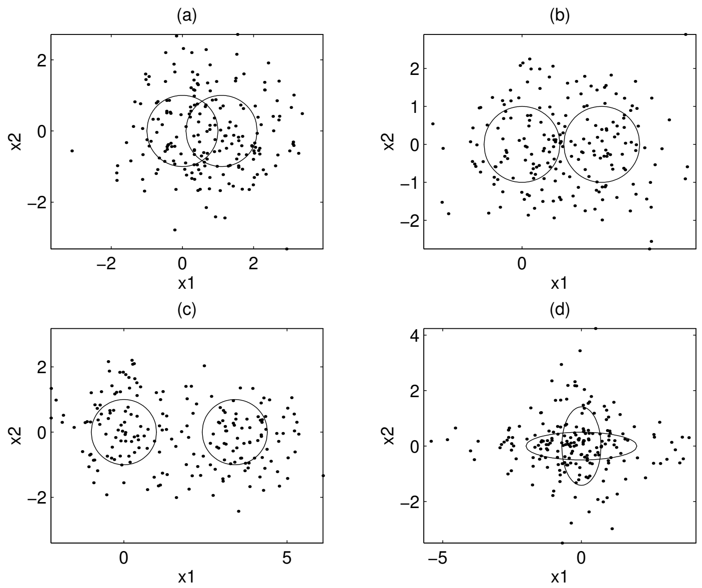


FIG. 4.2 – *(a), (b), (c) : un échantillon de* $[p\lambda I]$ *aux trois degrés de mélanges. (d) : avec un chevauchement maximal (centres égaux),* $[p\lambda B_k]$ *a une erreur théorique de classement n'excédant pas 22%. Les ellipses de variance tracées sont les courbes isodensité dont l'aire intérieure contient 70% de l'aire totale.*

aléatoires donne des résulats très proches si le nombre de ces tirages est assez grand. Notre choix permet alors de réduire le temps de calcul. Les critères L, AIC, AIC3, BIC, ICOMP, NEC, LP, E, CLM et C sont ensuite calculés.

Mesure de performance

Les résultats des simulations sont bien sûr très nombreux et il est difficile d'identifier les caractéristiques des critères. Nous proposons un outil de synthèse résumant la performance de chacun d'entre eux. Nous notons

$$\text{erreur}(\mathcal{M}) = \frac{\text{nombre de points mal classés par } \mathcal{M}}{n}$$

l'erreur de classement associée au choix du modèle $\mathcal{M}$ (équation 1.11 définie au chapitre 1). Cette erreur est une estimation de $\mathrm{P}_e(u_{\hat{\theta}})$ avec $\hat{\boldsymbol{\theta}}$ estimé sous contrainte de $\mathcal{M}$, estimation réalisée par l'équation 3.21. Remarquons que cet estimateur ne correspond pas au taux d'erreur apparent bien connu pour sous-estimer la vraie probabilité d'erreur $\mathrm{P}_e(u_{\hat{\theta}})$, car les labels $\boldsymbol{z}$ n'ont pas été utilisés pour estimer $\hat{\boldsymbol{\theta}}$.

À partir de ce critère, nous proposons une mesure de *performance relative* d'un critère $\mathcal{C}$ pour un échantillon par

$$\frac{\min(\text{erreur}(\mathcal{M}))}{\text{erreur}(\{\mathcal{M} : \arg(\mathcal{C}(\mathcal{M}))\})}.$$

Le numérateur correspond à l'erreur minimale de classement obtenue parmi tous les modèles $\mathcal{M}$ avec l'échantillon. Le dénominateur représente l'erreur de classement que l'on obtient avec le modèle $\mathcal{M}$ choisi au sens du critère $\mathcal{C}$. Cet indice est donc compris entre 0 et 1. Il indique, pour le critère étudié, une bonne performance lorsqu'on s'approche de 1 et mauvaise lorsqu'on s'en éloigne. Donc, via cette mesure, nous avons implicitement adopté l'erreur de classement comme le *critère de référence*. Nous ne jugeons alors pas un critère sur la détection du «bon» modèle sous-jacent aux données mais plutôt sur la capacité à choisir un modèle classifiant le plus correctement possible. Nous avons par ailleurs remarqué que ces deux notions de modèle le plus correctement classifiant et de modèle sous-jacent aux données se rejoignent : c'est souvent le modèle ayant vraiment généré l'échantillon qui minimise l'erreur de classement (ceci avait deja été constaté par Celeux et Govaert [38]).

Résultats

Sur chacun des 30×73 échantillons, nous estimons les valeurs des 10 critères. On peut voir un exemple sur le tableau 4.2 (la colonne «erreur» du tableau correspond à l'erreur empirique de classement commise par le modèle).

Puis nous en déduisons les performances relatives de chacun d'entre eux. Les moyennes et écart-types des performances relatives (calculées à partir des 2190 tableaux de même nature que le tableau 4.2) des critères L, AIC, AIC3, BIC, ICOMP, NEC, LP, E, CLM et C sont disponibles dans les tableaux 4.3 à 4.6 pour chacun des 28 modèles simulés avec les 3 (ou 2) degrés de mélange (la colonne «erreur minimale de référence» de ces tableaux correspond à l'erreur empirique minimale obtenue parmi tous les modèles). De ces tableaux, nous tirons des histogrammes de la performance moyenne de tous les critères pour nos 73 jeux de données (voir la figure 4.3).

La moyenne de la performance moyenne est un indicateur global de la performance de chaque critère. Trois critères sont alors en tête : AIC3 avec une valeur de 0.845, AIC avec 0.843 et BIC avec 0.841. L'écart-type de cette performance moyenne est assez similaire pour AIC3 et AIC et légèrement plus grand pour BIC qui peut avoir des performances moyennes près de 0.6. Ensuite arrive ICOMP avec une moyenne de 0.832 puis L avec 0.828. Les critères CLM (0.772), C (0.738), LP (0.629) et E (0.621) sont assez médiocres. La vraisemblance classifiante CLM néamnoins en tête. Les critères CLM et CL sont donc meilleurs que les critères classifiabilité LP et E utilisés seuls. NEC est très décevant en dernière place de ce classement avec 0.605 et un fort écart-type. Les critères NEC, LP et E sont les

Modèles	erreur	L	AIC	AIC3	BIC	ICOMP	NEC	LP	E	CLM	C
$[p\lambda I]$	0.270	-628.3	1266.7	1271.7	1283.2	1257.2	3.3	34.7	66.3	-663.1	-694.7
$[p\lambda_k I]$	0.270	-628.3	1268.6	1274.6	1288.4	1257.3	3.3	34.7	66.4	-663.1	-694.7
$[p\lambda B]$	0.340	-627.2	1266.5	1272.5	1286.3	1257.4	4.7	39.6	73.6	-666.9	-700.9
$[p\lambda_k B]$	0.340	-627.1	1268.3	1275.3	1291.4	1257.2	4.6	39.9	73.9	-667.1	-701.0
$[p\lambda B_k]$	0.330	-626.6	1267.3	1274.3	1290.4	1256.6	4.5	39.2	73.1	-665.9	-699.8
$[p\lambda_k B_k]$	0.330	-626.5	1269.1	1277.1	1295.5	1256.6	4.5	39.7	73.7	-666.3	-700.3
$[p\lambda C]$	0.145	-622.3	1258.6	1265.6	1281.7	1252.0	64.2	45.0	79.8	-667.3	-702.1
$[p\lambda_k C]$	0.200	-621.9	1259.9	1267.9	1286.3	1252.7	49.8	45.9	80.7	-667.9	-702.7
$[p\lambda C_k]$	0.140	-621.8	1261.6	1270.6	1291.3	1251.8	45.7	45.5	80.8	-667.3	-702.6
$[p\lambda_k C_k]$	0.250	-620.8	1261.6	1271.6	1294.6	1248.7	27.6	40.7	76.2	-661.5	-697.0
$[p\lambda DA_kD']$	0.235	-621.7	1259.5	1267.5	1285.9	1251.0	45.3	44.5	80.6	-666.3	-702.4
$[p\lambda_k DA_kD']$	0.250	-620.8	1259.6	1268.6	1289.3	1247.7	27.7	40.9	76.4	-661.7	-697.2
$[p\lambda D_kAD'_k]$	0.140	-621.8	1259.6	1267.6	1286.0	1251.2	46.1	46.1	81.5	-667.9	-703.3
$[p\lambda_k D_kAD'_k]$	0.150	-621.6	1261.3	1270.3	1291.0	1251.4	42.0	45.3	80.0	-667.0	-701.6
$[p_k\lambda I]$	0.280	-628.2	1268.4	1274.4	1288.1	1258.6	3.3	33.9	65.6	-662.1	-693.8
$[p_k\lambda_k I]$	0.275	-628.1	1270.3	1277.3	1293.4	1262.4	3.2	33.9	65.2	-662.1	-693.4
$[p_k\lambda B]$	0.375	-626.4	1266.9	1273.9	1290.0	1257.4	4.1	36.8	67.8	-663.2	-694.2
$[p_k\lambda_k B]$	0.410	-626.1	1268.3	1276.3	1294.6	1260.2	3.5	32.1	59.9	-658.2	-686.1
$[p_k\lambda B_k]$	0.345	-625.9	1267.9	1275.9	1294.3	1257.5	4.0	37.4	68.6	-663.4	-694.5
$[p_k\lambda_k B_k]$	0.355	-625.6	1269.2	1278.2	1298.9	1260.3	3.5	33.9	61.6	-659.6	-687.2
$[p_k\lambda C]$	0.140	-621.8	1259.7	1267.7	1286.1	1251.8	43.3	37.7	72.8	-659.6	-694.7
$[p_k\lambda_k C]$	0.140	-621.8	1261.7	1270.7	1291.4	1253.3	41.6	36.6	71.2	-658.4	-693.1
$[p_k\lambda C_k]$	0.200	-621.1	1262.2	1272.2	1295.2	1253.8	27.6	34.9	68.2	-656.0	-689.3
$[p_k\lambda_k C_k]$	0.235	-620.8	1263.6	1274.6	1299.9	1257.9	27.4	39.7	75.0	-660.6	-695.8
$[p_k\lambda DA_kD']$	0.205	-621.1	1260.2	1269.2	1289.9	1253.0	27.5	34.5	67.2	-655.6	-688.4
$[p_k\lambda_k DA_kD']$	0.215	-620.6	1261.3	1271.3	1294.3	1254.8	17.2	26.7	50.2	-647.4	-670.9
$[p_k\lambda D_kAD'_k]$	0.135	-621.6	1261.2	1270.2	1290.9	1254.1	39.6	41.1	76.5	-662.8	-698.1
$[p_k\lambda_k D_kAD'_k]$	0.135	-621.6	1263.2	1273.2	1296.2	1257.1	39.7	41.3	76.6	-663.0	-698.3

TAB. 4.2 – *Tableau des valeurs des critères pour un échantillon parmi les 73×30 échantillons. Sur cet exemple, le modèle simulé est $[p\lambda C]$ avec un degré moyen de mélange.*

Modèle simulé	Degré de mélange	Erreur min. de référence	Performance relative									
			L	AIC	AIC3	BIC	ICOMP	NEC	LP	E	CLM	C
$[p\lambda I]$	fort	0.275 (0.034)	0.688 (0.167)	0.803 (0.161)	0.815 (0.155)	0.896 (0.115)	0.801 (0.143)	0.673 (0.153)	0.665 (0.146)	0.665 (0.145)	0.670 (0.150)	0.667 (0.144)
	moyen	0.137 (0.022)	0.634 (0.214)	0.788 (0.220)	0.861 (0.150)	0.917 (0.050)	0.805 (0.219)	0.718 (0.218)	0.630 (0.224)	0.621 (0.223)	0.627 (0.218)	0.623 (0.221)
	faible	0.041 (0.017)	0.757 (0.186)	0.848 (0.164)	0.872 (0.147)	0.865 (0.154)	0.793 (0.193)	0.810 (0.155)	0.825 (0.171)	0.822 (0.153)	0.790 (0.159)	0.787 (0.156)
$[p\lambda_k I]$	fort	0.304 (0.036)	0.785 (0.118)	0.819 (0.129)	0.852 (0.114)	0.867 (0.122)	0.828 (0.124)	0.705 (0.099)	0.676 (0.090)	0.684 (0.088)	0.676 (0.089)	0.684 (0.086)
	moyen	0.163 (0.030)	0.823 (0.148)	0.879 (0.112)	0.907 (0.089)	0.942 (0.046)	0.847 (0.151)	0.571 (0.123)	0.543 (0.109)	0.528 (0.107)	0.586 (0.141)	0.537 (0.105)
	faible	0.048 (0.017)	0.820 (0.162)	0.863 (0.137)	0.867 (0.133)	0.888 (0.102)	0.851 (0.147)	0.837 (0.152)	0.577 (0.224)	0.579 (0.249)	0.841 (0.164)	0.815 (0.188)
$[p\lambda B]$	fort	0.278 (0.033)	0.716 (0.132)	0.749 (0.157)	0.794 (0.159)	0.823 (0.163)	0.757 (0.151)	0.598 (0.084)	0.669 (0.120)	0.673 (0.118)	0.687 (0.122)	0.681 (0.121)
	moyen	0.131 (0.021)	0.664 (0.238)	0.782 (0.219)	0.731 (0.246)	0.778 (0.244)	0.793 (0.217)	0.426 (0.218)	0.613 (0.257)	0.579 (0.241)	0.598 (0.236)	0.595 (0.239)
	faible	0.048 (0.016)	0.823 (0.133)	0.886 (0.107)	0.886 (0.102)	0.883 (0.096)	0.867 (0.103)	0.786 (0.223)	0.725 (0.296)	0.729 (0.293)	0.830 (0.129)	0.756 (0.240)
$[p\lambda_k B]$	fort	0.288 (0.034)	0.810 (0.130)	0.846 (0.113)	0.879 (0.125)	0.917 (0.101)	0.888 (0.098)	0.642 (0.092)	0.654 (0.091)	0.652 (0.091)	0.664 (0.093)	0.666 (0.090)
	moyen	0.143 (0.022)	0.834 (0.137)	0.861 (0.147)	0.900 (0.130)	0.920 (0.108)	0.912 (0.077)	0.512 (0.205)	0.488 (0.194)	0.490 (0.189)	0.574 (0.213)	0.507 (0.180)
	faible	0.048 (0.014)	0.915 (0.093)	0.916 (0.096)	0.916 (0.093)	0.920 (0.091)	0.907 (0.096)	0.855 (0.200)	0.540 (0.328)	0.527 (0.296)	0.920 (0.099)	0.876 (0.185)
$[p\lambda B_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.135 (0.031)	0.886 (0.098)	0.927 (0.059)	0.923 (0.066)	0.931 (0.066)	0.908 (0.073)	0.452 (0.084)	0.454 (0.087)	0.444 (0.080)	0.803 (0.194)	0.611 (0.209)
	faible	0.043 (0.016)	0.881 (0.130)	0.857 (0.148)	0.852 (0.158)	0.877 (0.153)	0.846 (0.163)	0.441 (0.224)	0.598 (0.254)	0.523 (0.255)	0.871 (0.160)	0.863 (0.149)
$[p\lambda_k B_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.133 (0.024)	0.935 (0.058)	0.930 (0.067)	0.946 (0.057)	0.943 (0.058)	0.937 (0.060)	0.431 (0.078)	0.436 (0.088)	0.429 (0.091)	0.836 (0.213)	0.534 (0.231)
	faible	0.053 (0.015)	0.911 (0.084)	0.922 (0.084)	0.933 (0.070)	0.921 (0.070)	0.930 (0.068)	0.461 (0.135)	0.636 (0.197)	0.586 (0.198)	0.925 (0.074)	0.861 (0.155)
$[p\lambda C]$	fort	0.310 (0.036)	0.767 (0.140)	0.793 (0.139)	0.832 (0.131)	0.908 (0.097)	0.811 (0.135)	0.744 (0.091)	0.734 (0.117)	0.733 (0.119)	0.747 (0.126)	0.745 (0.130)
	moyen	0.143 (0.032)	0.684 (0.205)	0.832 (0.221)	0.851 (0.189)	0.870 (0.161)	0.832 (0.202)	0.452 (0.133)	0.591 (0.224)	0.564 (0.208)	0.630 (0.226)	0.597 (0.216)
	faible	0.044 (0.012)	0.794 (0.186)	0.832 (0.169)	0.838 (0.142)	0.877 (0.130)	0.830 (0.160)	0.583 (0.317)	0.748 (0.246)	0.772 (0.253)	0.789 (0.171)	0.773 (0.222)

TAB. 4.3 – *Moyenne (et écart-type) de la performance relative des critères.*

Modèle simulé	Degré de mélange	Erreur min. de référence	Performance relative									
			L	AIC	AIC3	BIC	ICOMP	NEC	LP	E	CLM	C
$[p\lambda_k C]$	fort	0.314 (0.036)	0.789 (0.119)	0.801 (0.127)	0.843 (0.141)	0.846 (0.138)	0.847 (0.114)	0.702 (0.090)	0.696 (0.085)	0.696 (0.085)	0.706 (0.099)	0.705 (0.093)
	moyen	0.137 (0.025)	0.835 (0.128)	0.931 (0.082)	0.926 (0.083)	0.942 (0.065)	0.928 (0.111)	0.452 (0.112)	0.499 (0.140)	0.496 (0.143)	0.588 (0.176)	0.545 (0.137)
	faible	0.044 (0.014)	0.873 (0.128)	0.879 (0.120)	0.876 (0.117)	0.870 (0.121)	0.876 (0.118)	0.574 (0.335)	0.684 (0.251)	0.619 (0.263)	0.884 (0.104)	0.865 (0.132)
$[p\lambda C_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.136 (0.022)	0.908 (0.070)	0.928 (0.070)	0.931 (0.068)	0.936 (0.072)	0.909 (0.066)	0.478 (0.159)	0.424 (0.106)	0.421 (0.099)	0.838 (0.171)	0.757 (0.214)
	faible	0.038 (0.013)	0.839 (0.157)	0.886 (0.135)	0.872 (0.142)	0.892 (0.107)	0.836 (0.148)	0.746 (0.261)	0.427 (0.308)	0.438 (0.320)	0.850 (0.141)	0.837 (0.154)
$[p\lambda_k C_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.155 (0.023)	0.950 (0.068)	0.949 (0.074)	0.949 (0.074)	0.958 (0.043)	0.941 (0.073)	0.420 (0.060)	0.417 (0.066)	0.415 (0.068)	0.926 (0.118)	0.531 (0.225)
	faible	0.041 (0.015)	0.838 (0.180)	0.849 (0.173)	0.843 (0.183)	0.839 (0.181)	0.869 (0.180)	0.681 (0.333)	0.260 (0.236)	0.282 (0.264)	0.846 (0.181)	0.840 (0.180)
$[p\lambda D A_k D']$	fort	0.305 (0.033)	0.802 (0.124)	0.807 (0.134)	0.812 (0.132)	0.867 (0.119)	0.827 (0.124)	0.689 (0.086)	0.719 (0.091)	0.714 (0.088)	0.749 (0.125)	0.745 (0.125)
	moyen	0.154 (0.023)	0.817 (0.159)	0.827 (0.182)	0.880 (0.151)	0.890 (0.135)	0.861 (0.158)	0.430 (0.078)	0.517 (0.151)	0.534 (0.171)	0.660 (0.148)	0.614 (0.145)
	faible	0.053 (0.015)	0.830 (0.146)	0.893 (0.101)	0.902 (0.104)	0.908 (0.098)	0.895 (0.124)	0.421 (0.343)	0.597 (0.350)	0.564 (0.329)	0.858 (0.140)	0.838 (0.180)
$[p\lambda_k D A_k D']$	fort	0.288 (0.034)	0.824 (0.111)	0.830 (0.131)	0.833 (0.135)	0.832 (0.146)	0.861 (0.131)	0.706 (0.086)	0.688 (0.094)	0.692 (0.092)	0.683 (0.103)	0.682 (0.094)
	moyen	0.156 (0.024)	0.899 (0.103)	0.891 (0.139)	0.893 (0.140)	0.918 (0.090)	0.891 (0.138)	0.448 (0.103)	0.448 (0.104)	0.446 (0.102)	0.575 (0.195)	0.529 (0.163)
	faible	0.040 (0.011)	0.851 (0.145)	0.839 (0.154)	0.859 (0.134)	0.866 (0.127)	0.853 (0.132)	0.263 (0.278)	0.308 (0.280)	0.273 (0.268)	0.859 (0.149)	0.828 (0.166)
$[p\lambda D_k A D'_k]$	fort	0.292 (0.026)	0.873 (0.105)	0.888 (0.103)	0.899 (0.097)	0.871 (0.145)	0.923 (0.056)	0.718 (0.148)	0.644 (0.091)	0.649 (0.094)	0.655 (0.102)	0.665 (0.109)
	moyen	0.152 (0.023)	0.869 (0.119)	0.884 (0.115)	0.889 (0.114)	0.907 (0.082)	0.886 (0.106)	0.602 (0.221)	0.477 (0.138)	0.463 (0.114)	0.685 (0.205)	0.532 (0.178)
	faible	0.042 (0.016)	0.786 (0.138)	0.784 (0.158)	0.777 (0.168)	0.783 (0.150)	0.791 (0.164)	0.734 (0.157)	0.583 (0.289)	0.589 (0.289)	0.759 (0.138)	0.766 (0.140)
$[p\lambda_k D_k A D'_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.149 (0.028)	0.890 (0.095)	0.926 (0.078)	0.945 (0.067)	0.946 (0.068)	0.930 (0.087)	0.516 (0.169)	0.456 (0.104)	0.455 (0.107)	0.651 (0.232)	0.516 (0.169)
	faible	0.042 (0.011)	0.844 (0.113)	0.881 (0.135)	0.896 (0.122)	0.896 (0.122)	0.868 (0.135)	0.651 (0.288)	0.487 (0.290)	0.506 (0.306)	0.835 (0.164)	0.778 (0.243)

TAB. 4.4 – *Moyenne (et écart-type) de la performance relative des critères (suite).*

Modèle simulé	Degré de mélange	Erreur min. de référence	Performance relative									
			L	AIC	AIC3	BIC	ICOMP	NEC	LP	E	CLM	C
$[p_k\lambda I]$	fort	0.289 (0.033)	0.813 (0.134)	0.781 (0.135)	0.747 (0.138)	0.706 (0.113)	0.695 (0.130)	0.847 (0.128)	0.880 (0.107)	0.877 (0.108)	0.881 (0.108)	0.877 (0.109)
	moyen	0.144 (0.028)	0.650 (0.216)	0.731 (0.212)	0.766 (0.186)	0.738 (0.139)	0.664 (0.161)	0.843 (0.187)	0.794 (0.169)	0.770 (0.193)	0.760 (0.203)	0.752 (0.200)
	faible	0.037 (0.012)	0.752 (0.206)	0.750 (0.212)	0.801 (0.180)	0.833 (0.152)	0.727 (0.198)	0.822 (0.156)	0.790 (0.175)	0.799 (0.167)	0.759 (0.167)	0.777 (0.170)
$[p_k\lambda_k I]$	fort	0.292 (0.031)	0.827 (0.129)	0.775 (0.149)	0.752 (0.142)	0.696 (0.126)	0.734 (0.142)	0.834 (0.116)	0.841 (0.104)	0.874 (0.104)	0.845 (0.111)	0.878 (0.104)
	moyen	0.147 (0.028)	0.830 (0.172)	0.824 (0.176)	0.783 (0.147)	0.781 (0.141)	0.744 (0.169)	0.778 (0.244)	0.567 (0.241)	0.571 (0.247)	0.615 (0.255)	0.622 (0.265)
	faible	0.043 (0.012)	0.844 (0.163)	0.860 (0.157)	0.906 (0.098)	0.919 (0.091)	0.802 (0.196)	0.911 (0.094)	0.831 (0.149)	0.834 (0.140)	0.844 (0.170)	0.859 (0.127)
$[p_k\lambda B]$	fort	0.298 (0.032)	0.818 (0.133)	0.808 (0.133)	0.770 (0.132)	0.750 (0.123)	0.739 (0.121)	0.720 (0.152)	0.878 (0.146)	0.880 (0.142)	0.891 (0.136)	0.903 (0.127)
	moyen	0.160 (0.023)	0.698 (0.215)	0.684 (0.210)	0.687 (0.221)	0.614 (0.226)	0.641 (0.197)	0.540 (0.212)	0.703 (0.222)	0.754 (0.211)	0.746 (0.209)	0.736 (0.215)
	faible	0.043 (0.014)	0.757 (0.202)	0.814 (0.214)	0.798 (0.236)	0.822 (0.210)	0.796 (0.204)	0.800 (0.187)	0.730 (0.275)	0.758 (0.256)	0.841 (0.122)	0.792 (0.185)
$[p_k\lambda_k B]$	fort	0.281 (0.033)	0.849 (0.107)	0.794 (0.166)	0.765 (0.167)	0.718 (0.163)	0.765 (0.177)	0.637 (0.114)	0.778 (0.144)	0.792 (0.151)	0.842 (0.102)	0.841 (0.116)
	moyen	0.168 (0.035)	0.844 (0.137)	0.807 (0.187)	0.775 (0.188)	0.736 (0.181)	0.778 (0.177)	0.562 (0.268)	0.575 (0.248)	0.570 (0.227)	0.644 (0.252)	0.657 (0.245)
	faible	0.058 (0.015)	0.872 (0.129)	0.912 (0.128)	0.904 (0.130)	0.869 (0.160)	0.916 (0.124)	0.851 (0.189)	0.591 (0.332)	0.441 (0.280)	0.864 (0.137)	0.835 (0.182)
$[p_k\lambda B_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.148 (0.024)	0.911 (0.102)	0.919 (0.103)	0.915 (0.108)	0.894 (0.139)	0.889 (0.135)	0.444 (0.153)	0.545 (0.102)	0.537 (0.111)	0.710 (0.182)	0.615 (0.156)
	faible	0.043 (0.017)	0.879 (0.117)	0.876 (0.148)	0.857 (0.150)	0.852 (0.151)	0.875 (0.111)	0.489 (0.304)	0.818 (0.186)	0.823 (0.180)	0.851 (0.120)	0.842 (0.139)
$[p_k\lambda_k B_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.150 (0.022)	0.900 (0.101)	0.897 (0.115)	0.895 (0.123)	0.881 (0.154)	0.900 (0.120)	0.382 (0.074)	0.445 (0.099)	0.442 (0.100)	0.624 (0.212)	0.603 (0.208)
	faible	0.050 (0.016)	0.916 (0.080)	0.928 (0.094)	0.917 (0.100)	0.923 (0.102)	0.928 (0.094)	0.364 (0.148)	0.750 (0.225)	0.530 (0.258)	0.907 (0.091)	0.883 (0.136)
$[p_k\lambda C]$	fort	0.303 (0.030)	0.704 (0.090)	0.704 (0.090)	0.704 (0.090)	0.704 (0.090)	0.710 (0.106)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)
	moyen	0.171 (0.030)	0.696 (0.216)	0.756 (0.204)	0.749 (0.198)	0.673 (0.186)	0.684 (0.183)	0.539 (0.175)	0.722 (0.195)	0.718 (0.207)	0.762 (0.187)	0.737 (0.195)
	faible	0.059 (0.016)	0.790 (0.214)	0.805 (0.222)	0.822 (0.223)	0.887 (0.157)	0.696 (0.250)	0.539 (0.310)	0.819 (0.200)	0.832 (0.208)	0.847 (0.162)	0.832 (0.208)

TAB. 4.5 – *Moyenne (et écart-type) de la performance relative des critères (suite).*

Modèle simulé	Degré de mélange	Erreur min. de référence	Performance relative									
			L	AIC	AIC3	BIC	ICOMP	NEC	LP	E	CLM	C
$[p_k\lambda_k C]$	fort	0.280 (0.030)	0.798 (0.138)	0.758 (0.153)	0.737 (0.149)	0.678 (0.123)	0.723 (0.162)	0.641 (0.098)	0.782 (0.139)	0.774 (0.149)	0.820 (0.120)	0.830 (0.135)
	moyen	0.167 (0.036)	0.898 (0.123)	0.804 (0.199)	0.773 (0.202)	0.738 (0.185)	0.786 (0.178)	0.435 (0.125)	0.461 (0.148)	0.453 (0.145)	0.602 (0.237)	0.527 (0.192)
	faible	0.052 (0.015)	0.882 (0.144)	0.873 (0.185)	0.884 (0.185)	0.817 (0.242)	0.869 (0.181)	0.608 (0.371)	0.798 (0.230)	0.758 (0.255)	0.881 (0.159)	0.888 (0.147)
$[p_k\lambda C_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.151 (0.032)	0.941 (0.065)	0.947 (0.059)	0.928 (0.092)	0.907 (0.116)	0.936 (0.064)	0.608 (0.142)	0.644 (0.141)	0.641 (0.143)	0.717 (0.163)	0.665 (0.134)
	faible	0.057 (0.014)	0.895 (0.119)	0.926 (0.115)	0.969 (0.054)	0.950 (0.083)	0.909 (0.122)	0.540 (0.205)	0.512 (0.184)	0.484 (0.168)	0.896 (0.128)	0.903 (0.118)
$[p_k\lambda_k C_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.141 (0.027)	0.920 (0.075)	0.907 (0.099)	0.909 (0.095)	0.911 (0.092)	0.914 (0.090)	0.411 (0.185)	0.626 (0.153)	0.625 (0.152)	0.753 (0.165)	0.683 (0.141)
	faible	0.062 (0.017)	0.933 (0.083)	0.934 (0.082)	0.931 (0.084)	0.882 (0.148)	0.933 (0.084)	0.467 (0.343)	0.466 (0.214)	0.466 (0.214)	0.946 (0.082)	0.919 (0.114)
$[p_k\lambda DA_kD']$	fort	0.289 (0.040)	0.819 (0.141)	0.757 (0.146)	0.723 (0.133)	0.653 (0.110)	0.737 (0.152)	0.618 (0.084)	0.742 (0.164)	0.743 (0.168)	0.849 (0.121)	0.838 (0.133)
	moyen	0.146 (0.038)	0.810 (0.172)	0.787 (0.174)	0.779 (0.174)	0.755 (0.200)	0.710 (0.167)	0.451 (0.224)	0.767 (0.178)	0.784 (0.162)	0.821 (0.121)	0.818 (0.118)
	faible	0.047 (0.014)	0.731 (0.220)	0.694 (0.275)	0.692 (0.278)	0.691 (0.280)	0.677 (0.288)	0.563 (0.297)	0.672 (0.208)	0.648 (0.203)	0.717 (0.190)	0.670 (0.161)
$[p_k\lambda_k DA_kD']$	fort	0.274 (0.026)	0.822 (0.131)	0.775 (0.146)	0.754 (0.150)	0.701 (0.138)	0.799 (0.151)	0.611 (0.104)	0.741 (0.168)	0.748 (0.167)	0.847 (0.125)	0.868 (0.112)
	moyen	0.171 (0.033)	0.833 (0.194)	0.801 (0.174)	0.756 (0.174)	0.689 (0.184)	0.809 (0.173)	0.402 (0.144)	0.515 (0.214)	0.529 (0.219)	0.657 (0.233)	0.683 (0.222)
	faible	0.049 (0.015)	0.845 (0.173)	0.847 (0.177)	0.812 (0.201)	0.778 (0.228)	0.839 (0.179)	0.279 (0.272)	0.525 (0.329)	0.550 (0.328)	0.836 (0.163)	0.827 (0.165)
$[p_k\lambda D_kAD'_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.158 (0.023)	0.899 (0.104)	0.907 (0.106)	0.884 (0.134)	0.876 (0.150)	0.853 (0.140)	0.702 (0.114)	0.698 (0.115)	0.698 (0.115)	0.710 (0.117)	0.706 (0.114)
	faible	0.043 (0.016)	0.862 (0.143)	0.841 (0.163)	0.838 (0.162)	0.845 (0.141)	0.864 (0.153)	0.671 (0.279)	0.511 (0.248)	0.487 (0.222)	0.837 (0.179)	0.794 (0.235)
$[p_k\lambda_k D_kAD'_k]$	fort	-	-	-	-	-	-	-	-	-	-	-
	moyen	0.170 (0.027)	0.912 (0.089)	0.900 (0.099)	0.864 (0.145)	0.765 (0.210)	0.898 (0.120)	0.488 (0.180)	0.585 (0.214)	0.626 (0.192)	0.648 (0.218)	0.613 (0.199)
	faible	0.054 (0.020)	0.887 (0.180)	0.882 (0.185)	0.882 (0.185)	0.836 (0.241)	0.880 (0.178)	0.866 (0.202)	0.646 (0.250)	0.636 (0.248)	0.885 (0.181)	0.866 (0.202)

TAB. 4.6 – *Moyenne (et écart-type) de la performance relative des critères (suite et fin).*

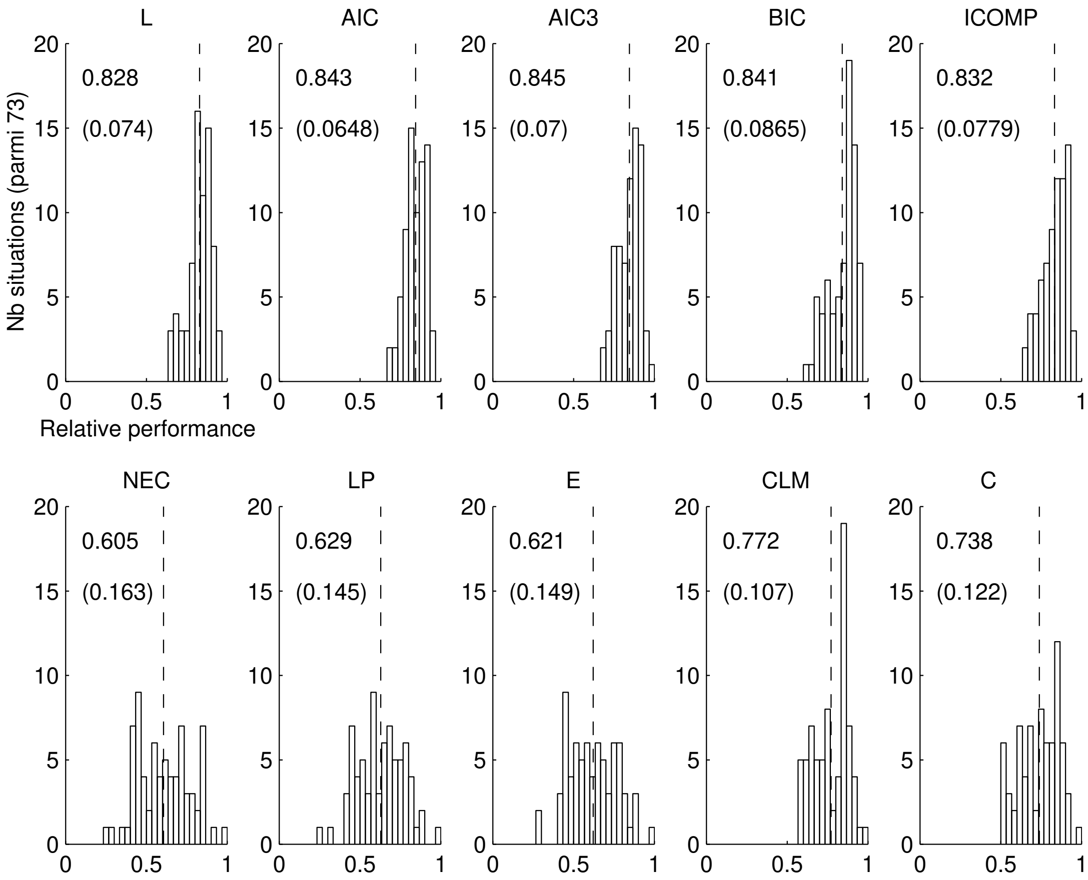


FIG. 4.3 – *Histogramme de la performance relative moyenne des 73 situations pour les critères. La moyenne de cette performance moyenne est tracée en tirets, sa valeur est marquée en haut. Son écart-type est entre parenthèses.*

seuls à passer sous la barre des 0.5 et ils ne semblent donc pas du tout adaptés à la détection du modèle gaussien.

erreur	L	AIC	AIC3	BIC	ICOMP	NEC	LP	E	CLM	C
8.60	10.72	8.40	8.04	7.67	9.08	7.95	8.46	8.50	9.67	9.47

TAB. 4.7 – *Nombre moyen de paramètres des modèles choisis par les critères. Le nombre moyen des paramètres des modèles sous-jacents est égal à 8.14.*

Pour appréhender de manière concise les types de modèles choisis par les critères, le tableau 4.7 répertorie le nombre moyen de paramètres que ces critères choisissent. Comme on pouvait s'y attendre, le critère AIC3 sélectionne des modèles plus complexes que BIC mais plus simples que AIC. Le critère AIC3 ayant une performance supérieure à ces deux critères, il réalise donc un compromis plus efficace entre la vraisemblance et la complexité du modèle. ICOMP produit un compromis encore moins efficace que AIC et BIC car il retient systématiquement des modèles trop complexes. Il choisit néanmoins des modèles plus simples que ne le fait le critère L de maximum log-vraisemblance. À ce propos, L devrait théoriquement toujours choisir le modèle le plus complexe $[p_k \lambda_k C_k]$ donc la moyenne des degrés de liberté devrait être égale à 11. La valeur obtenue n'est que de 10.72 car, en pratique, on n'a pas toujours obtenu le maximum de la vraisemblance. Les critères CLM et C choisissent des modèles relativement complexes (respectivement 9.67 et 9.47 degrés de liberté moyens). Le critère C semble donc plus pénalisant que le critère CLM, propriété que nous avions déjà suggérée au chapitre précédent. Les critères classification LP et E choisissent des modèles plus simples que les deux critères qui pénalisent la vraisemblance par la classifiabilité mais ils sont nettement moins performants (0.772 et 0.738 respectivement). Le critère LP est meilleur que le critère d'entropie ce qui explique les meilleurs résultats de CLM par rapport à C. Le critère NEC, quant à lui, choisit des modèles très simples (7.95) et bien en dessous du nombre moyen de degrés de libertés des vrais modèles sous-jacents (8.14). Le critère d'erreur, critère de référence, choisit au contraire des modèles plus complexes (en moyenne) que les vrais modèles sous-jacents. On pourrait alors se demander si ce critère d'erreur n'introduit pas alors un léger biais pour mesurer les performances.

4.2.2 Données réelles

Les données

Les simulations précédentes ont permis de se faire une idée du comportement des critères étudiés dans certaines situations typiques de mélange gaussien. On se propose de valider ce travail de simulation par une expérience réalisée sur un jeu de données bidimensionnelles réelles d'hémophilie provenant de Habbema et al.

[55]. Elles sont composées de 75 points et représentent des personnes réparties en deux groupes : celui des femmes normales d'effectif $n_1 = 30$ et celui des femmes hémophiles A d'effectif $n_2 = 45$. Un individu est défini par deux variables quantitatives $\boldsymbol{x}^1 = \log_{10}(\text{activité AHF})$ et $\boldsymbol{x}^2 = \log_{10}(\text{antigène AHF})$. Ces données sont représentées sur la figure 4.4(a) et dans le tableau E.1 de l'annexe E.

Les deux classes de données fournissent un exemple de distributions approximativement gaussiennes comme le montre l'analyse de McLachlan et Basford [70]. Ces mêmes auteurs ont déjà utilisé ces données pour montrer la sensibilité de EM au choix d'un modèle approprié ou non.

Conditions d'expérience et résultats

Connaissant l'appartenance des points, on peut estimer les «vrais» paramètres du mélange. Ils sont répertoriés dans le tableau 4.8. L'erreur de classement obtenue avec ces estimateurs est égale à 0.12, donc correspond à une imbrication d'ordre moyen suivant notre échelle de mélange.

classe	$\hat{p}_k$	$\hat{\boldsymbol{\mu}}_k$	$\hat{\lambda}_k$	$\hat{a}_k$	$\hat{\alpha}_k$
$k=1$	0.4	$(-0.135\ -0.078)'$	0.011	3.023	42.29
$k=2$	0.6	$(-0.308\ -0.006)'$	0.018	2.145	45.23

TAB. 4.8 – *Estimation des «vrais» paramètres du mélange connaissant l'origine des points. L'erreur de classement obtenue est égale à 0.12 donc correspond à un mélange plutôt moyen.*

On lance EM initialisé au hasard 20 fois et on ne garde que l'essai fournissant la vraisemblance maximale. On utilise tour à tour chacun des 28 modèles gaussiens. Les valeurs de chaque critère sont rassemblées dans le tableau 4.9. Il ressort que l'erreur minimale est égale à 0.15 avec le modèle $[p_k\lambda_k D_k A D_k']$. Aucun critère ne détecte ce modèle mais AIC et AIC3 choisissent tous deux le modèle $[p\lambda_k C]$ qui fournit la deuxième plus petite erreur de classement soit 0.16. AIC3 offre d'ailleurs une plus grande marge de sécurité que AIC pour ce choix vis-à-vis des valeurs qu'ils prennent pour d'autres critères. En effet, les valeurs de AIC pour $[p\lambda_k C]$ et $[p\lambda_k D_k A D_k']$ sont assez proches et il s'en est fallu de peu que AIC ne détecte ce modèle plus complexe, modèle fournissant une erreur de classement moins bonne. On retrouve donc une tendance de AIC à surestimer la complexité. Le choix effectué par AIC et AIC3 n'est pas aberrant car les vrais paramètres indiquent des proportions, des matrices forme et orientation relativement peu différentes mais par contre des volumes bien distincts (voir tableau 4.8). Le modèle $[p_k\lambda_k C_k]$ qui correspondrait aux «vrais» paramètres est choisi par L (ce qui est normal car ce modèle est le plus complexe) et fournit une erreur de 0.20. Le nombre de paramètres libres est certainement trop grand pour le nombre de points et

Deg. lib.	Modèles	erreur	L	AIC	AIC3	BIC	ICOMP	NEC	LP	E	CLM	C
5	$[p\lambda I]$	0.387	69.9	-129.7	-124.7	-118.1	-135.6	2.8	13.0	23.1	56.8	46.8
6	$[p\lambda_k I]$	0.360	70.7	-129.3	-123.3	-115.4	-134.5	2.7	13.6	24.4	57.0	46.2
6	$[p\lambda B]$	0.373	69.9	-127.8	-121.8	-113.9	-122.9	3.0	12.8	22.6	57.1	47.3
7	$[p\lambda_k B]$	0.347	70.7	-127.4	-120.4	-111.1	-119.1	2.8	12.7	23.4	57.9	47.3
7	$[p\lambda B_k]$	0.347	69.9	-125.8	-118.8	-109.5	-119.0	2.9	12.3	21.9	57.6	48.0
8	$[p\lambda_k B_k]$	0.347	70.7	-125.4	-117.4	-106.8	-114.4	2.8	13.0	23.7	57.7	47.0
7	$[p\lambda C]$	0.187	73.4	-132.9	-125.9	-116.6	-134.6	11.0	9.39	19.3	64.0	54.1
8	$[p\lambda_k C]$	0.160	76.0	**-135.9**	**-127.9**	-117.4	-136.0	4.0	8.01	16.7	68.0	59.2
9	$[p\lambda C_k]$	0.227	75.0	-132.0	-123.0	-111.2	-125.8	9.8	19.5	32.4	55.5	42.7
10	$[p\lambda_k C_k]$	0.200	77.0	-134.1	-124.1	-110.9	-132.9	2.9	6.95	15.3	70.1	61.7
8	$[p\lambda D A_k D']$	0.213	74.8	-133.7	-125.7	-115.1	-131.3	10.0	18.5	32.1	56.3	42.7
9	$[p\lambda_k D A_k D']$	0.200	76.4	-134.9	-125.9	-114.0	-134.6	3.2	7.24	15.3	69.2	61.2
8	$[p\lambda D_k A D'_k]$	0.200	74.5	-133.0	-125.0	-114.4	-134.2	7.1	10.3	19.5	64.1	55.0
9	$[p\lambda_k D_k A D'_k]$	0.200	76.7	-135.3	-126.3	-114.5	-133.7	3.5	8.44	17.3	68.2	59.4
6	$[p_k\lambda I]$	0.387	72.2	-132.3	-126.3	**-118.4**	-136.2	1.5	6.44	15.6	65.7	56.6
7	$[p_k\lambda_k I]$	0.387	72.2	-130.3	-123.3	-114.1	-128.8	1.5	6.34	15.4	65.8	56.8
7	$[p_k\lambda B]$	0.387	72.3	-130.5	-123.5	-114.3	-129.6	1.5	5.84	14.6	66.4	57.7
8	$[p_k\lambda_k B]$	0.387	72.3	-128.5	-120.5	-110.0	-124.1	1.4	5.65	14.2	66.6	58.0
8	$[p_k\lambda B_k]$	0.387	72.9	-129.8	-121.8	-111.2	-121.3	1.2	5.26	12.6	67.6	60.3
9	$[p_k\lambda_k B_k]$	0.400	73.1	-128.2	-119.2	-107.3	-116.6	**1.0**	4.81	11.2	68.3	61.9
8	$[p_k\lambda C]$	0.213	73.5	-131.0	-123.0	-112.4	-133.8	11.0	9.29	19.2	64.2	54.2
9	$[p_k\lambda_k C]$	0.387	76.7	-135.3	-126.3	-114.5	**-139.1**	1.7	4.1	8.63	72.6	68.0
10	$[p_k\lambda C_k]$	0.293	75.4	-130.8	-120.8	-107.7	-127.1	4.9	8.66	17.9	66.8	57.5
11	$[p_k\lambda_k C_k]$	0.200	**77.0**	-132.1	-121.1	-106.6	-130.4	2.9	6.81	15.2	70.2	61.9
9	$[p_k\lambda D A_k D']$	0.293	75.4	-132.8	-123.8	-112.0	-128.2	5.1	9.11	18.7	66.3	56.7
10	$[p_k\lambda_k D A_k D']$	0.413	76.8	-133.6	-123.6	-110.4	-131.5	1.4	**3.26**	**7.33**	**73.5**	**69.5**
9	$[p_k\lambda D_k A D'_k]$	0.293	75.5	-133.1	-124.1	-112.2	-135.0	3.8	6.92	14.6	68.6	61.0
10	$[p_k\lambda_k D_k A D'_k]$	**0.147**	76.7	-133.4	-123.4	-110.2	-131.7	3.5	8.43	17.2	68.3	59.5

TAB. 4.9 – *Valeur des critères obtenue avec EM sur les données réelles Habbema. Les valeurs optimales de choix de modèles sont en gras.*

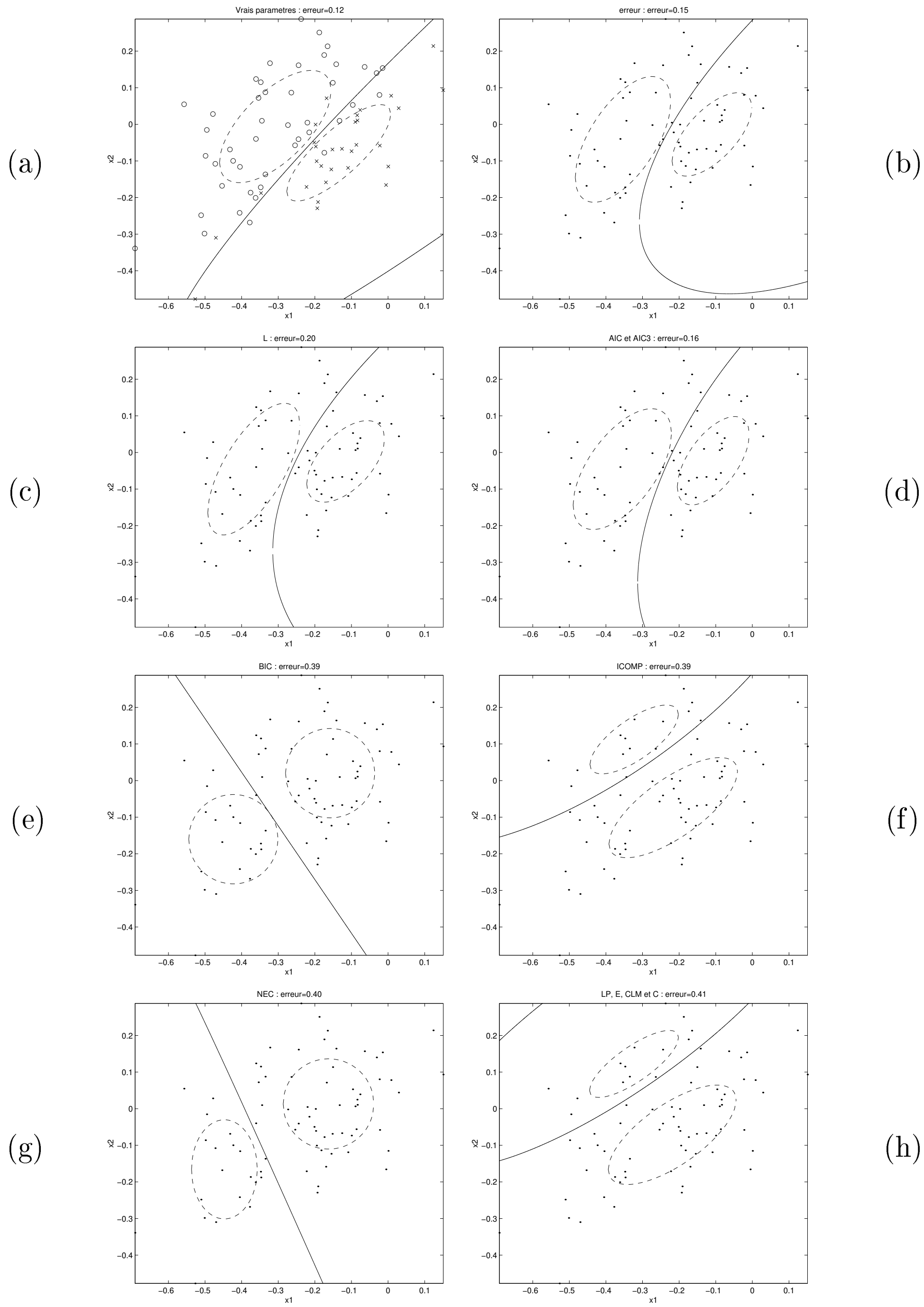


FIG. 4.4 – *Ellipses de variances (en tirets) et frontière de classification (en continu) choisies par les différents critères : (a) «vrais» paramètres, (b) erreur, (c) L, (d) AIC et AIC3, (e) BIC, (f) ICOMP, (g) NEC, (h) LP, E, CLM et C.*

l'estimation manque alors de robustesse par EM. Les critères BIC, ICOMP, NEC, CLM, C, LP et C sont très médiocres car avoisinant les 40% d'erreur. BIC détecte le modèle sphérique $[p_k\lambda_k I]$ à 7 degrés de liberté tandis que $[p\lambda_k C]$ en comporte un de plus. On retrouve alors une tendance qu'avait BIC avec le nombre de classes et les données simulées précédentes, c'est-à-dire le fait de pénaliser trop fortement la complexité. Comme nous l'avions constaté avec les données simulées aussi, ICOMP choisit un modèle trop complexe (il permet les proportions libres avec le modèle $[p_k\lambda_k C]$) et NEC choisit un modèle inadapté (il retient le modèle diagonal $[p_k\lambda_k B_k]$ alors que les classes ne sont visiblement pas alignées suivant les axes). Les critères LP, E, CLM et C retiennent le modèle $[p_k\lambda_k D A_k D']$ qui est un des modèles les plus complexes (10 degrés de liberté).

Sur la figure 4.4, on peut voir que les deux classes sont assez correctement identifiées par les critères d'erreur, L, AIC et AIC3. Par contre, on comprend les très mauvais résultats de BIC (erreur 0.39), ICOMP (0.39), NEC (0.40) et le groupe CLM, C, LP, E (0.41) au vu des frontières de classification sur cette figure.

En conclusion, on constate donc que les résultats sur les données réelles confirment les grandes lignes du comportement des critères obtenu sur les données simulées. Les critères AIC3 et AIC sont plutôt adaptés à la détection du modèle gaussien. Les critères BIC et NEC sous-estiment la complexité et tous les autres la surestime.

4.3 Expériences en discrimination

Nous proposons de poursuivre l'étude du comportement de certains critères dans un contexte de discrimination (Celeux et al. [32]). Il s'agit des critères L, AIC, AIC3, BIC, AWE, NEC et des critères de validation croisée VC^- et VC^+. Les critères NEC et L sont utilisés pour la même raison que précédemment. Le critère AWE n'est pourtant, en principe, pas construit pour trouver le modèle gaussien (voir la section 3.3.2). Mais, dans ce contexte de discrimination, le maximum de vraisemblance est absolument identique au maximum de vraisemblance classifiante et la différence entre les critères AIC, AIC3, BIC et AWE réside alors uniquement dans le terme de pénalisation. Il peut donc être intéressant d'étudier l'influence de cette pénalisation.

4.3.1 Données simulées intensives

Conditions d'expérience

L'étude du comportement des critères se fait sur des données simulées générées avec les mêmes paramètres que ceux utilisés à la section 4.2.1. Il y a néanmoins deux changements importants :

1. nous ne considérons que les jeux de paramètres avec proportions égales ($p_1 = p_2 = 0.5$) ;
2. deux tailles d'échantillon sont considérées : des échantillons de petite taille ($n = 10$) et des échantillons de taille moyenne ($n = 50$).

Dans ces conditions, le nombre total de jeux de paramètres est alors de 74 ({9 modèles (p_k égales) $\times$ 3 distances + 5 modèles (p_k égales) $\times$ 2 distances} $\times$ 2 tailles d'échantillon). Pour chacun de ces 74 jeux de paramètres nous avons généré 30 échantillons. Pour tous les échantillons, nous estimons les paramètres avec les 14 modèles gaussiens (les proportions des modèles sont égales). Les critères L, AIC, AIC3, BIC, AWE, NEC, VC$^-$ et VC$^+$ sont alors calculés.

Mesure de performance

Comme en classification automatique, le traitement des données nécessite la mise en œuvre d'un outil de synthèse résumant la performance de chacun des critères. Nous proposons une méthode proche de la précédente en mesurant la *performance relative* d'un critère $\mathcal{C}$ pour un échantillon par

$$\frac{\min(\mathrm{P}_e(\mathcal{M}))}{\mathrm{P}_e(\{\mathcal{M} :\ \arg(\mathcal{C}(\mathcal{M}))\})}.$$

Le numérateur correspond à l'erreur minimale de classement obtenue parmi tous les modèles $\mathcal{M}$ pour l'échantillon concerné. Le dénominateur représente l'erreur de classement obtenue avec le modèle choisi au sens du critère $\mathcal{C}$. Cet indice est donc compris entre 0 et 1. Il indique une bonne performance du critère étudié lorsqu'il s'approche de 1, et mauvaise lorsqu'il s'en éloigne. Nous ne jugeons donc toujours pas un critère sur son taux de détection du «bon» modèle sous-jacent, mais plutôt sur sa capacité à choisir un modèle fournissant une bonne règle de classement $u_{\hat{\theta}}$. En pratique $\mathrm{P}_e(\mathcal{M})$, l'erreur de classement $\mathrm{P}_e(u_{\hat{\theta}})$, est estimée précisément à l'aide d'un échantillon de 10000 points i.i.d. de la vraie distribution.

Résultats pour les échantillons de petite taille

Pour chacun des 30 échantillons de taille 10 engendrés par chacune des 37 situations, nous estimons les valeurs des 8 critères (voir un exemple sur le tableau 4.10) puis nous en déduisons les performances relatives de chacun d'entre eux. Utilisant toutes les performances relatives obtenues à partir des 30×34 tableaux, nous calculons les moyennes et écart-types des performances relatives de tous les critères (tableaux 4.11 et 4.12). Nous déduisons alors des histogrammes de la performance relative moyenne des critères (figure 4.5). L'analyse de ces histogrammes permet de ranger les critères dans l'ordre décroissant de performance suivant : VC$^-$, VC$^+$, AIC3, NEC, AWE, BIC, AIC et L.

Modèles	P_e	L	AIC	AIC3	BIC	AWE	NEC	VC
$[p\lambda I]$	0.172	-28.6	67.3	72.3	68.9	95.4	-1.13	0.4
$[p\lambda_k I]$	0.156	-28.6	69.3	75.3	71.1	103.0	-1.14	0.4
$[p\lambda B]$	0.201	-28.6	69.3	75.3	71.1	103.0	-1.13	0.4
$[p\lambda_k B]$	0.215	-28.6	71.3	78.3	73.4	110.6	-1.13	0.4
$[p\lambda B_k]$	0.276	-26.5	67.1	74.1	69.2	106.3	-1.95	0.3
$[p\lambda_k B_k]$	0.345	-26.4	68.9	76.9	71.3	113.8	-2.07	0.4
$[p\lambda C]$	0.214	-28.6	71.3	78.3	73.4	110.5	-1.04	0.4
$[p\lambda_k C]$	0.278	-28.6	73.2	81.2	75.7	118.1	-1.03	0.4
$[p\lambda C_k]$	0.434	-25.7	69.5	78.5	72.3	120.0	-2.37	0.5
$[p\lambda_k C_k]$	0.465	-25.5	71.1	81.1	74.1	127.2	-2.77	0.5
$[p\lambda D A_k D']$	0.430	-25.8	67.6	75.6	70.0	112.4	-2.33	0.5
$[p\lambda_k D A_k D']$	0.467	-25.5	69.1	78.1	71.8	119.6	-2.72	0.5
$[p\lambda D_k A D'_k]$	0.257	-26.3	68.7	76.7	71.1	113.6	-1.64	0.4
$[p\lambda_k D_k A D'_k]$	0.356	-26.2	70.4	79.4	73.1	120.9	-1.69	0.5

TAB. 4.10 – *Tableau des valeurs des critères pour un échantillon de taille 10 parmi les 37×30 échantillons de taille 10. Sur cet exemple, le modèle simulé est $[p\lambda C]$ avec un degré moyen de mélange.*

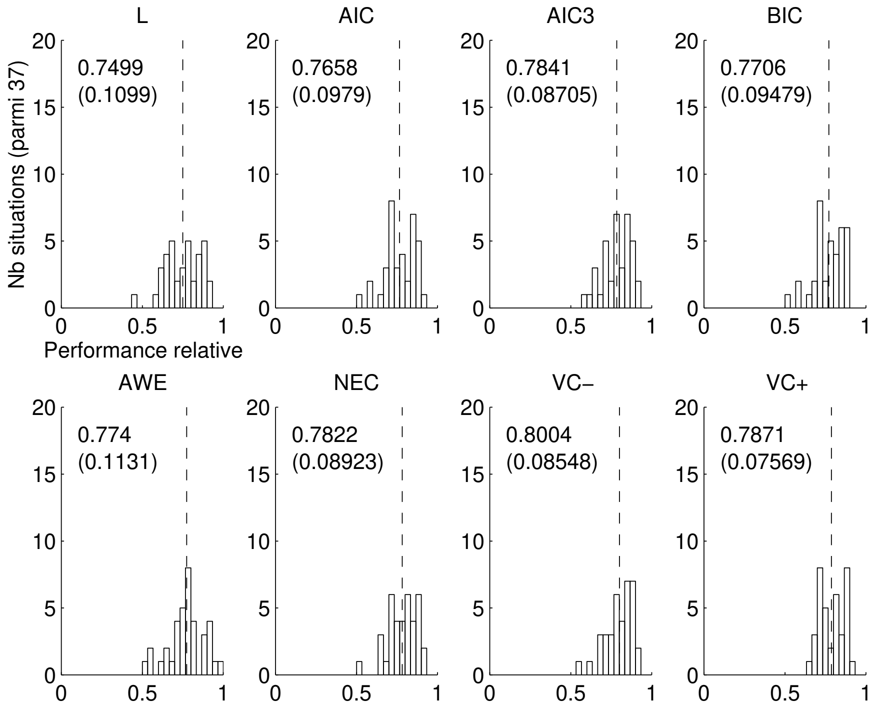


FIG. 4.5 – *Histogramme de la performance relative moyenne des 37 situations pour les critères dans le cas $n = 10$. La moyenne de cette performance moyenne est tracée en tirets, sa valeur est marquée en haut. Son écart-type est entre parenthèses.*

Modèle simulé	Degré de mélange	Performance relative							
		L	AIC	AIC3	BIC	AWE	NEC	VC^-	VC^+
	fort	0.813 (0.090)	0.895 (0.116)	0.911 (0.113)	0.896 (0.117)	0.969 (0.059)	0.844 (0.108)	0.932 (0.110)	0.869 (0.110)
$[p\lambda I]$	moyen	0.727 (0.185)	0.789 (0.191)	0.851 (0.188)	0.807 (0.194)	0.956 (0.051)	0.845 (0.149)	0.897 (0.163)	0.801 (0.171)
	faible	0.454 (0.259)	0.507 (0.314)	0.649 (0.350)	0.532 (0.313)	0.921 (0.171)	0.654 (0.242)	0.915 (0.159)	0.635 (0.247)
	fort	0.891 (0.077)	0.881 (0.101)	0.861 (0.106)	0.862 (0.103)	0.815 (0.098)	0.898 (0.077)	0.875 (0.100)	0.889 (0.090)
$[p\lambda_k I]$	moyen	0.760 (0.160)	0.773 (0.174)	0.877 (0.135)	0.797 (0.179)	0.876 (0.128)	0.819 (0.173)	0.853 (0.158)	0.806 (0.152)
	faible	0.603 (0.249)	0.707 (0.266)	0.763 (0.229)	0.748 (0.260)	0.782 (0.164)	0.805 (0.183)	0.807 (0.203)	0.758 (0.242)
	fort	0.858 (0.114)	0.891 (0.116)	0.889 (0.120)	0.888 (0.116)	0.893 (0.091)	0.869 (0.110)	0.866 (0.120)	0.878 (0.117)
$[p\lambda B]$	moyen	0.679 (0.180)	0.777 (0.208)	0.852 (0.145)	0.806 (0.184)	0.800 (0.186)	0.805 (0.168)	0.825 (0.172)	0.767 (0.164)
	faible	0.609 (0.216)	0.705 (0.246)	0.777 (0.223)	0.713 (0.251)	0.736 (0.193)	0.644 (0.170)	0.767 (0.205)	0.701 (0.241)
	fort	0.882 (0.081)	0.863 (0.085)	0.843 (0.087)	0.858 (0.089)	0.813 (0.094)	0.878 (0.081)	0.850 (0.099)	0.872 (0.103)
$[p\lambda_k B]$	moyen	0.783 (0.177)	0.757 (0.171)	0.772 (0.173)	0.767 (0.171)	0.762 (0.163)	0.846 (0.133)	0.780 (0.157)	0.777 (0.157)
	faible	0.658 (0.260)	0.673 (0.262)	0.668 (0.235)	0.685 (0.244)	0.712 (0.218)	0.709 (0.202)	0.727 (0.218)	0.711 (0.183)
	fort	-	-	-	-	-	-	-	-
$[p\lambda B_k]$	moyen	0.794 (0.183)	0.818 (0.175)	0.817 (0.167)	0.825 (0.173)	0.735 (0.161)	0.745 (0.224)	0.784 (0.151)	0.763 (0.177)
	faible	0.639 (0.239)	0.708 (0.248)	0.729 (0.259)	0.718 (0.246)	0.676 (0.243)	0.648 (0.227)	0.734 (0.229)	0.713 (0.211)
	fort	-	-	-	-	-	-	-	-
$[p\lambda_k B_k]$	moyen	0.782 (0.168)	0.758 (0.180)	0.771 (0.184)	0.763 (0.181)	0.710 (0.144)	0.816 (0.130)	0.786 (0.142)	0.825 (0.147)
	faible	0.705 (0.237)	0.714 (0.244)	0.704 (0.229)	0.711 (0.235)	0.605 (0.170)	0.705 (0.207)	0.696 (0.220)	0.725 (0.194)
	fort	0.837 (0.113)	0.849 (0.110)	0.883 (0.109)	0.852 (0.107)	0.924 (0.079)	0.831 (0.114)	0.877 (0.106)	0.859 (0.104)
$[p\lambda C]$	moyen	0.693 (0.190)	0.707 (0.193)	0.742 (0.208)	0.702 (0.195)	0.790 (0.196)	0.744 (0.173)	0.783 (0.207)	0.729 (0.202)
	faible	0.682 (0.246)	0.754 (0.248)	0.778 (0.204)	0.770 (0.242)	0.716 (0.177)	0.737 (0.178)	0.821 (0.205)	0.770 (0.193)

TAB. 4.11 – *Moyenne (et écart-type) de la performance relative des critères avec 10 points.*

Modèle simulé	Degré de mélange	Performance relative							
		L	AIC	AIC3	BIC	AWE	NEC	VC^-	VC^+
$[p\lambda_k C]$	fort	0.860 (0.073)	0.843 (0.091)	0.841 (0.093)	0.838 (0.092)	0.812 (0.106)	0.903 (0.086)	0.881 (0.103)	0.860 (0.082)
	moyen	0.782 (0.160)	0.770 (0.163)	0.786 (0.161)	0.771 (0.163)	0.797 (0.142)	0.780 (0.171)	0.848 (0.115)	0.821 (0.157)
	faible	0.638 (0.285)	0.645 (0.269)	0.640 (0.260)	0.639 (0.273)	0.640 (0.199)	0.722 (0.202)	0.675 (0.185)	0.737 (0.268)
$[p\lambda C_k]$	fort	-	-	-	-	-	-	-	-
	moyen	0.883 (0.125)	0.900 (0.135)	0.889 (0.138)	0.896 (0.135)	0.789 (0.107)	0.784 (0.140)	0.827 (0.116)	0.867 (0.138)
	faible	0.685 (0.205)	0.726 (0.204)	0.711 (0.215)	0.732 (0.210)	0.740 (0.167)	0.784 (0.174)	0.711 (0.190)	0.713 (0.201)
$[p\lambda_k C_k]$	fort	-	-	-	-	-	-	-	-
	moyen	0.909 (0.093)	0.852 (0.137)	0.847 (0.121)	0.846 (0.138)	0.773 (0.133)	0.793 (0.132)	0.792 (0.159)	0.823 (0.152)
	faible	0.762 (0.223)	0.707 (0.216)	0.720 (0.206)	0.715 (0.215)	0.556 (0.165)	0.705 (0.227)	0.693 (0.212)	0.719 (0.207)
$[p\lambda D A_k D']$	fort	0.889 (0.089)	0.885 (0.089)	0.900 (0.091)	0.886 (0.093)	0.885 (0.082)	0.894 (0.090)	0.898 (0.086)	0.894 (0.090)
	moyen	0.785 (0.161)	0.857 (0.132)	0.824 (0.147)	0.845 (0.131)	0.741 (0.170)	0.723 (0.190)	0.743 (0.179)	0.819 (0.165)
	faible	0.620 (0.254)	0.676 (0.264)	0.665 (0.254)	0.669 (0.263)	0.554 (0.261)	0.522 (0.244)	0.629 (0.266)	0.668 (0.257)
$[p\lambda_k D A_k D']$	fort	0.873 (0.084)	0.840 (0.102)	0.823 (0.118)	0.833 (0.110)	0.830 (0.129)	0.907 (0.093)	0.887 (0.115)	0.906 (0.102)
	moyen	0.748 (0.197)	0.706 (0.179)	0.723 (0.174)	0.716 (0.173)	0.725 (0.181)	0.747 (0.142)	0.756 (0.152)	0.757 (0.165)
	faible	0.690 (0.255)	0.588 (0.230)	0.580 (0.219)	0.569 (0.222)	0.513 (0.233)	0.685 (0.271)	0.548 (0.198)	0.697 (0.243)
$[p\lambda D_k A D'_k]$	fort	0.905 (0.077)	0.885 (0.111)	0.856 (0.121)	0.884 (0.111)	0.783 (0.106)	0.884 (0.103)	0.835 (0.116)	0.875 (0.108)
	moyen	0.814 (0.174)	0.856 (0.166)	0.873 (0.165)	0.870 (0.155)	0.931 (0.090)	0.900 (0.105)	0.881 (0.143)	0.881 (0.143)
	faible	0.639 (0.233)	0.693 (0.231)	0.783 (0.218)	0.721 (0.228)	0.919 (0.135)	0.851 (0.173)	0.859 (0.184)	0.722 (0.188)
$[p\lambda_k D_k A D'_k]$	fort	-	-	-	-	-	-	-	-
	moyen	0.834 (0.170)	0.804 (0.155)	0.789 (0.148)	0.799 (0.152)	0.797 (0.127)	0.807 (0.132)	0.847 (0.133)	0.852 (0.142)
	faible	0.582 (0.203)	0.577 (0.239)	0.623 (0.245)	0.584 (0.237)	0.660 (0.234)	0.707 (0.189)	0.729 (0.219)	0.669 (0.248)

TAB. 4.12 – *Moyenne (et écart-type) de la performance relative des critères avec 10 points (suite et fin).*

P_e	L	AIC	AIC3	BIC	AWE	NEC	VC^-	VC^+
7.33	10.00	7.49	6.91	7.32	5.42	7.85	6.10	8.14

TAB. 4.13 – *Nombre moyen de paramètres des modèles choisis par les critères avec 10 points. Le nombre moyen des paramètres des modèles sous-jacents est égal à 7.54.*

Le tableau 4.13 représente le nombre moyen de paramètres des modèles choisis par chacun des critères. La colonne $\mathbf{P}_e$ correspond au choix qu'on aurait obtenu en prenant le modèle minimisant $\mathbf{P}_e(u_{\hat{\theta}})$. On constate que le critère AIC3 choisit des modèles plus simples que les critères L, AIC et BIC, mais choisit des modèles plus complexes que le critère AWE. Ce comportement pouvait se prévoir avec le degré de pénalisation de la complexité présent dans chacun des critères (remarquons que AIC3 est plus pénalisant que BIC pour $n = 10$, tandis que pour $n = 50$ ce sera le contraire). Le critère AIC3 étant le meilleur parmi les autres critères de pénalisation de la complexité, il semble donc réaliser un bon compromis entre la vraisemblance et la complexité du modèle. Comme nous pouvions aussi le prévoir, le tableau 4.13 montre que le critère VC$^-$ choisit des modèles plus simples que le critère VC$^+$. Le critère de validation croisée VC$^-$ est donc meilleur que le critère VC$^+$ car il préfère les modèles plus simples. À ce sujet, il est à noter que la vraie probabilité d'erreur $\mathbf{P}_e$, elle-même, choisit des modèles plus simples que les vrais modèles sous-jacents. Il semble donc, dans le cas d'échantillons de petites tailles, qu'il soit préférable de choisir des modèles assez simples pour obtenir une règle de classement acceptable. Le critère NEC donne de bons résultats car de l'ordre du critère ICOMP. Il choisit pourtant des modèles plus complexes que tous les autres critères, excepté le critère L.

Le critère VC$^-$ choisit des modèles très simples en moyenne : le modèle linéaire sphérique $[p\lambda I]$ (46 % des cas), le modèle sphérique à volumes libres $[p\lambda_k I]$ (12 %) et le modèle linéaire général $[p\lambda C]$ (11 %). Il faut toutefois se garder de tomber dans l'excès en choisissant systématiquement des modèles trop simples. Par exemple, le critère AWE choisit principalement des modèles linéaires sphériques $[p\lambda I]$ (73 % des cas) mais est nettement moins bon que le critère de validation croisée VC$^-$.

Résultats pour les échantillons de taille moyenne

Les moyennes et écart-types des performances relatives sont disponibles dans les tableaux 4.14 et 4.15. Les histogrammes associés sont sur la figure 4.6. Le classement des critères par performance décroissante est cette fois le suivant : AIC3, BIC, AIC, L, VC$^+$, AWE, VC$^-$ et NEC. Ce classement par performance relative moyenne ne doit néanmoins pas cacher que l'ensemble des critères (critère de vraisemblance inclus) est assez bon. En effet, la performance relative moyenne sur tous les modèles est maintenant de 0.9275 alors qu'elle ne dépassait pas 0.7767 pour des échantillons plus petits.

AIC3 semble de nouveau réaliser un compromis efficace entre la vraisemblance et la complexité d'un modèle. Il se situe entre les critères L et AIC qui choisissent des modèles trop complexes et les critères BIC et AWE qui choisissent des modèles trop simples (voir le tableau 4.16). On peut constater que le critère AWE pénalise d'ailleurs la complexité de manière excessive car même le critère de vraisemblance est meilleur. Les critères de validation croisée ont des performances en baisse en comparaison des critères de pénalisation. Comme le critère AWE, ils sont moins bons que le critère de vraisemblance. De plus, le critère de validation croisée VC$^+$, critère qui choisit des modèles plus complexes que VC$^-$, est meilleur que le critère VC$^-$. On constate aussi que la vraie probabilité d'erreur $\mathbf{P}_e$ choisit en moyenne des modèles plus complexes que les modèles sous-jacents aux données (tableau 4.16). Il semble alors qu'une taille d'échantillon plus conséquente que dans le cas précédent permette d'obtenir une discrimination

Modèle simulé	Degré de mélange	Performance relative							
		L	AIC	AIC3	BIC	AWE	NEC	VC^-	VC^+
$[p\lambda I]$	fort	0.952 (0.045)	0.968 (0.049)	0.978 (0.039)	0.983 (0.035)	0.989 (0.017)	0.958 (0.046)	0.978 (0.035)	0.969 (0.040)
	moyen	0.944 (0.054)	0.965 (0.049)	0.980 (0.046)	0.988 (0.022)	0.994 (0.011)	0.970 (0.026)	0.978 (0.031)	0.967 (0.039)
	faible	0.910 (0.077)	0.940 (0.079)	0.948 (0.082)	0.958 (0.078)	0.988 (0.013)	0.933 (0.073)	0.960 (0.050)	0.924 (0.080)
$[p\lambda_k I]$	fort	0.906 (0.101)	0.904 (0.101)	0.901 (0.101)	0.905 (0.099)	0.867 (0.098)	0.895 (0.099)	0.911 (0.093)	0.903 (0.097)
	moyen	0.986 (0.020)	0.983 (0.018)	0.980 (0.022)	0.978 (0.023)	0.958 (0.055)	0.968 (0.059)	0.935 (0.074)	0.960 (0.065)
	faible	0.910 (0.066)	0.941 (0.066)	0.954 (0.062)	0.945 (0.071)	0.900 (0.115)	0.865 (0.122)	0.847 (0.130)	0.878 (0.103)
$[p\lambda B]$	fort	0.942 (0.038)	0.970 (0.032)	0.978 (0.025)	0.982 (0.024)	0.985 (0.029)	0.892 (0.083)	0.921 (0.088)	0.925 (0.083)
	moyen	0.943 (0.043)	0.955 (0.050)	0.961 (0.046)	0.967 (0.043)	0.994 (0.011)	0.931 (0.064)	0.955 (0.055)	0.958 (0.044)
	faible	0.934 (0.056)	0.951 (0.053)	0.963 (0.043)	0.968 (0.040)	0.970 (0.041)	0.901 (0.111)	0.934 (0.098)	0.941 (0.060)
$[p\lambda_k B]$	fort	0.955 (0.049)	0.966 (0.046)	0.974 (0.042)	0.980 (0.037)	0.924 (0.108)	0.912 (0.087)	0.935 (0.060)	0.943 (0.054)
	moyen	0.978 (0.026)	0.973 (0.033)	0.977 (0.033)	0.973 (0.043)	0.918 (0.090)	0.944 (0.074)	0.892 (0.125)	0.955 (0.074)
	faible	0.946 (0.053)	0.962 (0.052)	0.961 (0.048)	0.971 (0.041)	0.928 (0.092)	0.793 (0.157)	0.816 (0.153)	0.923 (0.082)
$[p\lambda B_k]$	fort	-	-	-	-	-	-	-	-
	moyen	0.977 (0.026)	0.987 (0.023)	0.987 (0.024)	0.982 (0.039)	0.990 (0.033)	0.907 (0.127)	0.965 (0.069)	0.956 (0.089)
	faible	0.880 (0.080)	0.864 (0.089)	0.866 (0.083)	0.867 (0.081)	0.880 (0.081)	0.520 (0.160)	0.779 (0.184)	0.867 (0.100)
$[p\lambda_k B_k]$	fort	-	-	-	-	-	-	-	-
	moyen	0.838 (0.092)	0.832 (0.095)	0.830 (0.090)	0.830 (0.090)	0.817 (0.087)	0.773 (0.161)	0.813 (0.111)	0.817 (0.103)
	faible	0.905 (0.045)	0.901 (0.046)	0.905 (0.053)	0.905 (0.053)	0.831 (0.101)	0.520 (0.191)	0.708 (0.179)	0.844 (0.127)
$[p\lambda C]$	fort	0.949 (0.061)	0.961 (0.059)	0.965 (0.059)	0.970 (0.053)	0.960 (0.056)	0.942 (0.061)	0.944 (0.074)	0.948 (0.072)
	moyen	0.966 (0.033)	0.976 (0.035)	0.980 (0.029)	0.983 (0.022)	0.892 (0.121)	0.832 (0.122)	0.956 (0.081)	0.964 (0.064)
	faible	0.949 (0.044)	0.959 (0.042)	0.971 (0.035)	0.960 (0.077)	0.780 (0.219)	0.733 (0.196)	0.889 (0.179)	0.949 (0.066)

TAB. 4.14 – *Moyenne (et écart-type) de la performance relative des critères avec 50 points.*

Modèle simulé	Degré de mélange	Performance relative							
		L	AIC	AIC3	BIC	AWE	NEC	VC$^-$	VC$^+$
$[p\lambda_k C]$	fort	0.958 (0.046)	0.971 (0.051)	0.969 (0.064)	0.970 (0.063)	0.916 (0.104)	0.940 (0.051)	0.942 (0.081)	0.933 (0.091)
	moyen	0.973 (0.042)	0.973 (0.045)	0.971 (0.042)	0.967 (0.047)	0.883 (0.112)	0.854 (0.161)	0.900 (0.120)	0.912 (0.112)
	faible	0.951 (0.049)	0.966 (0.046)	0.972 (0.043)	0.966 (0.051)	0.820 (0.184)	0.685 (0.148)	0.862 (0.139)	0.908 (0.088)
$[p\lambda C_k]$	fort	-	-	-	-	-	-	-	-
	moyen	0.974 (0.042)	0.979 (0.042)	0.980 (0.038)	0.981 (0.038)	0.985 (0.037)	0.855 (0.125)	0.923 (0.118)	0.950 (0.085)
	faible	0.968 (0.026)	0.982 (0.023)	0.986 (0.022)	0.989 (0.019)	0.979 (0.058)	0.786 (0.103)	0.874 (0.137)	0.919 (0.104)
$[p\lambda_k C_k]$	fort	-	-	-	-	-	-	-	-
	moyen	0.986 (0.028)	0.986 (0.033)	0.985 (0.034)	0.975 (0.052)	0.924 (0.097)	0.936 (0.117)	0.904 (0.094)	0.922 (0.093)
	faible	0.968 (0.045)	0.966 (0.048)	0.963 (0.061)	0.963 (0.061)	0.925 (0.094)	0.697 (0.201)	0.833 (0.127)	0.938 (0.088)
$[p\lambda D A_k D']$	fort	0.971 (0.036)	0.968 (0.048)	0.964 (0.047)	0.969 (0.031)	0.938 (0.058)	0.927 (0.088)	0.908 (0.082)	0.937 (0.073)
	moyen	0.964 (0.033)	0.972 (0.034)	0.974 (0.036)	0.973 (0.039)	0.959 (0.068)	0.885 (0.159)	0.928 (0.121)	0.940 (0.095)
	faible	0.951 (0.063)	0.961 (0.044)	0.967 (0.037)	0.958 (0.048)	0.933 (0.059)	0.715 (0.302)	0.945 (0.051)	0.956 (0.054)
$[p\lambda_k D A_k D']$	fort	0.919 (0.072)	0.920 (0.070)	0.917 (0.073)	0.924 (0.072)	0.906 (0.090)	0.891 (0.069)	0.896 (0.073)	0.911 (0.069)
	moyen	0.973 (0.047)	0.969 (0.053)	0.966 (0.052)	0.962 (0.063)	0.912 (0.084)	0.829 (0.159)	0.890 (0.125)	0.917 (0.123)
	faible	0.955 (0.059)	0.949 (0.070)	0.937 (0.071)	0.924 (0.083)	0.880 (0.126)	0.651 (0.238)	0.853 (0.159)	0.930 (0.081)
$[p\lambda D_k A D'_k]$	fort	0.985 (0.014)	0.988 (0.011)	0.989 (0.010)	0.989 (0.010)	0.750 (0.160)	0.975 (0.069)	0.980 (0.039)	0.985 (0.014)
	moyen	0.976 (0.019)	0.980 (0.021)	0.981 (0.021)	0.978 (0.023)	0.956 (0.021)	0.967 (0.023)	0.965 (0.025)	0.978 (0.017)
	faible	0.923 (0.076)	0.942 (0.060)	0.943 (0.060)	0.943 (0.060)	0.963 (0.041)	0.945 (0.057)	0.945 (0.074)	0.910 (0.092)
$[p\lambda_k D_k A D'_k]$	fort	-	-	-	-	-	-	-	-
	moyen	0.986 (0.018)	0.984 (0.020)	0.984 (0.020)	0.984 (0.020)	0.836 (0.122)	0.949 (0.076)	0.890 (0.102)	0.955 (0.060)
	faible	0.940 (0.071)	0.939 (0.084)	0.933 (0.087)	0.924 (0.091)	0.853 (0.113)	0.822 (0.125)	0.877 (0.107)	0.926 (0.095)

TAB. 4.15 – *Moyenne (et écart-type) de la performance relative des critères avec 50 points (suite et fin).*

P_e	L	AIC	AIC3	BIC	AWE	NEC	VC$^-$	VC$^+$
8.11	10.00	7.84	7.59	7.45	6.65	7.76	7.06	8.84

TAB. 4.16 – *Nombre moyen de paramètres des modèles choisis par les critères avec 50 points. Le nombre moyen des paramètres des modèles sous-jacents est égal à 7.54.*

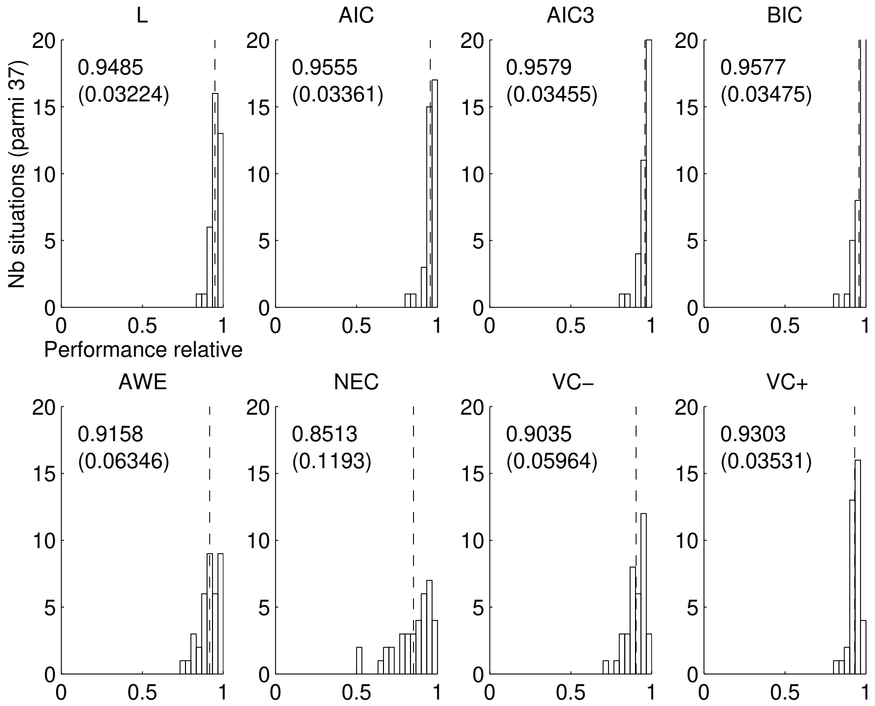


FIG. 4.6 – *Histogramme de la performance relative moyenne des 37 situations pour les critères dans le cas $n = 50$. La moyenne de cette performance moyenne est tracée en tirets, sa valeur est marquée en haut. Son écart-type est entre parenthèses.*

efficace même avec un modèle relativement complexe.

4.3.2 Autres simulations

L'étude précédente se voulait systématique, donc nous avions du nous limiter à un cas restreint. Cette section suggère détendre ces résultats à d'autres situations : choix d'un modèle parmi un ensemble de modèles inadaptés aux données et choix d'un modèle en grande dimension.

Comportement avec des modèles inadaptés

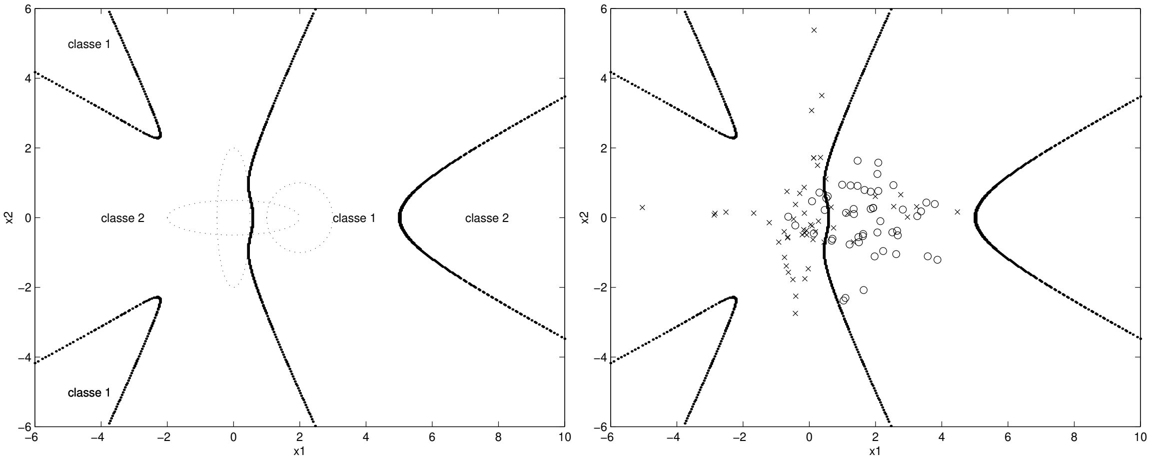
Nous nous proposons d'étudier le comportement des critères pour la détection d'un modèle gaussien lorsque les données ne proviennent pas nécessairement de classes gaussiennes. Nous choisissons deux classes bidimensionnelles, la première étant gaussienne, la seconde étant elle-même un mélange de deux lois gaussiennes donc étant finalement non gaussienne. Nous avons le mélange

$$f(\boldsymbol{x}) = 0.5\ \phi_1(\boldsymbol{x}) + 0.5\ \phi_2(\boldsymbol{x})$$

avec

$$\begin{aligned}\phi_1(\boldsymbol{x}) &= \mathcal{N}((2,0)', \boldsymbol{I})\\ \phi_2(\boldsymbol{x}) &= 0.5\ \mathcal{N}((0,0)', \mathrm{diag}(0.25,4)) + 0.5\ \mathcal{N}((0,0)', \mathrm{diag}(4,0.25)).\end{aligned}$$

Le degré de mélange obtenu est d'ordre moyen car $\mathrm{P}_e(u) \approx 0.15$. La figure 4.7 montre la frontière de classement associée à cette distribution, elle fournit les ellipses de variance de chaque gaussienne et elle donne aussi un exemple d'échantillon.



(a) (b)

FIG. 4.7 – *Mélange d'une classe gaussienne et d'une classe mélange de deux gaussiennes : frontière de classification et matrices de variance des trois gaussiennes (a) et exemple d'un échantillon (b).*

Nous générons 30 échantillons de taille 10 provenant de cette distribution et, sur chacun d'entre eux, nous estimons l'ensemble des critères pour les 14 modèles. La performance relative et le nombre moyen de paramètres associés à chaque critère sont déduits (voir le tableau 4.17). Nous constatons alors que le critère de validation croisée VC^- est le plus performant. On remarque qu'il choisit des modèles assez simples mais moins simples que le critère AWE. Le critère NEC est correct car il occupe la deuxième position. Ensuite viennent les critères AWE et VC^+.

	P_e	L	AIC	AIC3	BIC	AWE	NEC	VC^-	VC^+
perf. rela.	1.000	0.685	0.723	0.758	0.751	0.799	0.810	0.832	0.799
nb. param.	7.13	10.00	7.03	6.50	6.70	5.43	7.50	5.96	7.73

TAB. 4.17 – *Performance relative et nombre moyen de paramètres retenu pour chaque critère dans le cas de 10 points avec des modèles inadaptés.*

Nous recommençons le même processus avec des échantillons de 50 points et nous obtenons les résultats du tableau 4.18. Tous les critères s'améliorent, AWE et VC^- étant en tête. La validation croisée semble confirmer le bon comportement qui avait été constaté sur les simulations intensives de la section précédente. De plus, la validation croisée semble mieux résister que la plupart des critères à un modèle inadapté.

	P_e	L	AIC	AIC3	BIC	AWE	NEC	VC^-	VC^+
perf. rela.	1.000	0.882	0.881	0.882	0.888	0.929	0.883	0.918	0.899
nb. param.	6.96	10.00	6.76	6.23	6.10	5.36	6.70	6.43	7.76

TAB. 4.18 – *Performance relative et nombre moyen de paramètres retenu pour chaque critère dans le cas de 50 points avec des modèles inadaptés.*

Comportement en grande dimension

L'objectif est ici d'étudier le comportement des critères précédents sur un exemple en grande dimension. Nous choisissons la dimension 10 avec deux classes gaussiennes de même proportion et de même matrice de variance. Les centres et les matrices de variance sont égaux à :

$$\boldsymbol{\mu}_1 = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad \boldsymbol{\mu}_2 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad \boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 3 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 4 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 5 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 6 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 7 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 8 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 9 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 10 \end{pmatrix}.$$

Le degré de mélange ainsi obtenu est d'ordre moyen.

Nous générons 30 échantillons de 60 points à partir de cette distribution. Le tableau 4.19 regroupe les performances relatives et le nombre moyen de paramètres de chaque critère.

	P_e	L	AIC	AIC3	BIC	AWE	NEC	VC^-	VC^+
perf. rela.	1.000	0.972	0.985	0.994	0.996	0.997	0.402	0.997	0.972
nb. param.	87.00	130.00	80.70	76.63	75.40	75.00	36.76	75.00	129.93

TAB. 4.19 – *Performance relative et nombre moyen de paramètres retenu pour chaque critère dans le cas d'une grande dimension.*

L'ensemble des critère est assez bon, excepté le critère NEC qui s'effrondre complètement. Les critères VC^- et AWE donnent des résultats strictement équivalents sur notre exemple et ils correspondent aux deux meilleurs critères, suivis de près par BIC et AIC3. Ainsi le critère AIC3 semble confirmer, même en grande dimension, le bon comportement que nous avions constaté sur des données simulées.

4.4 Conclusion

Nous avons comparé bon nombre de critères du chapitre précédent pour sélectionner le modèle gaussien en classification automatique puis en discrimination. Notre jugement de validité d'un critère s'est fait sur la base d'un critère de performance relative. Ce critère mesurait la qualité de la partition en classification automatique et la qualité de la règle de classement en discrimination. L'optique était alors résolument classification. Nous avons constater que :

- en classification automatique, le critère AIC3 a un bon comportement aussi bien sur des données simulées que sur des données réelles. Par contre les critères classification que sont NEC, CLM, C, LP et E ne semblent pas adaptés au choix du modèle gaussien ;
- en discrimination, le critère AIC3 a montré aussi son bon comportement sur bon nombre de simulations. Pourtant le critère qui semble le plus robuste, aussi bien pour des tailles d'échantillons différentes que pour des dimensions différentes, que pour des modèles inadaptés, semble être le critère de validation croisée VC^-.

Finalement, le bon comportement général du critère AIC3 est à retenir de cette étude. Il est vrai qu'en discrimination la validation croisée donne très souvent des résultats meilleurs mais la différence de performance n'est pas très marquée (excepté le cas d'un modèle inadapté où la validation est bien meilleure). On peut alors se demander si la simplicité de calcul du critère AIC3 n'est pas préférable au coût en calcul qu'implique la validation croisée, et ce pour, finalement, des performances très voisines.

Chapitre 5

Sélection du nombre de classes : étude comparative

5.1 Introduction

Le chapitre 3 a présenté de nombreuses méthodes utilisables pour trouver le nombre de classes en modèle de mélange : critères de type AIC, critère NEC. De nouvelles propositions ont été faites également : détection d'un coude de la vraisemblance avec une hypothèse de précision, critères de pénalisation de la vraisemblance par la classifiabilité comme la vraisemblance classifiante. Cette partie se propose de mener une étude sur des données simulées et sur des données réelles pour comparer le comportement de ces différents critères. Nous voyons alors tout l'intérêt des critères pénalisant la vraisemblance par la classifiabilité (C, CL et CLM). Nous concluons que CL nous semble être un critère préférable aux deux autres.

La section 5.2 se focalise sur des données simulées. Tout d'abord, nous considérons de manière intensive des données uni- et bidimensionnelles variées : différentes séparations des classes, différentes matrices de variances, diverses proportions et plusieurs nombres de classes. Ensuite, nous étudions un exemple de cinq classes en dimension plus élevée, la dimension quatre. Les algorithmes EM et CEM avaient été utilisés avec des modèles gaussiens adaptés pour les essais précédents et nous menons alors une discussion sur l'influence de mauvais modèles gaussiens sur la détection du nombre de classes. Enfin, nous envisageons d'utiliser les critères sur des classes non gaussiennes mais tout en utilisant un modèle gaussien : la bonne résistance des critères de classification est ainsi mise en évidence.

La section 5.3 étend les résultats précédents à des exemples de données réelles : les Iris de Fisher (données classiques et donc incontournables), les papillons et les durées d'éruption d'un geyser. Les conclusions déduites des données simulées sont alors confirmées et enrichies.

5.2 Données simulées

Nous proposons d'étudier le comportement des critères mélange (AIC, AIC3, BIC et ICOMP) et des critères classification (NEC, CLM, CLM2, CL, CL2 et AWE) lors de la recherche du nombre de classes. Cette étude s'appuie ici sur des données simulées variées : différentes dimensions (1,2 et 4), différents nombres de classes (1, 2, 3 et 5), diverses matrices de variance pour les classes (matrices sphériques, diagonales, etc.), plusieurs séparations entres les classes (forte, moyenne, faible), deux tailles d'échantillon (200 et 625) et enfin deux types de proportions (proportions égales ou différentes). Nous considérons aussi un exemple où une classe n'est pas gaussienne. Sur chacun des échantillons générés par ces jeux de paramètres, nous utilisons généralement un modèle gaussien adapté (au moins en ce qui concerne les matrices de variance), c'est-à-dire que nous supposons que celui-ci est connu. Néanmoins, nous menons aussi une étude sur la résistance des critères lorsque ce modèle n'est pas le plus adapté (cas d'un modèle trop simple et cas d'un modèle trop complexe). Le critère ICOMP, pour l'ensemble de cette section, a été calculé par la méthode de Monte-Carlo adoptée par Cutler et Windham, méthode que nous avons décrite au chapitre 3 : son calcul nécessite donc l'utilisation des résultats de l'annexe C

Remarquons que nous n'avons pas essayé notre procédure de détection d'un coude de la vraisemblance qui utilise la notion de précision sur les données. En effet, dans le cas des données simulées, la précision nous semble trop subjective. Nous reportons alors l'emploi de cette procédure dans le cas des données réelles.

5.2.1 Données uni- et bidimensionnelles

Les données

Nous proposons de commencer l'étude des critères avec des données simulées en dimension un et en dimension deux. Nous prenons des nombres de classes variés (de une à trois classes), des matrices de variance assez diverses (matrices sphériques, diagonales, etc.), une taille d'échantillon égale à 200. À chaque jeu de matrices de variance, nous associons deux types de proportions (sauf dans le cas monoclasse) : proportions égales puis proportions différentes. À chaque association matrice de variance – proportions, nous associons trois catégories de centres de classes (sauf dans le cas monoclasse de nouveau) : les centres donnant un degré de mélange fort, ceux donnant un degré de mélange moyen et ceux donnant un degré de mélange faible. La mesure du degré de mélange n'a pas été donnée, contrairement au chapitre précédent, par l'erreur théorique de classement associé au paramètre $\boldsymbol{\theta}$:

$$\mathbf{P}_e(u_\theta) = \sum_{k=1}^{K} p_k \int_{\mathbb{R}^d} \mathcal{L}(u_\theta(\boldsymbol{x}), k)\phi_k(\boldsymbol{x}|\boldsymbol{a}_k)d\boldsymbol{x}$$

où $\mathcal{L}$ est le coût 0–1. En effet, nous reprochons à cette mesure d'indiquer facilement une bonne séparation des classes si l'une d'entre elles est fortement peuplée par rapport aux autres. Pour pallier ce problème, nous proposons la mesure s qui est une variante de $\mathbf{P}_e$ donnant la même importance à toutes les classes :

$$s(\boldsymbol{\theta}) = \sum_{k=1}^{K} \frac{1}{K} \int_{\mathbb{R}^d} \mathcal{L}(u_\theta(\boldsymbol{x}), k)\phi_k(\boldsymbol{x}|\boldsymbol{a}_k)d\boldsymbol{x}.$$

Ainsi, par exemple, si deux classes ont même centre avec $p_1 = 0.9$ et $p_2 = 0.1$ (les proportions sont donc très différentes) et que la densité de la première est presque toujours supérieure à celle de la seconde, nous avons $\mathbf{P}_e \simeq 0.1$ et $s \simeq 0.5$. Notre mesure s représente donc bien mieux le fait que les classes sont indiscernables. Nous remarquons bien sûr que les deux mesures sont équivalentes ($s = \mathbf{P}_e$) si les classes ont même proportion. Les trois degrés de mélange sont ensuite définis par $s_1 = 0.15$ (mélange fort), $s_2 = 0.10$ (mélange moyen) et $s_3 = 0.05$ (mélange faible).

Au total, nous avons défini 9 jeux différents de matrices de variance. À chaque jeu nous associons les trois degrés de mélange et les deux types de proportions (excepté, dans les deux cas, la situation monoclasse). Nous obtenons finalement les 44 jeux de paramètres distincts (7 matrices de variance $\times$ 2 proportions $\times$ 3 séparations + 2 matrices de variance) qui sont sur le tableau 5.1. La figure 5.1 donne une vue d'ensemble des types de configurations considérés.

N°	d	K	s	p_k	$\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$
1	1	1	s_3	$=$	
2		2	s_1	$=$	$\boldsymbol{\mu}_2 = 2.0$
				$\neq$	$\boldsymbol{\mu}_2 = 2.2$
			s_2	$=$	$\boldsymbol{\mu}_2 = 2.6$
				$\neq$	$\boldsymbol{\mu}_2 = 2.7$
			s_3	$=$	$\boldsymbol{\mu}_2 = 3.3$
				$\neq$	$\boldsymbol{\mu}_2 = 3.4$
3		3	s_1	$=$	$\boldsymbol{\mu}_2 = 2.4,\ \boldsymbol{\mu}_3 = 4.8$
				$\neq$	$\boldsymbol{\mu}_2 = 2.9,\ \boldsymbol{\mu}_3 = 5.2$
			s_2	$=$	$\boldsymbol{\mu}_2 = 2.9,\ \boldsymbol{\mu}_3 = 5.8$
				$\neq$	$\boldsymbol{\mu}_2 = 3.4,\ \boldsymbol{\mu}_3 = 6.0$
			s_3	$=$	$\boldsymbol{\mu}_2 = 3.6,\ \boldsymbol{\mu}_3 = 7.2$
				$\neq$	$\boldsymbol{\mu}_2 = 3.9,\ \boldsymbol{\mu}_3 = 7.35$
4		2	s_1	$=$	$\boldsymbol{\mu}_2 = 0.0,\ \boldsymbol{\Sigma}_2 = 0.0250$
				$\neq$	$\boldsymbol{\mu}_2 = 0.0,\ \boldsymbol{\Sigma}_2 = 0.0240$
			s_2	$=$	$\boldsymbol{\mu}_2 = 0.0,\ \boldsymbol{\Sigma}_2 = 0.0100$
				$\neq$	$\boldsymbol{\mu}_2 = 0.0,\ \boldsymbol{\Sigma}_2 = 0.0090$
			s_3	$=$	$\boldsymbol{\mu}_2 = 0.0,\ \boldsymbol{\Sigma}_2 = 0.0021$
				$\neq$	$\boldsymbol{\mu}_2 = 0.0,\ \boldsymbol{\Sigma}_2 = 0.0018$
5	2	1	s_3	$=$	

N°	d	K	s	p_k	$\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$
6	2	2	s_1	$=$	$\boldsymbol{\mu}_2 = (2.0, 0.0)')$
				$\neq$	$\boldsymbol{\mu}_2 = (2.2, 0.0)'$
			s_2	$=$	$\boldsymbol{\mu}_2 = (2.6, 0.0)'$
				$\neq$	$\boldsymbol{\mu}_2 = (2.7, 0.0)'$
			s_3	$=$	$\boldsymbol{\mu}_2 = (3.3, 0.0)'$
				$\neq$	$\boldsymbol{\mu}_2 = (3.4, 0.0)'$
7		3	s_1	$=$	$\boldsymbol{\mu}_2 = (1.8, 1.7)',\ \boldsymbol{\mu}_3 = (1.8, -1.7)'$
				$\neq$	$\boldsymbol{\mu}_2 = (2.0, 1.9)',\ \boldsymbol{\mu}_3 = (2.0, -1.9)'$
			s_2	$=$	$\boldsymbol{\mu}_2 = (2.0, 2.0)',\ \boldsymbol{\mu}_3 = (2.0, -2.0)'$
				$\neq$	$\boldsymbol{\mu}_2 = (2.2, 2.2)',\ \boldsymbol{\mu}_3 = (2.2, -2.2)'$
			s_3	$=$	$\boldsymbol{\mu}_2 = (2.6, 2.5)',\ \boldsymbol{\mu}_3 = (2.6, -2.5)'$
				$\neq$	$\boldsymbol{\mu}_2 = (2.7, 2.7)',\ \boldsymbol{\mu}_3 = (2.7, -2.7)'$
8		2	s_1	$=$	$\boldsymbol{\mu}_2 = (0.0, 0.0)',\ \boldsymbol{\Sigma}_2 = 0.100\boldsymbol{I}$
				$\neq$	$\boldsymbol{\mu}_2 = (0.0, 0.0)',\ \boldsymbol{\Sigma}_2 = 0.086\boldsymbol{I}$
			s_2	$=$	$\boldsymbol{\mu}_2 = (0.0, 0.0)',\ \boldsymbol{\Sigma}_2 = 0.055\boldsymbol{I}$
				$\neq$	$\boldsymbol{\mu}_2 = (0.0, 0.0)',\ \boldsymbol{\Sigma}_2 = 0.051\boldsymbol{I}$
			s_3	$=$	$\boldsymbol{\mu}_2 = (0.0, 0.0)',\ \boldsymbol{\Sigma}_2 = 0.022\boldsymbol{I}$
				$\neq$	$\boldsymbol{\mu}_2 = (0.0, 0.0)',\ \boldsymbol{\Sigma}_2 = 0.020\boldsymbol{I}$
9		2	s_1	$=$	$\boldsymbol{\mu}_2 = (2.6, 0.0)',\ \boldsymbol{\Sigma}_1 = \boldsymbol{A}_1,\ \boldsymbol{\Sigma}_2 = \boldsymbol{A}_2$
				$\neq$	$\boldsymbol{\mu}_2 = (2.7, 0.0)',\ \boldsymbol{\Sigma}_1 = \boldsymbol{A}_1,\ \boldsymbol{\Sigma}_2 = \boldsymbol{A}_2$
			s_2	$=$	$\boldsymbol{\mu}_2 = (3.3, 0.0)',\ \boldsymbol{\Sigma}_1 = \boldsymbol{A}_1,\ \boldsymbol{\Sigma}_2 = \boldsymbol{A}_2$
				$\neq$	$\boldsymbol{\mu}_2 = (3.4, 0.0)',\ \boldsymbol{\Sigma}_1 = \boldsymbol{A}_1,\ \boldsymbol{\Sigma}_2 = \boldsymbol{A}_2$
			s_3	$=$	$\boldsymbol{\mu}_2 = (4.6, 0.0)',\ \boldsymbol{\Sigma}_1 = \boldsymbol{A}_1,\ \boldsymbol{\Sigma}_2 = \boldsymbol{A}_2$
				$\neq$	$\boldsymbol{\mu}_2 = (4.6, 0.0)',\ \boldsymbol{\Sigma}_1 = \boldsymbol{A}_1,\ \boldsymbol{\Sigma}_2 = \boldsymbol{A}_2$

TAB. 5.1 – *Paramètres des simulations uni- et bivariées. Les degrés de mélange fort, moyen et faible sont respectivement notés s_1, s_2 et s_3. Les proportions égales entre classes sont notées «=», les proportions différentes sont notées «≠». Ce dernier cas correspond à $p_1 = 0.3, p_2 = 0.7$ quand $K = 2$ et à $p_1 = 0.1, p_2 = 0.3, p_3 = 0.6$ quand $K = 3$. Les centres $\boldsymbol{\mu}_1$ sont toujours centrés à l'origine. Si les matrices $\boldsymbol{\Sigma}_k$ ne sont pas explicites, elles correspondent à $\boldsymbol{\Sigma}_k = \boldsymbol{I}$. Nous notons enfin $\boldsymbol{A}_1 = diag(4, 1)$ et $\boldsymbol{A}_2 = diag(1, 9)$.*

Pour chacun des 44 jeux de paramètres, nous générons 30 échantillons de 200 points. Pour chaque échantillon, nous initialisons l'algorithme CEM avec les vrais paramètres si le nombre recherché de classes est égal au «vrai» nombre de classes dans le mélange,

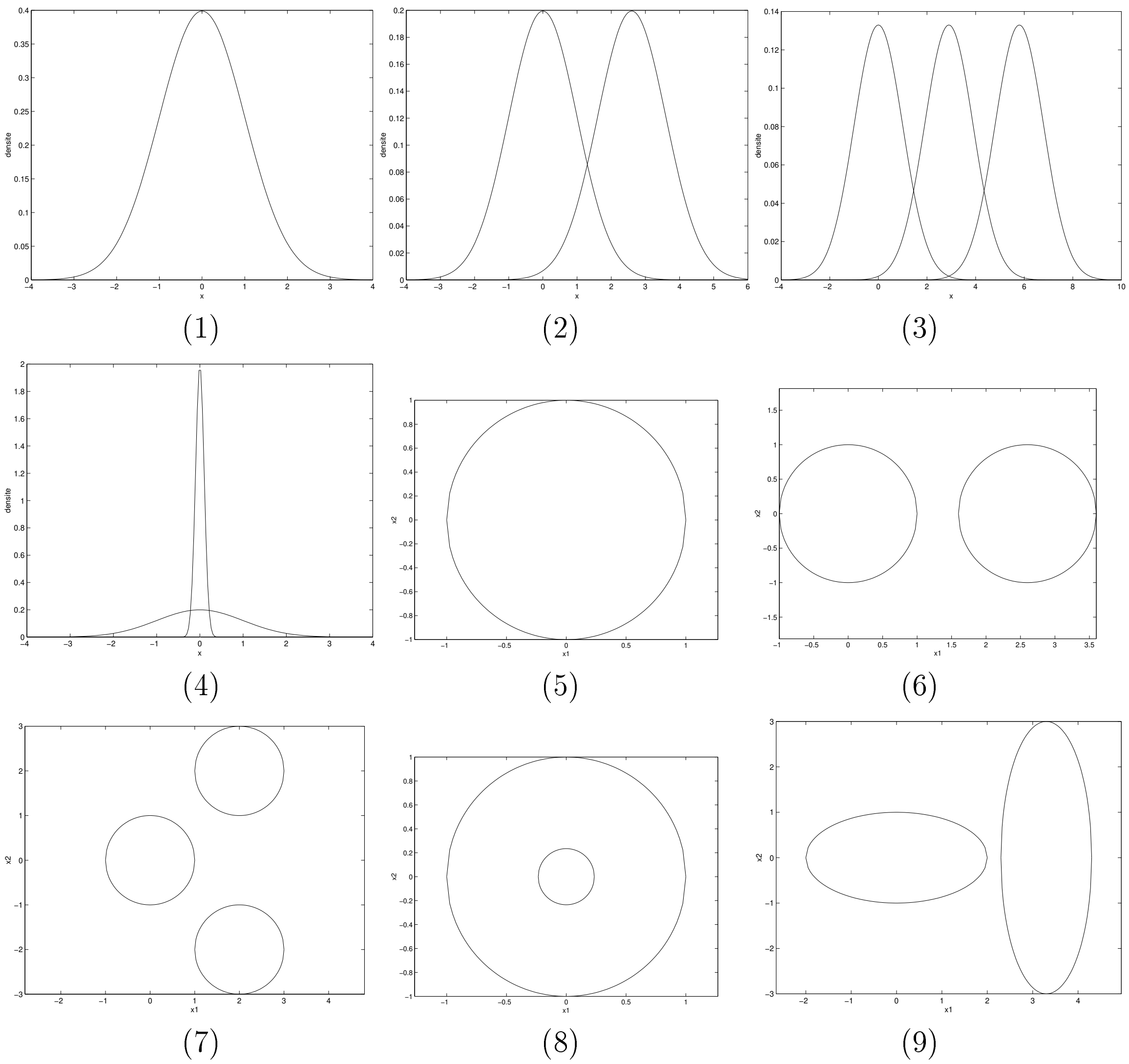


FIG. 5.1 – *Visualisation des densités pour les simulation univariées (figures (1) à (4)) et des ellipses de variance (figures (5) à (9)) pour les simulations bivariées. Seul le degré de mélange moyen avec proportions égales est représenté.*

	Paramètres initiaux de CEM	
	vrais	aléatoires
$\hat{\mathrm{KL}}(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$	0.0328	0.0361
erreur($\hat{\boldsymbol{z}}$)	0.0778	0.0835

TAB. 5.2 – *Comportement des estimateurs de $\boldsymbol{\theta}$ et de $\boldsymbol{z}$ obtenus par l'algorithme CEM quand cet algorithme est initialisé soit avec les vrais paramètres, soit avec le meilleur jeu de paramètres parmi 20 tirages aléatoires.*

sinon nous faisons 20 initialisations au hasard de l'algorithme pour ne retenir que la meilleure performance au sens du critère de vraisemblance classifiante. L'initialisation de CEM avec les vrais paramètres, lorsque cela est possible, permet de réduire le temps de simulation. Outre cet aspect pratique, cette stratégie est justifiée du point de vue des résultats puisque l'estimation du paramètre $\boldsymbol{\theta}$ ou de la partition $\boldsymbol{z}$ est assez proche entre les deux stratégies si le nombre d'initialisations aléatoires est suffisamment grand (20 par exemple). Nous avons illustré ceci sur la tableau 5.2 où nous avons généré 30 échantillons du jeu de paramètre numéro 9 du tableau 5.1 avec des proportions différentes et un degré de mélange moyen, puis où nous avons lancé CEM dans les deux situations avec le modèle $[p\lambda_k B_k]$: les estimateurs obtenus sont en effet assez proches. Nous utilisons ensuite les paramètres trouvés par CEM pour initialiser l'algorithme EM. Cela permet de nouveau de limiter le temps de calcul car nous évitons des initialisations aléatoires de EM. Néanmoins, lorsque l'algorithme CEM n'a pu fournir un résultat (c'est-à-dire que l'on a une solution singulière car des classes sont vides), ce qui peut être le cas lorsque le nombre de classes cherché est trop grand et que les proportions sont libres dans le modèle, nous faisons tout de même 20 initialisations aléatoires de EM.

Concernant le modèle gaussien utilisé par les algorithmes EM et CEM, les contraintes sur les matrices de variance correspondent le plus possible aux vrais paramètres ayant généré l'échantillon concerné. Les proportions sont successivement égales (même si en réalité elles sont différentes) puis laissées libres (même si elles sont égales). Ainsi, on utilise le modèle $[p\lambda I]$ puis le modèle $[p_k\lambda I]$ pour tous les échantillons provenant des jeux de paramètres 1 à 3 et 5 à 7 du tableau 5.1, le modèle $[p\lambda_k I]$ puis le modèle $[p_k\lambda_k I]$ pour les jeux 4 et 8, et enfin le modèle $[p\lambda_k B_k]$ puis le modèle $[p_k\lambda_k B_k]$ pour le jeu 9. Au sujet des différents nombres de classes considérés pour chaque échantillon, nous faisons tourner CEM et EM avec séquentiellement une à K_{sup} classes, où K_{sup} est l'entier immédiatement supérieur à $n^{0.3}$ (Bozdogan [25]) : pour 200 points, nous avons $K_{sup} = 5$. Enfin, pour chaque échantillon et pour chaque nombre de classes, nous calculons les différents critères : L, AIC, AIC3, BIC, ICOMP, NEC, CLM, CLM2, CL, CL2, C et AWE.

Résultats des simulations

Les résultats des simulations uni- et bidimensionnelles sont résumés dans le tableau 5.3. Ce tableau est lui-même réprésenté sur les histogrammes de la figure 5.2 pour les séparations moyenne et forte. Sur le tableau et sur les histogrammes, nous ne représentons pas les critères L, AIC et C : L surestime largement le nombre de classes, AIC le surestime aussi mais dans une moindre mesure et C a un comportement similaire à CLM avec une tendance marquée à choisir des nombres de classes légèrement inférieurs. De manière générale, nous retrouvons les comportements déjà énoncés de AIC3 et de BIC mais ICOMP semble avoir un comportement moins bon que celui constaté dans d'autres études car il se situe entre AIC et AIC3 en moyenne. Parmi ces critères de type mélange, il semble que ce soit BIC qui donne les meilleurs résultats si les classes sont bien séparées (d'ordre s_3). Concernant les critères classification, NEC essuie pour sa part de nombreux échecs dans les situations monoclasses. Les critères pénalisant la vraisemblance par la classifiabilité que sont les vraisemblances classifiantes CL et

CLM se distinguent par un comportement excellent pour des classes bien séparées avec un modèle à proportions égales : si les classes sont peu séparées, ces critères originaux préfèrent sous-estimer le nombre de classes. Par contre, l'usage de ces critères avec un modèle à proportions libres est à déconseiller car les critères surestiment le nombre de classes (nous l'avions d'ailleurs déjà prévu au chapitre 3). Il est néanmoins possible de conserver un modèle à proportions égales même si les proportions sont différentes en réalité. Mais les critères CL2 et CLM2, variantes très empiriques de CL et de CLM respectivement, semblent, au moins sur nos exemples, améliorer les critères CL et CLM lorsqu'un modèle à proportions libres est employé. Présentons maintenant le détail du comportement des critères.

Modèle avec proportions identiques

- **Données avec proportions identiques :** Les stratégies CL2 et CLM2 sont ici strictement équivalentes aux stratégies CL et CLM car le cas de classes «trop petites» n'est jamais réalisé. C'est la situation où les critères CL et CLM donnent les meilleurs résultats. Ils rivalisent avec, et même dépassent, tous les autres critères *si la séparation est forte* (s_3). Par contre, il est nécessaire d'avoir une bonne séparation des classes (au moins s_2) pour détecter le bon nombre de classes. C'est aussi le cas du critère AWE. Cette caractéristique de bonne séparabilité pour justifier de l'existence des classes semble imposée par la notion de vraisemblance classifiante car NEC, qui est aussi un critère de classification mais qui n'utilise pas de vraisemblance classifiante, détecte par contre des classes moins séparées.

  L'effet de la dimension est également important sur tous les critères mais surtout sur ceux utilisant un terme CL (CL, CL2, CLM, CLM2, AWE) : en dimension deux, les classes sont détectées avec une plus faible séparation qu'en dimension un. Ces mêmes critères ne surestiment aussi que très rarement le bon nombre de classes : lorsque les classes sont peu séparées, ils préfèrent agglomérer des classes entre elles. À ce propos, on peut noter que CL est le critère le plus performant comparé à CLM et AWE : CLM sous-estime le nombre de classes plus que CL et AWE beaucoup plus encore.

- **Données avec proportions différentes :** Dans ce cas le modèle est trop simple car les proportions du modèle sont égales tandis que celles des données sont différentes. Pour les mêmes raisons que précédemment, nous avons ici aussi CL2 et CLM2 respectivement équivalents à CL et à CLM. De plus, les remarques faites sur la dimension et la séparation s'appliquent toujours (excepté NEC qui est le plus performant en dimension un). Mais les performances des critères mélange AIC, AIC3, BIC et ICOMP s'effondrent totalement car ces critères surestiment exagérément le nombre de classes. Par contre les performances des critères classification CLM, CL, AWE et NEC, bien qu'en baisse aussi par rapport à la situation précédente, montrent une bonne tenue dans le cas d'une forte séparation. Il ressort donc que les critères de type classification sont plus résistants que les autres à un modèle inadapté (le modèle n'est pas assez complexe du point de vue des proportions). On retrouve encore l'ordre de sous-estimation croissant : CL, CLM, AWE.

Modèle à proportions libres Dans cette situation (les données ont des proportions aussi bien identiques que différentes), les stratégies CL et CLM surestiment alors nettement le nombre de classes avec en moyenne une surestimation plus forte pour CL. En dimension un et si les classes sont suffisamment séparées, CL et CLM donnent cependant de bons résultats avec un léger avantage à CL. Ce bon comportement de CL (83 voir 89 % de réussite en $d = 1$ pour s_3) semble pourtant en contradiction avec la théorie qui prévoyait un choix systématique de K_{sup} par ce critère. Par contre, en dimension 2, CL et CLM sont peu performants mêmes si les classes sont bien séparées et AWE est un critère de classification souvent plus adéquat dans ce cas malgré une forte tendance à sous-estimer le nombre de classes. Néanmoins, les stratégies CL2 et CLM2 donnent d'excellents résultats en éliminant la quasi totalité de la surestimation de CL et CLM. Elles s'améliorent aussi avec la dimension de l'espace et la séparation des classes.

5.2.2 Un exemple en dimension 4

Nous proposons de confirmer le bon comportement des critères de type vraisemblance classifiante en comparant l'ensemble des critères sur un jeu de données en dimension quatre. Le jeu de paramètres retenu a été utilisé à l'origine par Bozdogan [25] puis repris par Celeux et Soromenho [40]. Il comporte 625 points répartis en 5 classes. Les paramètres du modèle sont :

$$\begin{array}{lll}
p_1 = 0.12 & \boldsymbol{\mu}_1 = (10, 12, 10, 12)' & \boldsymbol{\Sigma}_1 = \boldsymbol{I} \\
p_2 = 0.16 & \boldsymbol{\mu}_2 = (8.5, 10.5, 8.5, 10.5)' & \boldsymbol{\Sigma}_2 = \boldsymbol{I} \\
p_3 = 0.20 & \boldsymbol{\mu}_3 = (12, 14, 12, 14)' & \boldsymbol{\Sigma}_3 = \boldsymbol{I} \\
p_4 = 0.24 & \boldsymbol{\mu}_4 = (13, 15, 7, 9)' & \boldsymbol{\Sigma}_4 = 4\boldsymbol{I} \\
p_5 = 0.28 & \boldsymbol{\mu}_5 = (7, 9, 13, 15)' & \boldsymbol{\Sigma}_5 = 9\boldsymbol{I}.
\end{array}$$

La séparation des classes est bonne car d'ordre s_3 ($\hat{s} \simeq 0.06$). Un exemple d'échantillon est donné sur la figure 5.3.

Nous générons 30 échantillons de cette distribution puis, sur chacun d'entre eux, nous faisons tourner CEM et EM de la même façon que précédemment. Nous utilisons successivement le modèle $[p\lambda_k I]$ à proportions égales et le modèle $[p_k\lambda_k I]$ à proportions libres. Nous calculons l'ensemble des critères pour tous les nombres de classes de $K = 1$ à $K_{sup} = 7$ (K_{sup} a de nouveau été défini par la borne $n^{0.3}$ de Bozdogan).

Les pourcentages de choix des différents nombres de classes sont disponibles sur la tableau 5.4 (les critères L, AIC et C ne sont pas représentés pour les mêmes raisons qu'auparavant). Les classes étant bien séparées sur l'exemple considéré (de l'ordre de s_3), chercher des classes de proportions égales bien qu'en réalité elles ne le soient pas n'est pas inutile au vu des résultats en dimension un et deux. Si l'on recherche donc des proportions égales, on retrouve ce qu'on avait déjà constaté, à savoir une forte surestimation des critères mélange même si BIC se comporte tout de même assez bien. CL et CLM sont excellents (de l'ordre de 90% de réussite) avec un léger avantage à CL encore. Ces critères ne surestiment jamais le nombre de classes. De plus, CL2 et CLM2 sont encore équivalents à CL et CLM dans ce cas. Le critère AWE est indecis entre 4 et 5 classes et NEC détecte encore plus nettement 4 classes mais sur ce point Celeux

p_k données	p_k modèle	d	Sep.	K	AIC3	BIC	ICOMP	NEC	CLM	CLM2	CL	CL2	AWE
égales	égales	1	s_1	=	69	58	59	61	24	24	27	27	17
				− / +	26 / 6	41 / 1	9 / 32	30 / 9	74 / 1	74 / 1	71 / 2	71 / 2	83 / 0
			s_2	=	92	89	77	82	42	42	43	43	36
				− / +	2 / 6	9 / 2	0 / 23	18 / 0	58 / 0	58 / 0	57 / 0	57 / 0	64 / 0
			s_3	=	92	97	88	79	93	93	94	94	81
				− / +	0 / 8	0 / 3	0 / 12	0 / 21	7 / 0	7 / 0	6 / 0	6 / 0	19 / 0
		2	s_1	=	96	100	44	88	43	43	51	51	13
				− / +	0 / 4	0 / 0	0 / 56	12 / 0	57 / 1	57 / 1	44 / 5	44 / 5	87 / 0
			s_2	=	93	99	43	90	86	86	93	93	57
				− / +	0 / 8	0 / 1	0 / 58	10 / 0	14 / 0	14 / 0	7 / 1	7 / 1	43 / 0
			s_3	=	97	99	57	87	100	100	100	100	100
				− / +	0 / 3	0 / 1	0 / 43	1 / 11	0 / 0	0 / 0	0 / 0	0 / 0	0 / 0
	libres	1	s_1	=	54	40	58	22	7	14	0	0	13
				− / +	42 / 3	60 / 0	23 / 19	20 / 58	71 / 22	73 / 12	66 / 34	67 / 33	86 / 1
			s_2	=	87	71	87	60	22	39	6	14	33
				− / +	8 / 6	27 / 2	3 / 10	17 / 23	57 / 21	57 / 4	56 / 39	56 / 30	67 / 0
			s_3	=	92	98	87	67	81	93	83	92	64
				− / +	0 / 8	0 / 2	0 / 13	3 / 31	6 / 13	6 / 2	5 / 12	6 / 3	36 / 0
		2	s_1	=	93	98	86	31	4	42	4	41	13
				− / +	0 / 7	3 / 0	0 / 14	1 / 68	43 / 53	50 / 8	36 / 60	38 / 21	88 / 0
			s_2	=	91	100	84	34	19	88	5	77	51
				− / +	0 / 9	0 / 0	0 / 16	2 / 64	11 / 70	12 / 0	6 / 89	6 / 18	49 / 0
			s_3	=	94	100	81	62	49	97	39	95	99
				− / +	0 / 6	0 / 0	0 / 19	3 / 35	0 / 51	0 / 3	0 / 61	0 / 5	1 / 0
différentes	égales	1	s_1	=	7	14	9	50	31	31	19	19	31
				− / +	10 / 83	20 / 66	33 / 58	1 / 49	67 / 2	67 / 2	67 / 14	67 / 14	68 / 1
			s_2	=	2	4	7	44	32	32	32	32	33
				− / +	0 / 98	6 / 90	11 / 82	0 / 56	67 / 1	67 / 1	67 / 1	67 / 1	67 / 0
			s_3	=	2	3	4	66	48	48	50	50	38
				− / +	0 / 98	0 / 97	0 / 96	3 / 31	52 / 0	52 / 0	42 / 8	42 / 8	62 / 0
		2	s_1	=	33	64	15	51	30	30	19	19	17
				− / +	1 / 67	9 / 27	0 / 85	37 / 13	58 / 12	58 / 12	49 / 32	49 / 32	83 / 0
			s_2	=	18	41	13	64	50	50	43	43	38
				− / +	1 / 81	3 / 57	0 / 87	33 / 3	41 / 9	41 / 9	30 / 28	30 / 28	63 / 0
			s_3	=	2	18	3	72	79	79	78	78	78
				− / +	0 / 98	0 / 83	0 / 98	27 / 2	13 / 8	13 / 8	6 / 16	6 / 16	23 / 0
	libres	1	s_1	=	68	56	69	22	6	23	6	7	32
				− / +	21 / 11	43 / 1	8 / 23	27 / 51	63 / 31	63 / 13	59 / 36	59 / 34	67 / 1
			s_2	=	90	84	83	37	12	33	13	18	36
				− / +	2 / 8	14 / 1	1 / 16	27 / 37	56 / 32	57 / 10	50 / 37	51 / 31	64 / 0
			s_3	=	90	100	91	86	72	98	89	99	64
				− / +	0 / 10	0 / 0	0 / 9	1 / 13	0 / 28	0 / 2	0 / 11	0 / 1	36 / 0
		2	s_1	=	88	88	78	26	16	62	19	63	39
				− / +	3 / 9	13 / 0	4 / 18	18 / 56	30 / 54	33 / 5	19 / 62	32 / 6	61 / 0
			s_2	=	91	98	78	38	30	88	22	87	57
				− / +	0 / 9	1 / 1	1 / 21	16 / 47	5 / 65	8 / 4	3 / 76	4 / 9	43 / 0
			s_3	=	92	100	88	71	36	98	22	98	100
				− / +	0 / 8	0 / 0	0 / 13	8 / 21	0 / 64	0 / 3	0 / 78	0 / 3	0 / 0

TAB. 5.3 – *Poucentage de sous-estimation (−), bonne estimation (=) et surestimation (+) du nombre de classes K pour les simulations uni- et bivariées. Par exemple, le critère AIC3, avec un degré moyen de mélange (s_2), dans le cas univarié, les classes ayant mêmes proportions, et le modèle imposant les proportions égales, détecte le bon nombre de classes dans 92% des cas, sous-estime ce nombre dans 2% des cas et le surestime dans 6% des situations.*

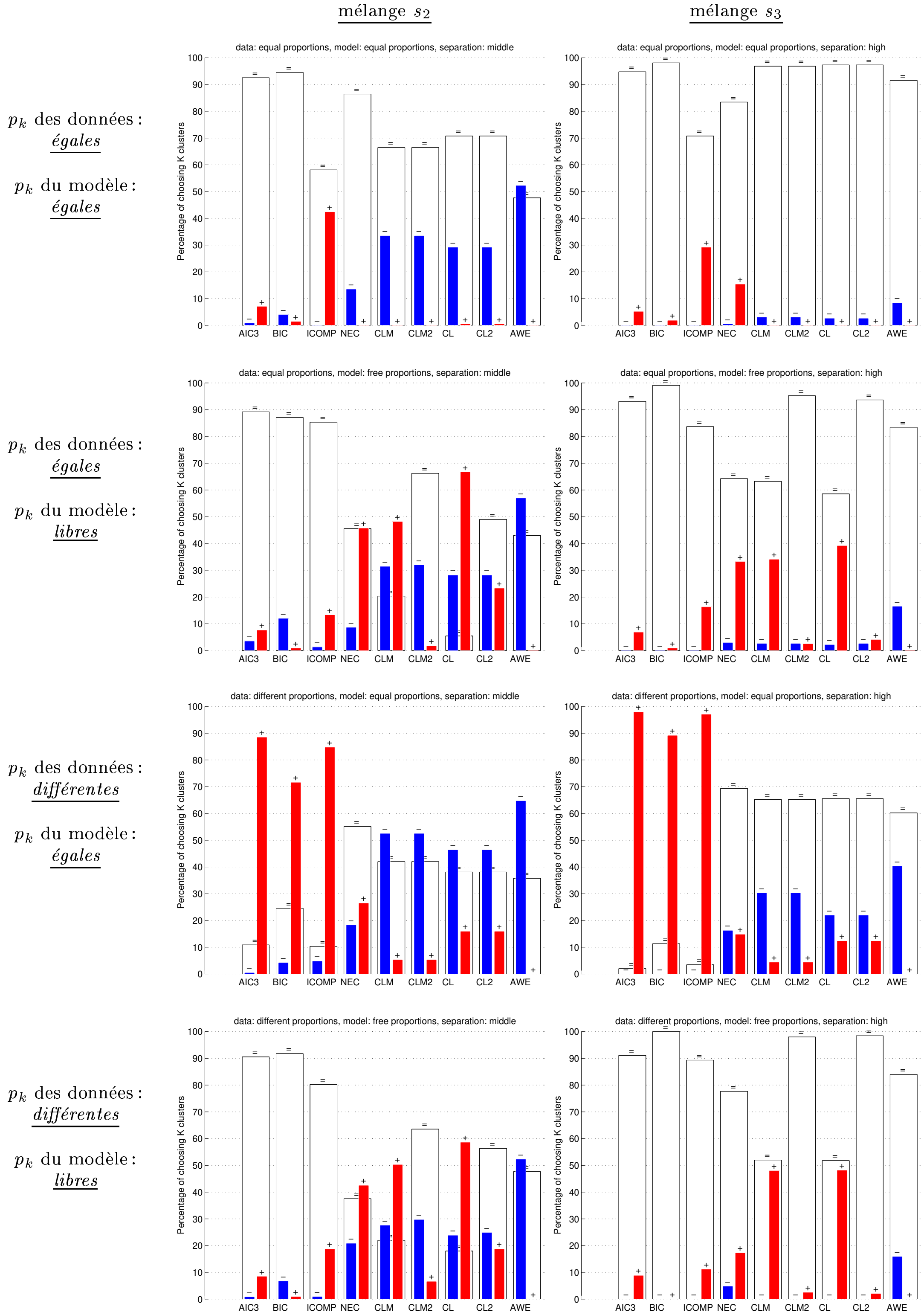


FIG. 5.2 – *Histogrammes des pourcentages de sous-estimation (barre «− »), de bonne estimation (barre blanche «= ») et de surestimation (barre «+ ») du nombre de classes. On étudie seulement le mélange moyen (s_2) et le mélange faible (s_3).*

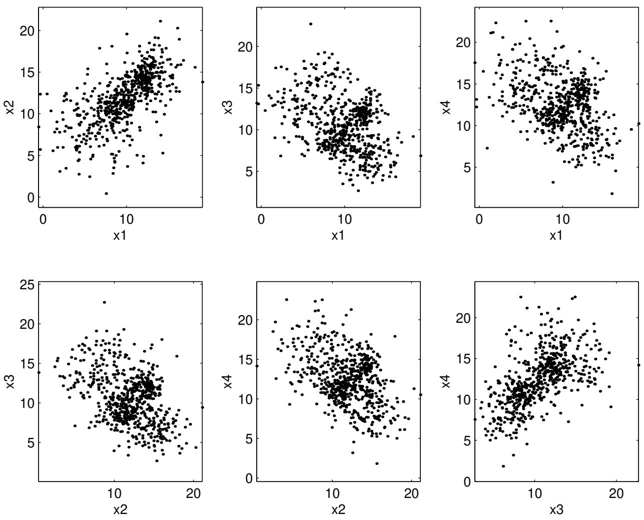


FIG. 5.3 – *Un échantillon de la distribution en dimension 4 : projection sur les 6 plans.*

modèle	K	AIC3	BIC	ICOMP	NEC	CLM	CLM2	CL	CL2	AWE
p_k égales	1	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0
	4	0	10	0	**93**	13	13	7	7	**53**
	5	**43**	**80**	**50**	7	**87**	**87**	**93**	**93**	47
	6	13	10	0	0	0	0	0	0	0
	7	**43**	0	**50**	0	0	0	0	0	0
p_k libres	1	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	0	0	0	0
	4	0	0	0	**67**	0	0	0	0	37
	5	**67**	**90**	**80**	17	13	**100**	**73**	**100**	**63**
	6	30	10	20	17	37	0	17	0	0
	7	3	0	0	0	**50**	0	10	0	0

TAB. 5.4 – *Pourcentages de choix du nombre de classes pour la simulation en dimension 4.*

et Soromenho [40] font remarquer que ce choix n'est pas déraisonnable pour le jeu de paramètres. Si les proportions du modèle sont libres, tous les critères mélange s'améliorent fortement avec tout de même une tendance à la surestimation. AWE s'améliore aussi, NEC commence à hésiter entre $K = 4$ et $K > 4$ et CLM surestime fortement le nombre de composantes. Par contre CL se maintient étonnement bien à 73% de réussite : ceci est du au fait que CEM n'a presque jamais pu converger pour $K > 4$ car des classes étaient vides. Les critères CL2 et CLM2, quant à eux, remplissent pleinement leur rôle en enrayant la surestimation de CL et CLM pour obtenir 100 % de réussite.

Cet exemple en dimension relativement élevé fait donc ressortir de nouveau le bon comportement des critères CL et CLM lorsque le modèle impose des proportions égales. Dans le cas des proportions libres, les critères CL2 et CLM2 continuent de remplir leur rôle, c'est-à-dire d'éviter la surestimation prévisible du nombre de classes par CL et CLM. Il reste aussi possible d'utiliser un modèle à proportions égales alors que les classes ont des proportions différentes car les critères classification sont assez résistants à un mauvais modèle.

5.2.3 Effet du modèle gaussien

Nous avons délibérément choisi, pour l'ensemble des simulations précédentes, un modèle gaussien adapté au moins pour les contraintes sur les matrices de variance. Il serait intéressant d'étudier aussi le comportement de l'ensemble des critères lorsque l'on utilise des modèles mal adaptés aux caractéristiques des matrices de variance des classes. Nous proposons alors de faire quelques essais dans les deux situations possibles de modèle inadéquate : soit le modèle est trop simple, soit le modèle est trop complexe.

modèle gaussien trop simple

Considérons tout d'abord le cas où le modèle gaussien est trop simple. Nous retenons le jeu de simulation numéro 9 du tableau 5.1 avec des proportions égales et une séparation forte (voir une illustration de la distribution sur la figure 5.4). Nous générons 30 échantillons de 200 points de cette distribution et nous utilisons les algorithmes EM et CEM de la même façon que précédemment. Les algorithmes sont utilisés successivement avec un modèle trop simple (le modèle sphérique $[p\lambda I]$) et un modèle relativement bien adapté (le modèle quadratique $[p\lambda_k C_k]$).

Les pourcentages de détection de chaque nombre de classes pour les deux modèles sont disponibles sur le tableau 5.5. Contrairement aux résultats constatés avec le modèle adapté $[p\lambda_k C_k]$, le modèle trop simple $[p\lambda I]$ conduit à une nette surestimation du nombre de classes par l'ensemble des critères. On retrouve l'ordre de surestimation croissant : AWE, C, CLM et CL. La figure 5.5 montre le comportement de la moyenne des critères dans le cas des deux modèles. Il ressort que les courbes associées à chaque critère sont très différentes, ce qui explique la mauvaise détection générale du nombre de classes.

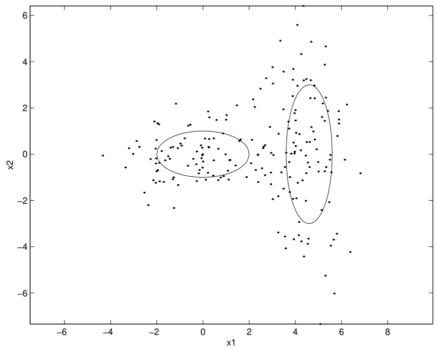


FIG. 5.4 – *Ellipses de variance et un échantillon associés à la distribution utilisée dans le cas d'un modèle trop simple.*

modèle	K	AIC	AIC3	BIC	ICOMP	NEC	CLM(2)	CL(2)	C	AWE
$[p\lambda I]$	1	0	0	0	0	0	10	3	**73**	**73**
	2	0	0	0	0	3	0	0	0	0
	3	0	0	0	0	**53**	33	13	20	23
	4	0	3	10	0	3	7	10	0	0
	5	**100**	**97**	**90**	**100**	40	**50**	**73**	7	3
$[p\lambda_k C_k]$	1	0	0	0	0	0	0	0	0	0
	2	**80**	**100**	**100**	**47**	**100**	**100**	**100**	**100**	**100**
	3	7	0	0	10	0	0	0	0	0
	4	10	0	0	13	0	0	0	0	0
	5	3	0	0	30	0	0	0	0	0

TAB. 5.5 – *Pourcentage de choix des nombres de classes en fonction d'un modèle gaussien adapté ($[p\lambda_k C_k]$) et d'un modèle trop simple ($[p\lambda I]$).*

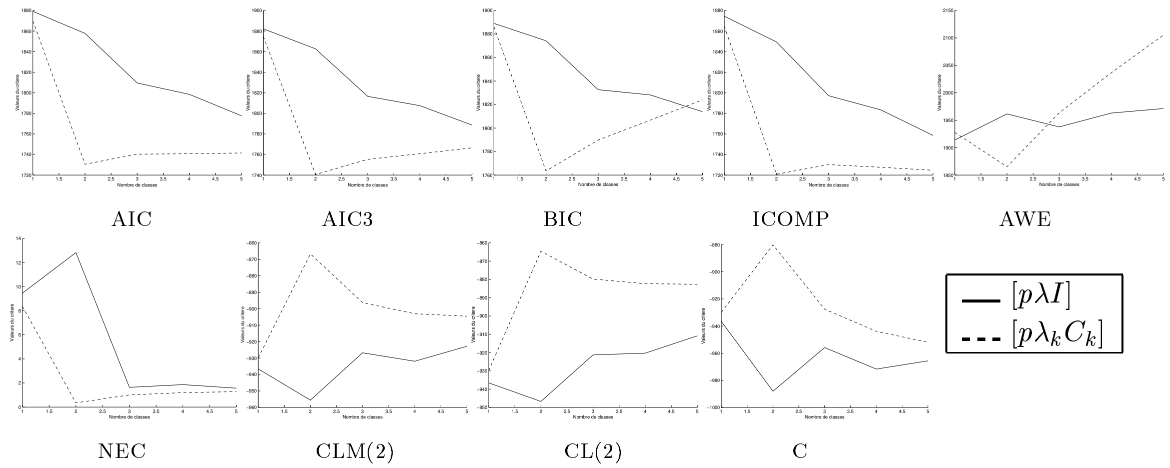


FIG. 5.5 – *Moyenne des critères sur les 30 simulations dans le cas où on utilise un modèle adapté ($[p\lambda_k C_k]$) et un modèle trop simple ($[p\lambda I]$). Le nombre de classes est en abscisse et la valeur du critère est en ordonnée.*

modèle gaussien trop complexe

Prenons maintenant un exemple qui permet détudier le comportement des critères lors de l'utilisation d'un modèle trop complexe. Nous retenons le jeu de paramètres numéro 2 du tableau 5.1 avec des proportions identiques et une séparation forte (une illustration de la distribution est donnée sur la figure 5.6). Ensuite, on suit la même démarche que dans l'étude qui précède et on obtient le tableau 5.6 des performances.

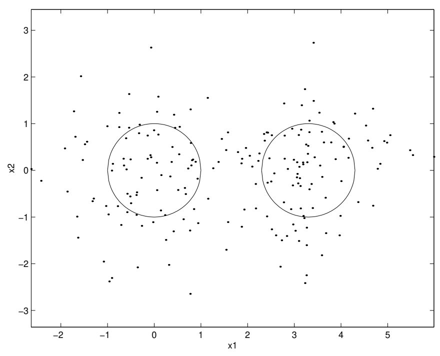


FIG. 5.6 – *Ellipses de variance et un échantillon associés à la distribution utilisée dans le cas d'un modèle trop complexe.*

Ce tableau montre que l'utilisation d'un modèle trop complexe aurait tendance, au contraire d'un modèle trop simple, à faire choisir par l'ensemble des critères un nombre de classes inférieur à celui trouvé avec un modèle adapté (excepté pour les critères AIC et ICOMP qui choisissent un nombre de classes supérieur). La figure 5.7 reporte la courbe de la moyenne des critères pour les deux modèles, et on voit nettement que les «pics» permettant la détection de deux classes sont nettement moins marqués avec le modèle complexe qu'avec le modèle simple.

modèle	K	AIC	AIC3	BIC	ICOMP	NEC	CLM(2)	CL(2)	C	AWE
$[p\lambda I]$	1	0	0	0	0	0	0	0	7	0
	2	**87**	**97**	**100**	33	**100**	**100**	**100**	**93**	**100**
	3	7	3	0	0	0	0	0	0	0
	4	3	0	0	27	0	0	0	0	0
	5	3	0	0	**40**	0	0	0	0	0
$[p\lambda_k C_k]$	1	0	0	13	0	17	13	10	**67**	**100**
	2	**77**	**100**	**87**	7	**80**	**87**	**87**	33	0
	3	10	0	0	17	0	0	0	0	0
	4	10	0	0	27	3	0	3	0	0
	5	3	0	0	**50**	0	0	0	0	0

TAB. 5.6 – *Pourcentage de choix des nombres de classes en fonction d'un modèle gaussien adapté ($[p\lambda I]$) et d'un modèle trop complexe ($[p\lambda_k C_k]$).*

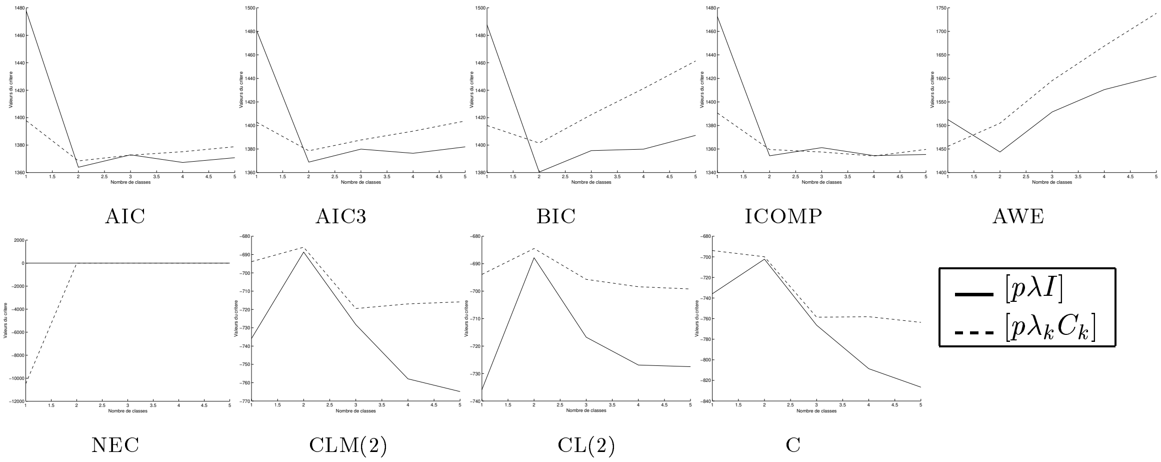


FIG. 5.7 – *Moyenne des critères sur les 30 simulations dans le cas où on utilise un modèle adapté ($[p\lambda I]$) et un modèle trop complexe ($[p\lambda_k C_k]$). Le nombre de classes est en abscisse et la valeur du critère est en ordonnée.*

En conclusion, il est important d'utiliser un modèle adapté lorsque l'on recherche le nombre de classes, car l'ensemble des critères actuels y est assez sensible. Il semble néanmoins plus judicieux d'utiliser un modèle trop complexe qu'un modèle trop simple.

5.2.4 Comportement avec une classe non gaussienne

Étudions maintenant les caractéristiques des critères lorsque l'on utilise un modèle gaussien alors qu'au moins une des classes n'est pas gaussienne. Nous considérons la distribution suivante composée d'un mélange d'une classe uniforme et d'une classe gaussienne :

$$f(\boldsymbol{x}) = 0.5\ \phi_1(\boldsymbol{x}) + 0.5\ \phi_2(\boldsymbol{x})$$

avec

$$\begin{aligned}\phi_1(\boldsymbol{x}) &= \mathcal{U}_{[-1,1]}(\boldsymbol{x}^1) \times \mathcal{U}_{[-1,1]}(\boldsymbol{x}^2)\\ \phi_2(\boldsymbol{x}) &= \mathcal{N}((3.3,0)', \boldsymbol{I}).\end{aligned}$$

Un exemple d'échantillon est représenté sur la figure 5.8 et on voit nettement que les classes sont bien séparées.

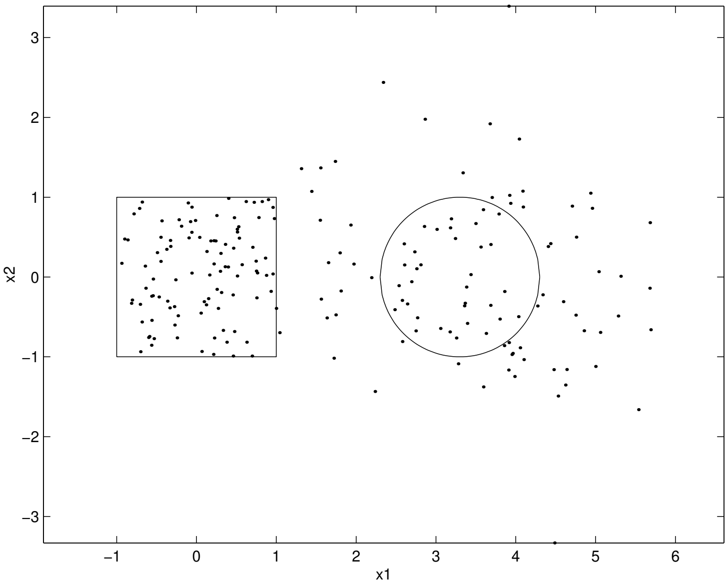


FIG. 5.8 – *Échantillon provenant d'un mélange d'une classe uniforme et d'une classe gaussienne.*

On génère de nouveau 30 échantillons de 200 points de cette distribution et on utilise les algorithmes EM et CEM de la même façon que les simulations précédentes avec le modèle gaussien $[p\lambda I]$. Les pourcentages de détection des nombres de classes sont rangés dans le tableau 5.7. On voit alors clairement que tous les critères mélange (AIC, AIC3, BIC et ICOMP) surestiment le nombre de classes tandis que tous les critères classification (NEC, CLM, CLM2, CL, CL2, C et AWE) détectent sans hésiter le bon nombre de classes dans tous les cas. Il semble donc que les critères classification soient moins sensibles que les critères mélange au fait que les données ne soient pas gaussiennes. Ceci peut s'expliquer par le fait que les critères classification font un compromis entre des classes bien gaussiennes (la qualité de la normalité est mesurée par la vraisemblance) et des classes bien séparées (la qualité de la séparation est mesurée par la classifiabilité). Dans notre exemple, les classes ne sont pas toutes gaussiennes mais la séparation est bien marquée, donc, même si l'adéquation modèle – données n'est pas très bonne, la classifiablité est correcte et compense ce manque de normalité. Ainsi, les critères de classification apportent une certaine robustesse vis à vis d'un modèle inadéquate.

K	AIC	AIC3	BIC	ICOMP	NEC	CLM(2)	CL(2)	C	AWE
1	0	0	0	0	0	0	0	0	0
2	7	27	**70**	**83**	**100**	**100**	**100**	**100**	**100**
3	0	0	0	0	0	0	0	0	0
4	**60**	**53**	30	7	0	0	0	0	0
5	33	20	0	10	0	0	0	0	0

TAB. 5.7 – *Pourcentage de choix du nombre de classes avec le modèle $[p\lambda I]$ lors d'un mélange d'une classe gaussienne et d'une classe uniforme.*

5.3 Données réelles

Cette section a pour but de confirmer, par l'utilisation de données réelles, les comportements constatés des critères précédents. En particulier, il s'agit de savoir si les critères pénalisant la vraisemblance par la classifiabilité (CL, CLM, CL2, CLM2, C) continuent à bien se comporter sur de telles données : nous avions remarqué un bon comportement de ceux-ci sur des données simulées lorsque les classes étaient bien séparées. Nous voulons aussi étudier le comportement de notre procédure de détection d'un coude de la vraisemblance par une hypothèse de précision. Cette procédure, rappelons-le, n'a pas été employée sur des données simulées car la précision devenait, dans ce cas, un terme trop subjectif.

5.3.1 Les Iris de Fisher

Les Iris de Fisher [5] sont un jeu de données aujourd'hui classique composé de 150 fleurs mesurées sur quatre variables (longueur et largeur du sépale, longueur et largeur du pétale) et reparties en trois groupes de même proportion (Iris Setosa, Iris Virginica, Iris Versicolor). Les valeurs sont reportées à l'annexe E sur le tableau E.3 et une représentation graphique est disponible sur la figure 5.9. Connaissant l'origine des individus, nous pouvons estimer les paramètres des gaussiennes. Il ressort alors que les matrices de variance ne sont pas trop différentes entre classes. Nous recherchons alors des classes avec le modèle $[\lambda C]$ à matrices de variance égales. Les proportions du modèle sont encore égales (modèle $[p\lambda C]$) puis libres (modèle $[p_k\lambda C]$). Nous calculons alors les critères précédents en initialisant EM et CEM 20 fois au hasard et en cherchant entre 1 et 5 classes (K_{sup} de Bozdogan). Remarquons qu'ici ICOMP est estimé non plus par la méthode de Cutler et Windham mais par l'approximation explicite de Bozdogan [26] que nous avions donnée à l'équation 3.11 du chapitre 3.

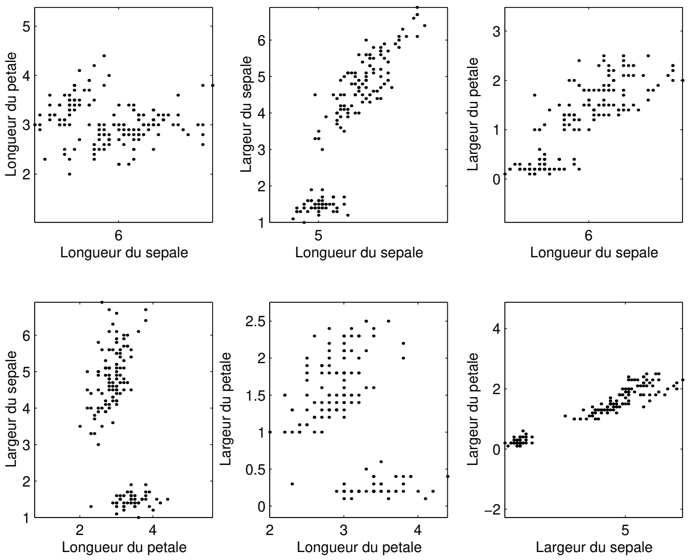


FIG. 5.9 – *Projection des Iris sur chacun des six plans.*

K	AIC3	BIC	ICOMP	NEC	CLM	CLM2	CL	CL2	AWE
				modèle à proportions égales					
1	801.8	829.9	804.6	8.2235	-379.9	-379.9	-379.9	-379.9	942.0
2	663.8	700.0	688.1	**0.0001**	-304.9	-304.9	-304.9	-304.9	844.2
3	578.7	622.9	**650.0**	0.0517	-258.8	**-258.8**	-258.5	**-258.5**	**803.8**
4	546.0	**598.2**	675.9	0.0458	**-236.2**	-	**-236.2**	-	811.2
5	**545.4**	605.7	727.4	0.1587	-239.5	-	-237.7	-	866.0
				modèle à proportions libres					
1	801.8	829.9	804.6	8.1804	-379.9	-379.9	-379.9	-379.9	942.0
2	649.8	688.0	671.8	**0.0001**	-296.4	-296.4	-296.4	-296.4	840.2
3	584.7	632.9	**649.9**	0.0514	-258.7	**-258.7**	-258.5	**-258.5**	**829.6**
4	533.0	**591.4**	655.6	0.0460	-226.1	-	-254.2	-	886.0
5	**527.5**	595.8	702.2	0.0736	**-217.9**	-	**-217.2**	-	877.0

TAB. 5.8 – *Valeurs des critères pour les Iris avec un modèle $[\lambda C]$ à proportions égales puis libres.*

Les valeurs des critères sont reportées dans le tableau 5.8 mais les critères AIC et C ne sont pas représentés : AIC a le même comportement que AIC3 et C a le même comportement que CLM. Il ressort que seuls CL2, CLM2, ICOMP et AWE détectent trois classes pour les deux situations où le modèle est à proportions égales ou est à proportions libres. Dans le cas de proportions égales uniquement, NEC trouve deux classes ; BIC, CLM et CL en trouvent quatre. L'existence d'une certaine hybridation entre les Iris Virginica et Versicolor est souvent admise et classiquement on peut diviser ce couple en un, deux ou trois groupes. Donc obtenir deux, trois ou quatre classes est cohérent. Pour le modèle à proportions libres, ICOMP, AWE, NEC, CL2, CLM2 et BIC restent dans la bonne plage. Les critères CL et CLM surestiment le nombre de composantes par contre.

On peut remarquer que lorsque l'on cherche des classes à proportions égales, on n'a plus l'équivalence CL, CL2 et CLM, CLM2 que nous avions constatée dans les simulations précédentes. Ceci est certainement la conséquence du faible nombre d'individus (150) qui permet d'obtenir plus facilement des classes de cardinal inférieur au nombre de paramètres libres.

Nous poursuivons l'étude des Iris en utilisant cette fois un modèle trop simple $[p\lambda I]$ et un modèle trop complexe $[p\lambda_k C_k]$. Les valeurs des critères sont respectivement rangées dans les tableaux 5.9 et 5.10. Dans le cas du modèle trop simple, nous remarquons le même comportement que celui constaté dans les exemples sur les données simulées : tous les critères surestiment le nombre de classes. Par contre, on constate que NEC se maintient au choix de deux classes donc est le seul critère à ne pas trouver cinq classes. Dans le cas du modèle trop complexe, tous les critères choisissent quatre classes, excepté NEC et AWE qui en trouvent deux. On conclut alors que les critères les plus stables dans le cas des Iris (c'est-à-dire les critères qui choisissent le même nombre de classes avec le modèle adapté $[p\lambda C]$ et le modèle trop complexe $[p\lambda_k C_k]$) sont les critères pénalisant la vraisemblance par la classifiabilité (CL,CLM), le critère classification NEC et le critère mélange BIC.

K	AIC3	BIC	ICOMP	NEC	CLM	CLM2	CL	CL2	AWE
1	1794.0	1804.0	1786.0	0.142	-889.5	-889.5	-889.5	-889.5	1845.0
2	1114.0	1132.0	1100.0	**0.004**	-544.1	-544.1	-544.1	-544.1	1205.8
3	847.6	873.7	820.6	0.016	-407.6	-407.6	-407.3	-407.3	984.0
4	747.6	781.8	710.4	0.024	-355.1	-355.1	-354.3	-354.3	930.2
5	**703.1**	**745.3**	**655.3**	0.022	**-325.6**	**-325.6**	**-325.3**	**-325.3**	**924.0**

TAB. 5.9 – *Valeurs des critères pour les Iris avec le modèle trop simple $[p\lambda I]$.*

La détection du nombre de classes avec la méthode cherchant un coude dans la courbe du maximum de vraisemblance a aussi été employée. Nous avons utilisé la précision $\boldsymbol{\gamma}^1 = \boldsymbol{\gamma}^2 = \boldsymbol{\gamma}^3 = \boldsymbol{\gamma}^4 = 0.05$ car les mesures sur les fleurs ont une précision de l'ordre du millimètre (voir les valeurs dans le tableau E.3 de l'annexe E). La figure 5.10 montre les courbes du maximum de vraisemblance pour chacun des quatre modèles précédents $[p\lambda C]$, $[p_k\lambda C]$, $[p\lambda I]$ et $[p\lambda_k C_k]$. Il ressort que la méthode retient quatre classes avec le modèle $[p\lambda C]$ (le modèle le plus adapté) mais qu'il s'en est fallu de peu

K	AIC3	BIC	ICOMP	NEC	CLM	CLM2	CL	CL2	AWE
1	801.8	830.0	804.6	8.2244	-379.9	-379.9	-379.9	-379.9	942.1
2	529.7	586.0	522.5	**0.0000**	-222.9	-222.9	-222.9	-222.9	**810.3**
3	487.3	571.8	472.7	0.0227	-182.4	-182.4	-182.1	-182.1	911.0
4	**-1423.8**	**-1311.3**	**-1003.1**	0.0038	**794.2**	**794.2**	**-177.0**	**-177.0**	1083.2
5	-1366.3	-1225.5	-913.9	0.0059	784.9	784.9	-182.8	-182.8	1277.0

TAB. 5.10 – *Valeurs des critères pour les Iris avec le modèle trop complexe* $[p\lambda_k C_k]$.

de détecter plus de classes. Tous autres modèles détectent plus de classes et il semble donc que notre méthode soit assez sensible au modèle retenu pour choisir le nombre de classes.

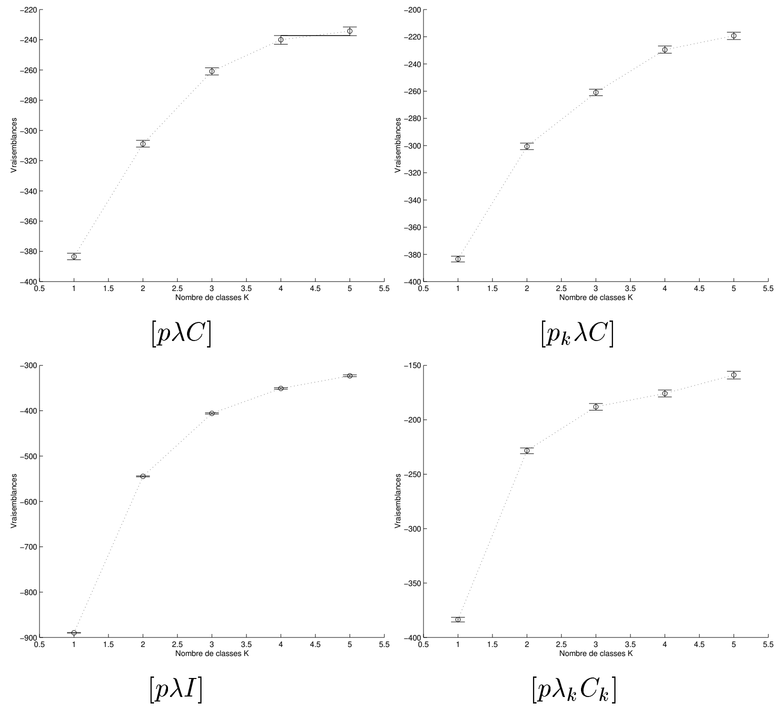


FIG. 5.10 – *Détection d'un coude de la vraisemblance pour les Iris avec les modèles* $[p\lambda C]$, $[p_k\lambda C]$, $[p\lambda I]$ *et* $[p\lambda_k C_k]$. *On utilise la précision* $\boldsymbol{\gamma}^1 = \boldsymbol{\gamma}^2 = \boldsymbol{\gamma}^3 = \boldsymbol{\gamma}^4 = 0.05$.

5.3.2 Les papillons

Poursuivons notre étude sur un jeu de données réelles de petite taille (23) en grande dimension (quatre). Il s'agit de 23 papillons (Celeux et Robert [39]) dont quatre caractéristiques sont mesurées (voir la figure E.1 pour le détail de ces caractéristiques).

L'ensemble des données sont disponibles à l'annexe E dans le tableau E.2. La figure 5.11 donne une illustration des papillons sur les six plans possibles. La bonne classification est connue et est composée de quatre classes.

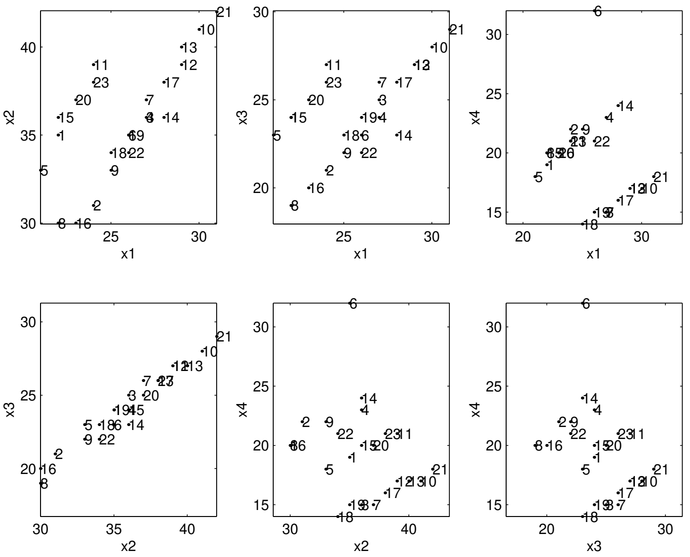
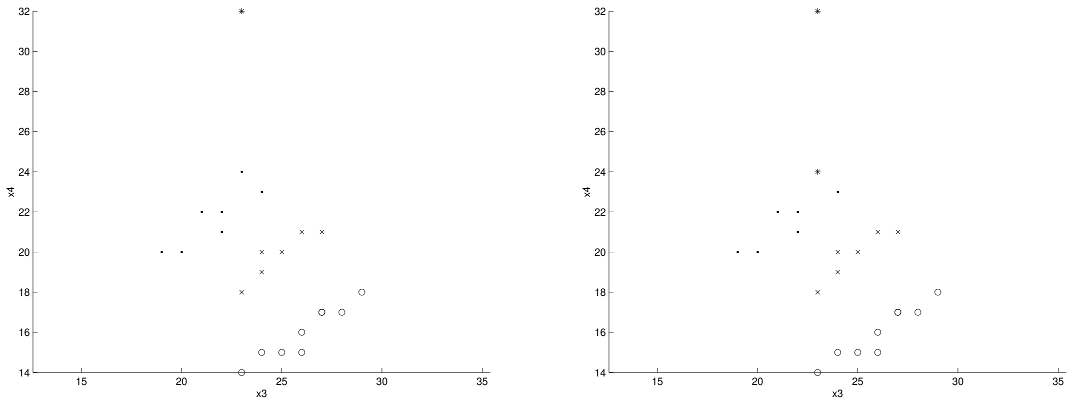


FIG. 5.11 – *Projection des papillons sur chacun des six plans.*

Comme précédemment, nous lançons EM et CEM 20 fois au hasard et nous utilisons le modèle $[p\lambda C]$. Les résultats sont stockés dans le tableau 5.11. Seuls les critères NEC et CL retiennent alors une partition en quatre classes. Les critères AIC3, BIC, CLM et CLM2 détectent cinq classes, le critère ICOMP en trouve trois et AWE est très pénalisant car il en choisit seulement deux. Lors de l'étude des données simulées, nous avions constaté un meilleur comportement de CL par rapport à CLM : cet exemple semble le confirmer. Les partitions obtenues par EM et CEM diffèrent uniquement par un seul point (le point 14) comme on peut le voir sur la figure 5.12. La bonne partition (celle des experts en papillons) correspond à celle obtenue par EM.

K	AIC3	BIC	ICOMP	NEC	CLM	CLM2	CL	CL2	AWE
1	409.3	411.2	398.5	-	-183.6	-183.6	-183.6	-183.6	497.0
2	381.2	383.6	383.8	0.000 10^{-6}	-163.6	-163.6	-163.6	-163.6	**494.1**
3	357.0	360.0	**367.2**	0.472 10^{-6}	-145.5	-145.5	-145.5	**-145.5**	495.0
4	316.4	320.0	385.2	**0.000** 10^{-6}	-119.2	-119.2	**-134.8**	-	510.6
5	**309.5**	**313.6**	426.9	0.311 10^{-6}	**-109.7**	**-109.7**	-140.5	-	559.1

TAB. 5.11 – *Valeurs des critères pour les papillons avec le modèle $[p\lambda C]$.*



partition par EM partition par CEM

FIG. 5.12 – *Partition des papillons obtenue par les algorithmes EM et CEM.*

Nous cherchons aussi le coude de la vraisemblance avec la précision $\boldsymbol{\gamma}^1 = \boldsymbol{\gamma}^2 = \boldsymbol{\gamma}^3 = \boldsymbol{\gamma}^4 = 0.5$ (les mesures des papillons se font au millimètre près d'après le tableau E.2). En utilisant encore le modèle $[p\lambda C]$, la figure 5.13 montre que l'on retient quatre classes avec une grande marge. Notre méthode est donc satisfaisante sur cet exemple.

5.3.3 The Old FaithFul Geyser

Le dernier jeu de données que nous avons utilisé regroupe 299 mesures de la durée d'éruption de «The Old Faithful Geyser», un geyser assez connu aux États-Unis (Yellowstone National Park, Wyoming). Les données proviennent de Azzalini et Bowman [6] et elles sont disponibles sur le tableau E.4 de l'annexe E. Ce geyser est réputé pour une certaine régularité dans la durée de ces éruptions : les éruptions durent généralement environ 2 minutes ou 4 minutes. Outre ces deux grandes classes d'éruptions, certains experts signalent l'existence d'une troisième petite classe à 3 minutes. La figure 5.14 donne une photo du geyser en éruption et montre les données et un histogramme de celles-ci : les deux grandes classes sont bien visibles.

Une remarque intéressante nous a été faite à ce sujet lors d'un séminaire : peut-on considérer que les 299 données sont *indépendantes*? En fait, il semble que non, puisque les éruptions alternent entre 2 et 4 minutes (2 mn puis 4 mn puis 2 mn, etc.). Conscient de ce problème, nous supposerons tout de même l'indépendance nécessaire pour nos hypothèses.

Admettant ceci, nous lançons EM et CEM 20 fois, mais en utilisant maintenant le modèle sphérique $[p\lambda I]$, puis nous calculons tous les critères. Leurs valeurs sont regroupées dans le tableau 5.12. On constate alors que tous les critères classification trouvent deux classes, tandis que tous les critères mélange en choisissent trois. La figure 5.15 indique les partitions obtenues dans le cas de deux classes et dans le cas de trois classes. Il apparaît que la partition en deux classes identifie bien les durées de deux et de quatre minutes, mais par contre la partition en trois classes ne détecte pas, comme on pourrait le penser, la classe des trois minutes : elle se contente de découper

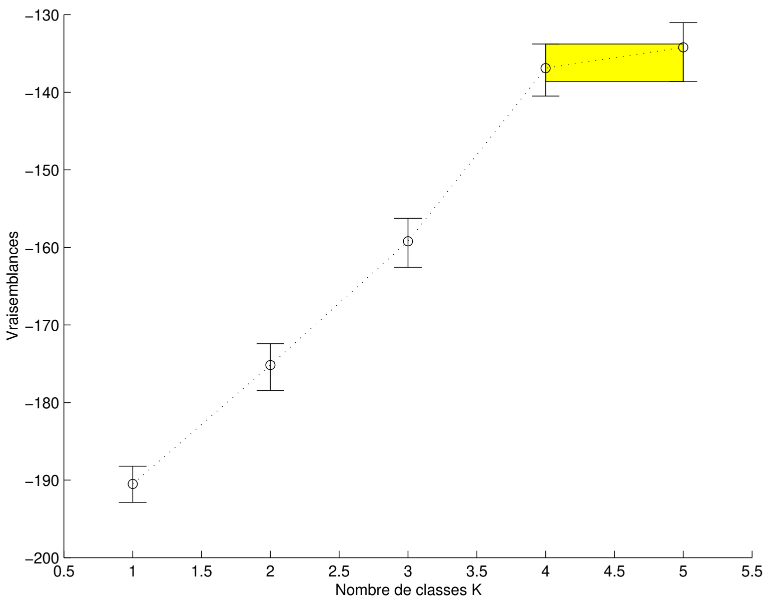


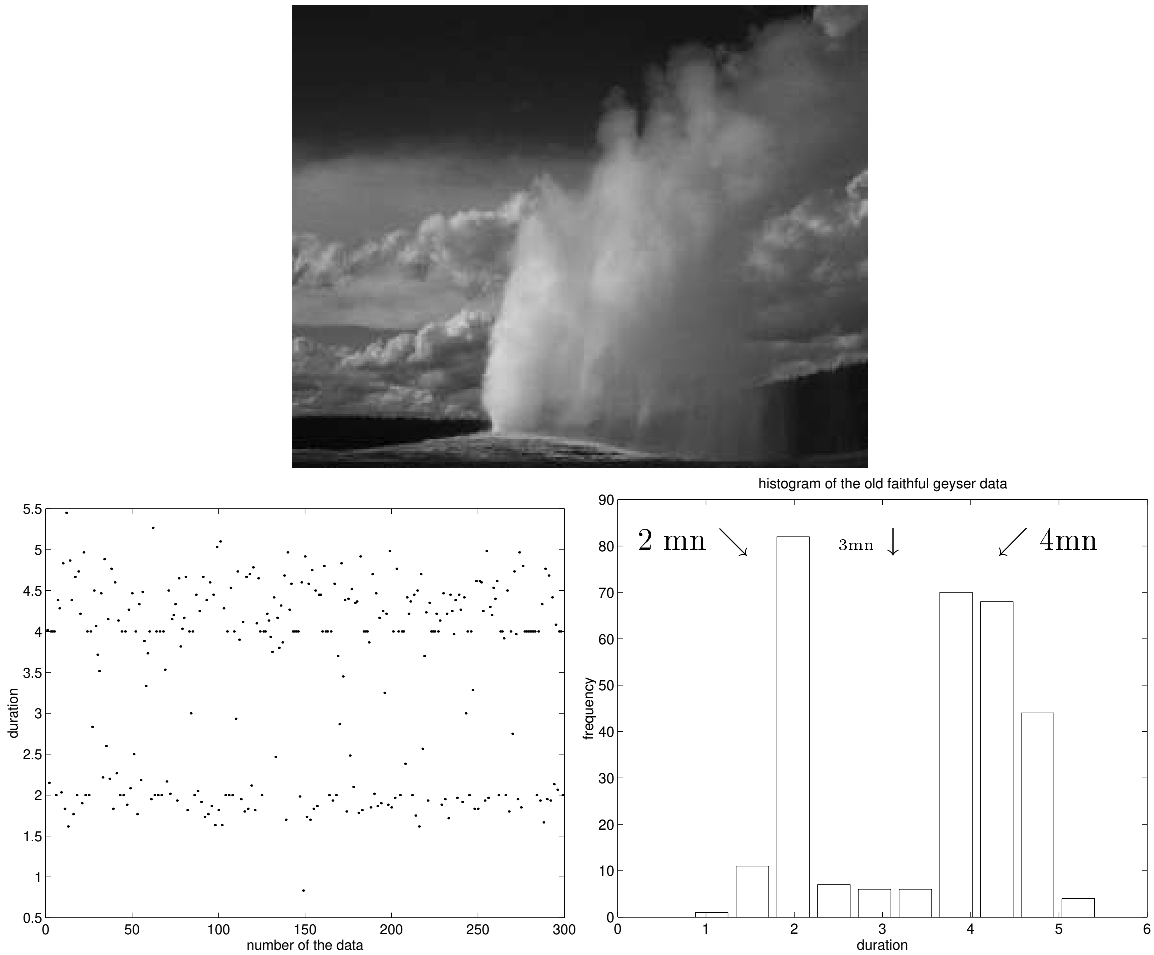
FIG. 5.13 – *Détection d'un coude de vraisemblance pour les papillons avec le modèle $[p\lambda C]$ et la précision $\boldsymbol{\gamma}^1 = \boldsymbol{\gamma}^2 = \boldsymbol{\gamma}^3 = \boldsymbol{\gamma}^4 = 0.5$.*

«artificiellement» la classe des quatre minutes en deux parties. L'explication de ce phénomène est en réalité assez simple : l'histogramme de la figure 5.14 montre bien que la classe de quatre minutes n'est pas gaussienne. Comme constaté sur les données simulées avec une classe non gaussienne, les critères mélange découpent la classe non gaussienne en plusieurs classes (deux dans le cas du geyser). Nous constatons donc de nouveau la bonne tenue des critères classification lorsque les classes sont «peu» gaussiennes mais bien séparées. D'autre part, la détection d'une troisième classe de trois minutes ne s'est pas faite car le modèle $[p\lambda I]$ recherche des classes de mêmes proportions et cette hypothétique troisième classe est vraiment peu peuplée par rapport aux deux autres.

K	AIC3	BIC	ICOMP	NEC	CLM	CLM2	CL	CL2	AWE
1	936.0	941.4	931.5	2.36	-465.0	-465.0	-465.0	-465.0	958.8
2	644.7	652.8	637.7	**0.01**	**-318.2**	**-318.2**	**-318.2**	**-318.2**	**679.8**
3	**618.3**	**629.1**	**609.3**	0.61	-358.8	-358.8	-336.9	-336.9	731.4
4	634.3	647.8	627.3	1.07	-426.9	-426.9	-348.4	-348.4	768.6
5	623.5	639.7	669.6	1.48	-501.5	-501.5	-362.5	-362.5	811.2
6	624.4	643.3	663.3	1.81	-540.6	-540.6	-384.2	-384.2	1069.2

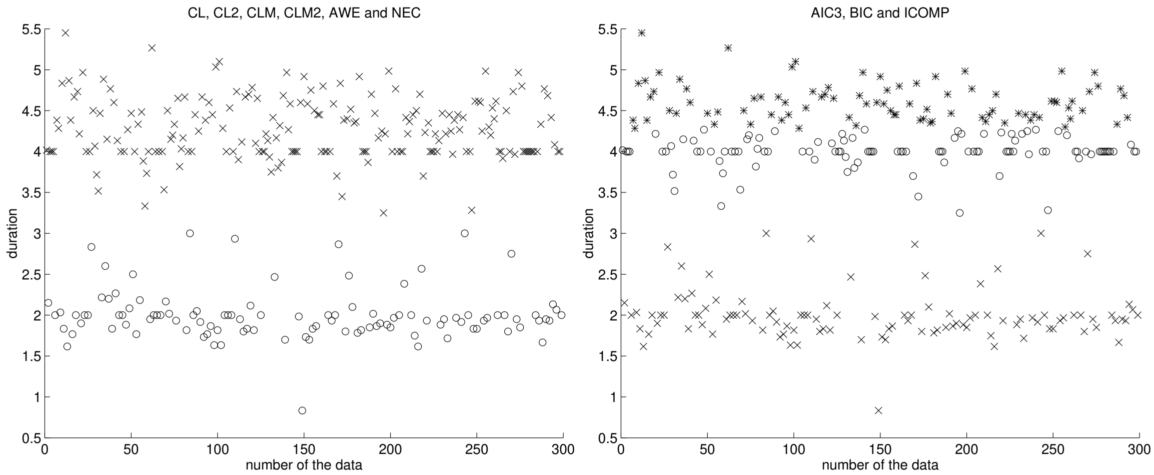
TAB. 5.12 – *Valeur des critères pour le geyser avec le modèle $[p\lambda I]$.*

Remarquons enfin que nous n'employons pas la procédure de détection de coude la vraisemblance car nous ne connaissons pas la précision des mesures (dans l'annexe E,



données histogramme

Fig. 5.14 – *Caractéristiques de* The Old Faithful Geyser.



partition en deux classes — partition en trois classes

FIG. 5.15 – *Partition en deux et en trois classes pour* The Old Faithful Geyser.

nous avons tronqué les valeurs de manière arbitraire, car nous avions 9 chiffres après la virgule !).

5.4 Conclusion

Pour trouver le nombre de classes, nous avons comparé, sur des données simulées et sur des données réelles, un certain nombre de critère décrits au chapitre 3. Nous avons alors constaté les grandes lignes suivantes :

- pour des classes peu séparées, les critères classification CL, CLM, C et AWE ont tendance à sous-estimer le nombre de classes : il faut alors des classes bien séparées pour une bonne détection. Le critère NEC, critère de classification ne pénalisant pas la vraisemblance par la classifiabilité au contraire des critères précédents, trouve des classes moins séparées ;
- l'ensemble des critères classification (CL, CLM, C, AWE et NEC) sont plus robustes que les critères mélange (AIC, AIC3, BIC et ICOMP) lors de l'utilisation d'un modèle inadapté (emploi d'un modèle gaussien alors que les classes ne sont pas gaussiennes, emploi d'un modèle à proportions égales alors que les classes ont des proportions différentes, modèle gaussien trop complexe) ;
- l'utilisation d'un modèle à proportions libres (comme $[p_k \lambda I]$) est à déconseiller avec CL et avec CLM car la surestimation du nombre de classes est excessive (c'était théoriquement prévisible au moins pour CL). Deux solutions se présentent alors : soit on utilise les critères empiriques CL2 et CLM2 car ils semblent donner de bons résultats, soit on utilise tout de même un modèle à proportions égales (comme le modèle $[p\lambda I]$) car, comme on l'a vu précédemment, les critères CL et CLM sont assez résistants à un modèle mal adapté si les classes sont bien séparées ;

- les critères classification semblent plus robustes que les critères mélange lorsque la dimension est grande (voir les exemples en dimension quatre avec les données simulées et les données réelles) : dans cette situation, les critères mélange ont tendance à surestimer le nombre de classes ;
- la procédure de détection du coude de la vraisemblance donne des résultats corrects mais semble assez sensible au choix du modèle gaussien. Plus d'expériences seraient d'ailleurs nécessaires pour approfondir son comportement.

En conclusion, concernant les critères mélange, BIC semble être le plus performant car il enraye un peu la surestimation du nombre de classes. Les critères classification semblent pourtant préférables pour les raisons évoquées précédemment. Néanmoins, NEC pose des difficultés de calcul pour le cas monoclasse, AWE est vraiment trop pénalisant et enfin CLM et C sont trop pénalisants (dans une moindre mesure que AWE tout de même). Le critère CL nous semble alors le critère le plus prometteur pour trouver le nombre de classes ; de plus sa simplicité est attrayante. Il nécessiterait néanmoins des adaptations pour les proportions libres (autres que CL2).

Conclusion générale

Bilan

Nous avons considéré la classification automatique et la discrimination dans le cadre d'une hypothèse de mélange gaussien. Notre objectif a été de proposer et de comparer des méthodes de choix de modèles : choix du modèle gaussien et choix du nombre de classes en classification automatique, choix uniquement du modèle gaussien en discrimination. L'hypothèse de mélange gaussien permettait d'utiliser un certain nombre de méthodes de choix de modèles liés à cet aspect mélange. Néanmoins, nous avons proposé des méthodes de choix de modèles qui, bien que tenant compte de l'hypothèse de mélange, considéraient aussi les objectifs propres de la classification automatique (c.-à-d. obtenir une partition des données) et de la discrimination (c.-à-d. obtenir une règle de classement).

En classification automatique, l'idée de base est de retenir le modèle (au sens général) qui produit des classes bien séparées. Nous avons traduit cette idée par un compromis entre l'adéquation modèle – données et une mesure de classifiabilité de celles-ci. Ce compromis a été réalisé en pénalisant la vraisemblance par la classifiabilité des données. Par ce procédé on retrouve, en utilisant une certaine mesure de classifiabilité, la vraisemblance classifiante de Symons. On peut aussi parfois interpréter cette vraisemblance classifiante comme une pénalisation du célèbre critère des k-means de Sebestyen. Tout d'abord, ce type de critère a été comparé aux critères classiques de type mélange pour trouver le modèle gaussien (le nombre de classes était connu). Il ressort que le critère AIC3 (critère peu justifié en théorie dans cette situation) est meilleur que notre approche. Ensuite, les critères ont été comparés pour trouver le nombre de classes (le modèle gaussien était connu) et l'intérêt des critères de classification dont ceux de type vraisemblance classifiante est, cette fois, pleinement mis en évidence. Ces critères résistent mieux à des classes non gaussiennes ou à l'utilisation de modèles inadaptés par exemple. Leur simplicité est aussi un aspect positif.

En discrimination, l'idée de base est de retenir le modèle gaussien qui fournit la plus petite erreur de classement. Cela nous conduit tout naturellement à considérer le critère classique de validation croisée. Ce critère a l'inconvénient d'être coûteux en calcul et nous avons proposé d'accélérer son calcul pour certains modèles gaussiens. Ensuite, nous avons comparé le critère de validation croisée aux critères mélange pour choisir un modèle gaussien. Il ressort que le critère de validation croisée se distingue ainsi que, de nouveau, le critère AIC3 (malgré, de nouveau, un manque de fondement théorique de ce critère dans ce cas). La validation croisée est néanmoins souvent plus robuste que AIC3 lorsqu'aucun des modèles n'est adapté aux données.

Perspectives

Dans la continuité de ce travail, nous voyons trois perspectives essentielles, toutes trois en classification automatique :

- utiliser les critères de type vraisemblance classifiante avec des mélanges non gaussiens pour trouver le nombre de classes, car rien ne s'y oppose théoriquement. Quelques essais précurseurs ont été faits dans le cadre du laboratoire avec des mélanges de Bernoulli sur des données binaires : les premiers résultats sont encourageants ;
- adapter les critères de type vraisemblance classifiante pour qu'ils trouvent le nombre de classes avec un *modèle à proportions libres* (jusqu'à présent, ces critères ne sont utilisables qu'avec des modèles à proportions égales) ;
- proposer et comparer des critères pour la détection *simultanée* du nombre de classes et du modèle de parcimonie.

Dans ce dernier cas, la plupart des critères mélange peuvent théoriquement résoudre ce problème mais nous avons vu, au moins pour le nombre de classes, l'intérêt qu'apportait la notion de classification. Nous pensons que l'optique classification est une démarche prometteuse donc à retenir pour résoudre ce difficile problème. L'adaptation de critères qui pénalisent la vraisemblance par la classifiabilité, comme la vraisemblance classifiante, est, à notre avis, une piste majeure à approfondir.

Annexe A

Le logiciel XEM

Objectif

Nous avons développé un outil informatique, le logiciel XEM, qui regroupe cinq des principales méthodes d'estimation présentées au chapitre 2 : l'algorithme EM et les quatre variantes CEM, SEM, SAEM et CAEM. Chaque algorithme a à sa disposition chacun des 28 modèles gaussiens.

Les méthodes bayésiennes ou de théorie du codage ne sont pas implantées car elles se ramènent asymptotiquement à ces algorithmes. Ce choix a été aussi motivé par la relative complexité de ces méthodes, surtout concernant la définition des lois a priori pour les modèles gaussiens. Dans l'avenir, il peut néanmoins s'avérer utile de remédier à cette lacune.

Ce logiciel a été écrit dans le langage S-PLUS (version 3.2). Actuellement la bibliothèque StatLib de S-PLUS ne propose que l'algorithme des k-means (`kmeans`) et le logiciel `mclust` qui utilise une agglomération hiérarchique avec quelques modèles gaussiens. Le logiciel SICLA de l'INRIA, écrit en Fortran, incorporait une fonction proche de XEM avec 8 modèles gaussiens sur les 14. L'avantage de S-PLUS est de fournir un environnement beaucoup plus convivial et ouvert pour traiter en amont et en aval les données relatives à XEM : graphiques, manipulation de matrices et vecteurs, fonctions de haut niveau déjà intégrées,... Nous présentons le logiciel XEM, puis nous donnons un exemple d'utilisation et enfin nous reportons le contenu du fichier d'aide associé.

Présentation du logiciel

De nombreux options et choix par défauts sont disponibles dans XEM. Une particularité de XEM est de pouvoir chaîner plusieurs algorithmes. Par exemple on peut utiliser CEM puis faire démarrer EM avec les résultats de CEM. Plusieurs méthodes d'initialisation de l'algorithme sont disponibles : centres aléatoires ou fournis par l'utilisateur, partition aléatoire ou fournie par l'utilisateur et paramètres complets fournis par l'utilisateur. Lorsque l'initialisation est aléatoire (centres ou partition aléatoires) plusieurs relances de l'ensemble de la procédure (donc tous les algorithmes chaînés) sont possibles pour éviter la dépendance vis à vis du choix initial.

La décroissance de la température dans les algorithmes SAEM et CAEM est géométrique : $\gamma^{m+1} = a\gamma^m$. La règle d'arrêt des algorithmes (excepté SEM) se fait soit par la convergence du critère $L(\boldsymbol{\theta}, \boldsymbol{c})$ (c'est-à-dire la vraisemblance ou vraisemblance classifiante en fonction de l'algorithme), soit par la stabilité de la partition. Un nombre maximum d'itérations est aussi défini. L'algorithme SEM ne s'arrêtera que lorsque ce nombre maximum sera atteint. Le début de la chaîne de Markov associée est retirée pour obtenir un estimateur ponctuel en moyennant sur le reste de la chaîne. À la sortie, sont fournis les paramètres $\boldsymbol{\theta}$ et $\boldsymbol{c}$ (partion floue ou partition), la valeur du critère et quelques informations sur le déroulement. Une vue globale de XEM est donnée à la figure A.1.

Un exemple d'utilisation

Voici un exemple d'utilisation avec les Iris d'Anderson [5].

Lancement de XEM :

```
> # define the model
> model <- list(prop="equal",vol="equal",orient="equal",shape="spherical")
> #
> # load Iris as data
> x <- rbind(iris[,,1], iris[,,2], iris[,,3])
> #
> # print data
> x
      Sepal L. Sepal W. Petal L. Petal W.
 [1,]      5.1      3.5      1.4      0.2
 [2,]      4.9      3.0      1.4      0.2
 [3,]      4.7      3.2      1.3      0.2
 [4,]      4.6      3.1      1.5      0.2
...
[147,]      6.3      2.5      5.0      1.9
[148,]      6.5      3.0      5.2      2.0
[149,]      6.2      3.4      5.4      2.3
[150,]      5.9      3.0      5.1      1.8
> #
> # run XEM with SAEM, EM then CEM, three wanted clusters and two runs of XEM
> XEM(x, K=3, model=model, tab.algo=c("SAEM","EM","CEM"), nb.XEM=2)
```

Détail des messages affichés à l'écran pendant l'exécution :

```
---- begin XEM no 1 ----

begin INIT
end   INIT

begin SAEM
end   SAEM

begin EM
end   EM

begin CEM
end   CEM

---- end   XEM no 1 ----

---- begin XEM no 2 ----

begin INIT
end   INIT
```

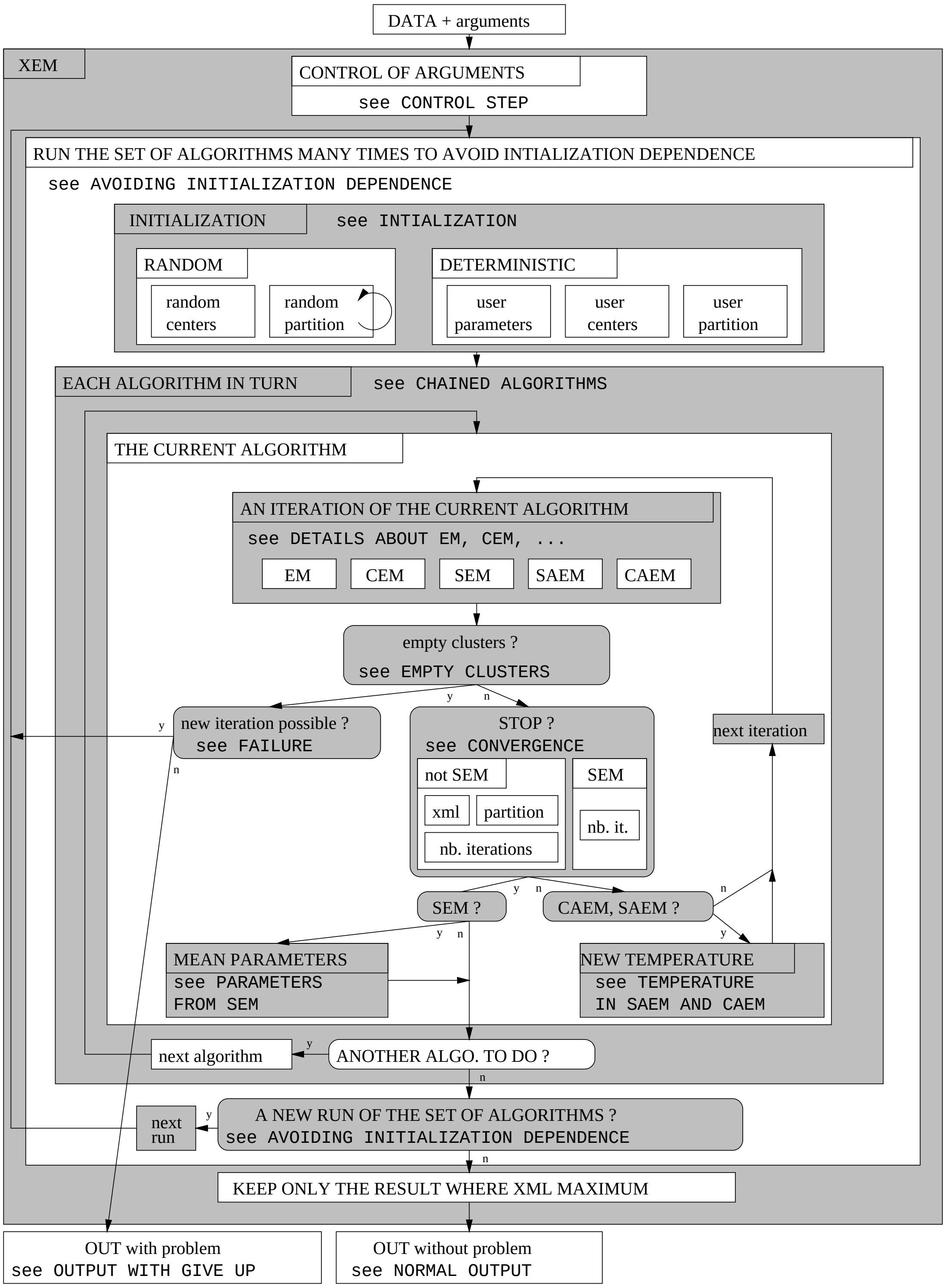


FIG. A.1 – *Algorithme du logiciel XEM. Le texte de type courrier comme* `see INITIALIZATION` *renvoie à la partie correspondante de l'aide détaillée S-PLUS.*

```
begin SAEM
empty clusters: restart XEM
end   SAEM

begin INIT
end   INIT

begin SAEM
end   SAEM

begin EM
end   EM

begin CEM
end   CEM

---- end   XEM no 2 ----
```

Sortie (l'échec n'est pas reporté car il n'est pas arrivé dans la meilleure itération) :

```
$give.up:
[1] F

$nb.failure:
[1] 0

$algo.done:
[1] "SAEM" "EM"   "CEM"

$p:
     cl. 1     cl. 2     cl. 3
 0.3333333 0.3333333 0.3333333

$mu:
      Sepal L. Sepal W. Petal L. Petal W.
cl. 1 6.853846 3.076923 5.715385 2.053846
cl. 2 5.883607 2.740984 4.388525 1.434426
cl. 3 5.006000 3.428000 1.462000 0.246000

$S:

, , cl. 1
          Sepal L.  Sepal W.  Petal L.  Petal W.
Sepal L. 0.1314261 0.0000000 0.0000000 0.0000000
Sepal W. 0.0000000 0.1314261 0.0000000 0.0000000
Petal L. 0.0000000 0.0000000 0.1314261 0.0000000
Petal W. 0.0000000 0.0000000 0.0000000 0.1314261

, , cl. 2
          Sepal L.  Sepal W.  Petal L.  Petal W.
Sepal L. 0.1314261 0.0000000 0.0000000 0.0000000
Sepal W. 0.0000000 0.1314261 0.0000000 0.0000000
Petal L. 0.0000000 0.0000000 0.1314261 0.0000000
Petal W. 0.0000000 0.0000000 0.0000000 0.1314261

, , cl. 3
          Sepal L.  Sepal W.  Petal L.  Petal W.
Sepal L. 0.1314261 0.0000000 0.0000000 0.0000000
Sepal W. 0.0000000 0.1314261 0.0000000 0.0000000
Petal L. 0.0000000 0.0000000 0.1314261 0.0000000
Petal W. 0.0000000 0.0000000 0.0000000 0.1314261

$cik:
```

```
      cl. 1 cl. 2 cl. 3
 [1,]     0     0     1
 [2,]     0     0     1
 [3,]     0     0     1
 [4,]     0     0     1
...
[147,]     0     1     0
[148,]     1     0     0
[149,]     1     0     0
[150,]     0     1     0

$xml:
[1] -407.3618
```

Fichier d'aide du logiciel

Nous reportons ici le fichier d'aide associé au logiciel XEM. Ce fichier a un format propre à S-PLUS. Il détaille l'ensemble des caractéristiques nécessaires à l'utilisation de XEM : entrées (obligatoires ou optionnelles), sorties, détails, références.

```
EM Algorithm and Variants


DESCRIPTION:
       Contains the iterative procedure  EM  and  four  variants:
       CEM,  SEM,  SAEM,  CAEM.  These algorithms can be chained.
       Different  initializations,  convergence  types  and  many
       other  features  can  be controled. 28 parsimonious models
       are  available  (constraints  on  mixing  proportions  and
       variance matrices).

USAGE:
       XEM(x, K, model, tab.algo="EM", p=NULL, mu=NULL, S=NULL,
       cik=NULL, tab.max.iter, tab.eps,tab.cut.SEM, tab.a,
       max.failure=10, d=dim(x)[2], n=dim(x)[1],
       cst=1/(2 * pi)^(d/2), mu.free=T, tiny.cluster=d + 1,
       type.init="random centers", max.trials.init=10,
       tab.crit.cvge, nb.XEM=10)

REQUIRED ARGUMENTS:
x:      multivariate data matrix (n x d).
K:     number of wanted clusters.  Default  value  is  the  first
       dimension  of  mu,  if  mu  is  supplied,  or  the  second
       dimension of cik, if cik is supplied.

OPTIONAL ARGUMENTS:
model:    a list with the model properties (see DETAILS):
      prop: proportions "equal" or "free"
       vol: volume "equal" or "free"
    orient: orientation "equal", "axis" or "free"
```

```
     shape: shape "spherical", "equal" or "free".
       The  default   values   are   prop="equal",   vol="equal",
       orient="equal" and shape="spherical".
tab.algo:   vector of algorithms to chain.  Possible  values  are
       "EM",  "CEM",  "SEM",  "CAEM" and "SAEM". Default value is
       "EM".
p:     vector of mixing proportions (K). See DETAILS for  default
       values.
mu:    matrix of means (K x d). See DETAILS for default values.
S:     array of variance matrices (d x d x K).  See  DETAILS  for
       default values.
cik:   matrix of (maybe fuzzy) partition (n x K). See DETAILS for
       default values.
tab.max.iter:    vector of the maximum number of  iterations  for
       each  algorithm  in  turn.  Must  be of the same length as
       tab.algo. Default  vector  contains  the  value  150.  See
       DETAILS.
tab.eps:    vector of the xml precision to  test  convergence  of
       each  algorithm  in  turn.  Must  be of the same length as
       tab.algo. Default vector contains  the  value  0.001.  See
       DETAILS.
tab.cut.SEM:   vector of the number of first parameters values to
       drop  when  SEM  is  used.   Must be of the same length as
       tab.algo, even if other  algorithms  than  SEM  are  used.
       Default vector contains the value 50. See DETAILS.
tab.a:     vector  of  linear  coefficients   to   decrease   the
       temperature  in  "SAEM"  and  "CAEM".  Must be of the same
       length as tab.algo.  Default  vector  contains  the  value
       0.95. See DETAILS.
max.failure:   maximum number of failures (empty clusters)  where
       to  restart  XEM  is  allowed.  Default  value  is 10. See
       DETAILS.
d:     space dimension. Default value is the second dimension  of
       x.
n:     sample size. Default value is the first dimension of x.
cst:    constant  term  for  normalization.  Default   value   is
       1/(2*pi)^(d/2).
mu.free:    logical flag for centers to be free  (TRUE)  or   not
       (FALSE).  In  this  last  case,  centers  are equal to the
       sample mean. See DETAILS.
tiny.cluster:    minimum number of data in a  cluster.  When  the
       population   of  a  cluster  reaches  this  number  in  an
       iteration, then a new XEM algorithm is started  if  it  is
       allowed. Default value is (d+1). See DETAILS.
type.init:     wanted initialization. Default  value  is  "random
```

```
       centers",   other   values   are   "user  centers",  "user
       parameters", "user partition" and "random partition".  See
       DETAILS.
max.trials.init:      maximum   number   of   trials   in   the
       initialization step to choose an acceptable (not too empty
       clusters) partition when type.init is "random  partition".
       See DETAILS.
tab.crit.cvge:   vector of the convergence criterion use in  each
       algorithm  in turn. It can be the "xml" criterion (default
       value) or the stability of "partition". Only the iteration
       given the best value of xml is kept. See DETAILS.
nb.XEM:   number of times to run the entire  procedure  to  avoid
       dependence  of initialization step. If K=1 or if type.init
       is deterministic ("user parameters", "user centers", "user
       partition"),  this  value  is  fixed  to  1, otherwise its
       default value is 10. See DETAILS.

VALUE:
       returns  a  list  of  many  arguments  (flag  of  good
       convergence, parameters, ... See DETAILS):
   give.up: logical flag indicating the convergence status of XEM
            (empty  clusters). It is equal to F if convergence is
            present and is equal to T otherwise.
nb.failure: number of failures (restart XEM when  empty  cluster)
            before the end.
 algo.done: vector given the list of done algorithms.
         p: vector of mixing proportions (K). NULL if give.up  is
            TRUE.
        mu:  matrix of means (K x d). NULL if give.up is TRUE.
         S:  array of variance matrices (d x  d  x  K).  NULL  if
            give.up is TRUE.
       cik: matrix of (maybe fuzzy) partition (n x  K).  NULL  if
            give.up is TRUE.
       xml: value of the criterion at the  convergence.  -Inf  if
            give.up is TRUE.

SIDE EFFECTS:
       None.

DETAILS:
       INITIALIZATION

       Two main types of  initialization  are  available:  random
       initializations ("random centers", "random partition") and
       deterministic ones ("user partition",  "user  parameters",
```

```
"user centers"). Type.init contains such a choice. Default
value is "random centers".

Random centers:  K different centers are choosen at random
among  data  x.  K  has  to be supplied. p is forced to be
equal between clusters and S to  be  spherical  and  equal
between clusters.

Random partition:  a partition cik is choosen  at  random.
If a cluster is empty (see EMPTY CLUSTERS), a new trial is
done. Maximum number of trials is max.trials.init. If this
number is reached, we leave XEM (see OUTPUT WHEN GIVE UP).
K has to be supplied.

User  partition:   a  (maybe  fuzzy)  partition   cik   is
supplied.  K  is  deduced  from cik. If a cluster is empty
(see EMPTY CLUSTERS), we leave XEM (see OUTPUT  WHEN  GIVE
UP).

User parameters:  all parameters (p, mu, S)  are  supplied
and K is deduced from mu.

User centers:  only mu is supplied and K is  deduced  from
it.  p  is forced to be equal between clusters and S to be
spherical and equal between clusters.


EMPTY CLUSTERS

A cluster is said empty if the size of the fuzzy partition
is  less  or  equal  then  tiny.cluster.  Default value of
tiny.cluster is (d+1).

FAILURE

When an empty cluster happens, we call that a failure  and
XEM  runs  a  new  initialization  if  possible, otherwise
leaves XEM (see OUTPUT WHEN GIVE UP). A new initialization
is  possible  only  if the number of failures is less than
max.failure and if it was  a  random  initialization  (see
INITIALIZATION) step and if K is greater than 1.

OUTPUT WHEN GIVE UP

If an empty cluster occurs in XEM (see EMPTY CLUSTERS) and
```

permission to restart XEM is denied (see INITIALIZATION and FAILURE), then we give up by leaving XEM. The output is a list with the following components: give.up=T, nb.failure=<<number of failures before leaving the current XEM>>, algo.done=<<vector of the names of algorithms ("EM", ...) which are done before leaving>>, p=NULL, mu=NULL, S=NULL, cik=NULL, xml=-Inf.

NORMAL OUTPUT

When no fatal problem occurs in XEM (see OUTPUT WHEN GIVE UP), it returns a list with the following components (features of the best trial: see AVOIDING INITIALIZATION DEPENDENCE): give.up=F, nb.failure=<<number of failures in the best trial (see FAILURE)>>, algo.done=<<vector of the names of algorithms ("EM", ...) which are done in the best trial>>, p, mu, S, cik, xml.

AVOIDING INTITIALIZATION DEPENDENCE

We can run the body of XEM (initialization step and then a sequence of maybe several basic algorithms as "EM", "CEM", etc, see CHAINED ALGORITHMS) several times and just keep the best run (the one which maximizes xml). The number of loops is controled by nb.XEM. Its default value is 10. But only ONE loop is possible if we search just one cluster (K=1) or if the initialization step is deterministic (see INITIALIZATION).

CHAINED ALGORITHMS

After the initialization step (see INITIALIZATION), we can chain one until many basic algorithms ("EM", "CEM", "SEM", "SAEM", "CAEM"). Such a chain is contained in tab.algo. When the first algorithm is finished (without giving up, see EMPTY CLUSTERS and FAILURE), it means that it arrived to convergence (see CONVERGENCE), the next algorithm is started with parameters of this previous one (see the special case of SEM in PARAMETERS FROM SEM), and so on until the last algorithm of the list. For instance, we can chain CEM and then EM.

CONVERGENCE

Convergence of SEM and of all other algorithms is

```
different:

SEM: we only stop when the current number of maximum
iterations in tab.max.iter is reached. There is no
dependence on tab.crit.cvge and tab.eps.

EM, CEM, SAEM, CAEM: two types of convergence are
available through tab.crit.cvge: "xml" or "partition". The
first one ("xml") stops the algorithm if the xml criterion
no longer increases as regards current value in tab.eps.
The second one ("partition") stops it when the partition
is stable. In both cases, if no convergence happens when
the current value of maximum iterations in tab.max.iter is
reached, the algorithm is even so stopped.


PARAMETERS FROM SEM

SEM gives as result a Markov chain of parameters (p, mu,
S) and of length the current value in tab.max.iter. So we
take as punctual estimators of these parameters the mean
of a part of this chain: values between current value of
tab.cut.SEM until current value of tab.max.iter.

NAMES OF DATA USED IN PARAMETERS

If data x have names on variables or on individuals, such
names are then used for parameters p, mu, S, cik. Clusters
are called cl. 1, cl. 2, and so on.

TEMPERATURE IN SAEM AND CAEM

SAEM and CAEM need both a decreasing function of
temperature. Temperature starts from 1 and linearly
decreases with the current linear coefficient in tab.a.
This current value must be between 0 and 1.

CONTROL STEP

At the very beginning of XEM, we control if some arguments
are correct and we may give some default values if
necessary. This step is only run once.

DETAILS ABOUT EM, CEM, SEM, SAEM, CAEM
```

See MODELS. See REFERENCES.

MODELS

A model corresponds to constraints on mixing proportions and variance matrices (see REFERENCES). 28 different models are available by combination. Models are used in the initialization step with random partition and user partition, and in the maximisation step of the algorithms too. The default model has equal mixing proportions between clusters, and spherical and equal variance matrices between clusters.

MU FREE OR NOT

mu.free is a flag indicating if the centers are free (T) or not (F), it means they are all equal to the mean of the sample. Such an argument is seldom used except in special cases as the calculus of the NEC criterion in the monocluster situation (model selection level).

REFERENCES:

G. CELEUX and G. GOVAERT. Gaussian Parsimonious Models. Pattern Recognition, 28(5):781-73, 1995.

A.P. DEMPSTER, N.M. LAIRD, and D.B. RUBIN. Maximum Likelikood from Incomplete Data with the EM Algorithm (with discussion). Journal of the Royal Statistical Society, Series B, 39:1-38, 1977.

G. CELEUX and G. GOVAERT. A classification EM algorithm for clustering and two clustering versions. Computational Statistics & Data Analysis, 14:315-332, 1992.

G. CELEUX and J. DIEBOLT. The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. Compu. Stat. Quater., 2(1):73-92, 1985.

G. CELEUX and J. DIEBOLT. Une version de type recuit simule de l'algorithme EM. Notes aux Comptes Rendus de l'Academie des Sciences, 310:119-124, 1990.

```
SEE ALSO:
       kmeans, mclust

EXAMPLES:
       model <- list(prop="equal", vol="equal", orient="equal",
        shape="equal") # fitted model for Iris
       x <- rbind(iris[,,1], iris[,,2], iris[,,3]) # Iris data
       XEM(x,model=model,K=3,tab.algo="CEM")
```

Annexe B

Le critère AIC

Nous reprenons ici les principales étapes de construction des critères AIC et AIC3. Le critère AIC a été introduit dans un contexte de mélange par Bozdogan [23] en s'appuyant sur les travaux d'Akaike [2] [3]. Le critère AIC3 est une version de AIC proposée par Bozdogan [21] [25] pour améliorer la performance du critère d'AIC dans le cas du choix du nombre de classes en mélange.

Construction de AIC

Hypothèse sur les données et les modèles Nous disposons d'un échantillon $\mathcal{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ de $\mathbb{R}^d$. Chaque donnée $\boldsymbol{x}_i$ $(1 \leq i \leq n)$ est une réalisation *i.i.d.* d'une variable aléatoire $\boldsymbol{X}_i$ de densité $f(\boldsymbol{x}_i|\boldsymbol{\theta}^*)$ avec $\boldsymbol{\theta}^* \in \mathbb{R}^\kappa$. Nous supposons avoir à choisir entre les κ modèles suivants :

$$\mathcal{M}_k = \{f(\boldsymbol{x}|\boldsymbol{\theta}_k), \boldsymbol{\theta}_k = (\theta_1, \theta_2, \ldots, \theta_k, 0, \ldots, 0)\} \quad 1 \leq k \leq \kappa$$

avec $\boldsymbol{\theta}_k$ paramètres du sous-espace $E_k \subset \mathbb{R}^k$. Ainsi, chaque modèle $\mathcal{M}_k$ a k paramètres libres et est une restriction du modèle $\mathcal{M}_{k+1}$, $\mathcal{M}_\kappa$ étant alors le modèle le plus général.

Le coût entropique L'opposée de l'information de Kullback-Leibler entre les deux densités $f(\boldsymbol{x}|\boldsymbol{\theta}^*)$ et $f(\boldsymbol{x}|\boldsymbol{\theta})$ s'écrit (voir relation 1.10)

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\theta}^*) = \mathrm{KL}(\boldsymbol{\theta}|\boldsymbol{\theta}^*) = \int \ln\left[\frac{f(\boldsymbol{x}|\boldsymbol{\theta}^*)}{f(\boldsymbol{x}|\boldsymbol{\theta})}\right] f(\boldsymbol{x}|\boldsymbol{\theta}^*)d\boldsymbol{x}. \tag{B.1}$$

Bozdogan propose d'utiliser cette mesure comme le coût de choisir le paramètre $\boldsymbol{\theta}$ pour $f(\boldsymbol{x}|\boldsymbol{\theta})$ alors que le vrai paramètre est $\boldsymbol{\theta}^*$. Ce coût est généralement appelé *coût entropique*. Remarquons que cette quantité est nulle si et seulement si $f(\boldsymbol{x}|\boldsymbol{\theta}) = f(\boldsymbol{x}|\boldsymbol{\theta}^*)$, et strictement positive sinon. Elle peut donc bien représenter un coût. Les variables aléatoires $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ étant *i.i.d.*, le coût $\mathcal{L}_n(\boldsymbol{\theta}|\boldsymbol{\theta}^*)$ entre les densités $f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n|\boldsymbol{\theta}^*)$ et $f(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n|\boldsymbol{\theta})$ s'écrit

$$\mathcal{L}_n(\boldsymbol{\theta}|\boldsymbol{\theta}^*) = n\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\theta}^*).$$

Choix d'un modèle par le risque moyen Le risque conditionnel associé au coût entropique $\mathcal{L}_n(\boldsymbol{\theta}|\boldsymbol{\theta}^*)$ et à une loi a posteriori $\pi(\boldsymbol{\theta}^*|\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n)$ sur $\boldsymbol{\theta}^*$ s'énonce

$$\int \mathcal{L}_n(\boldsymbol{\theta}|\boldsymbol{\theta}^*)\pi(\boldsymbol{\theta}^*|\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n)d\boldsymbol{\theta}^*.$$

La vraie valeur $\boldsymbol{\theta}^*$ est indépendante des données donc $\pi(\boldsymbol{\theta}^*) = \pi(\boldsymbol{\theta}^*|\mathcal{X})$. De plus, la valeur de $\boldsymbol{\theta}^*$ étant inconnue mais déterministe, il est possible de poser que sa distribution a priori $\pi(\boldsymbol{\theta}^*)$ est égale à la distribution de Dirac en $\boldsymbol{\theta}^*$. Dans ce cas, le coût $\mathcal{L}_n(\boldsymbol{\theta}|\boldsymbol{\theta}^*)$ représente aussi le risque conditionnel.

La valeur de $\boldsymbol{\theta}$ peut dépendre de l'échantillon, donc $\boldsymbol{\theta} = \boldsymbol{\theta}(\mathcal{X})$. Par abus de langage, mais pour ne pas alourdir les notations, nous noterons $\boldsymbol{\theta}$ aussi bien la fonction que la valeur pour un échantillon $\mathcal{X}$. Le risque moyen s'écrit alors

$$\begin{aligned} \mathcal{R}(\boldsymbol{\theta}) &= \int \mathcal{L}_n(\boldsymbol{\theta}|\boldsymbol{\theta}^*)f(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n|\boldsymbol{\theta}^*)d\boldsymbol{x}_1\ldots d\boldsymbol{x}_n \\ &= n\int \mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\theta}^*)f(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n|\boldsymbol{\theta}^*)d\boldsymbol{x}_1\ldots d\boldsymbol{x}_n \\ &= n\mathrm{E}[\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\theta}^*)]. \end{aligned} \tag{B.2}$$

Akaike et Bozdogan proposent de retenir le modèle $\mathcal{M}_k$ qui minimise le risque moyen $\mathcal{R}(\hat{\boldsymbol{\theta}}_k)$, avec $\hat{\boldsymbol{\theta}}_k$ l'estimateur du maximum de vraisemblance du paramètre de la densité f sous les contraintes de ce modèle. Cette quantité n'est pas directement calculable et nécessite d'être estimée. Nous allons voir que le critère AIC est un estimateur de 2 fois ce terme:

$$\mathrm{AIC}(\mathcal{M}_k) = 2\hat{\mathcal{R}}(\hat{\boldsymbol{\theta}}_k) + \mathrm{cste}.$$

On peut aussi remarquer que

$$\begin{aligned} \mathcal{L}_n(\hat{\boldsymbol{\theta}}_k|\boldsymbol{\theta}^*) &= -n\int \ln[f(\boldsymbol{x}|\hat{\boldsymbol{\theta}}_k)]f(\boldsymbol{x}|\boldsymbol{\theta}^*)d\boldsymbol{x} + \mathrm{cste} \\ &= -\mathrm{E}_X[\mathrm{L}(\hat{\boldsymbol{\theta}}_k)] + \mathrm{cste}, \end{aligned}$$

donc que

$$\begin{aligned} \mathcal{R}(\hat{\boldsymbol{\theta}}_k) &= -\int \mathrm{E}_X[\mathrm{L}(\hat{\boldsymbol{\theta}}_k(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n))]f(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n|\boldsymbol{\theta}^*)d\boldsymbol{x}_1\ldots d\boldsymbol{x}_n + \mathrm{cste} \\ &= -\mathbf{E}[\mathrm{E}_X[\mathrm{L}(\hat{\boldsymbol{\theta}}_k)]] + \mathrm{cste}. \end{aligned}$$

Ainsi AIC peut également être interprété comme un estimateur de l'espérance moyenne de la log-vraisemblance.

Estimation du risque moyen Le développement de Taylor à l'ordre 2 du coût $\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\theta}^*)$ au voisinage de $\boldsymbol{\theta}^*$ s'écrit

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\theta}^*) \simeq \underbrace{\mathcal{L}(\boldsymbol{\theta}^*|\boldsymbol{\theta}^*)}_{0} + (\boldsymbol{\theta}-\boldsymbol{\theta}^*)\underbrace{\nabla\mathcal{L}(\boldsymbol{\theta}^*|\boldsymbol{\theta}^*)}_{0 \text{ car optimum}} + \frac{1}{2}(\boldsymbol{\theta}-\boldsymbol{\theta}^*)'\boldsymbol{F}(\boldsymbol{\theta}^*)(\boldsymbol{\theta}-\boldsymbol{\theta}^*)$$

avec $\boldsymbol{F}(\boldsymbol{\theta}^*)$ matrice d'information de Fisher $(\kappa\times\kappa)$ pour un seul individu qui correspond à la matrice hessienne de $\mathcal{L}(\boldsymbol{\theta}^*|\boldsymbol{\theta}^*)$. Si la densité f a son paramètre $\boldsymbol{\theta}$ indépendant de son support, on a alors

$$\boldsymbol{F}(\boldsymbol{\theta}^*) = \mathrm{E}_X\left\{\left[\frac{\partial}{\partial\boldsymbol{\theta}}\ln f(\boldsymbol{X}|\boldsymbol{\theta})\right]\left[\frac{\partial}{\partial\boldsymbol{\theta}}\ln f(\boldsymbol{X}|\boldsymbol{\theta})\right]'\Bigg|_{\boldsymbol{\theta}^*}\right\}.$$

Les deux premiers termes du développement étant nuls, on pose

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{\theta}^*) \simeq \frac{1}{2}\|\boldsymbol{\theta}-\boldsymbol{\theta}^*\|_F^2 \tag{B.3}$$

avec $\|\boldsymbol{\theta}-\boldsymbol{\theta}^*\|_F^2 = (\boldsymbol{\theta}-\boldsymbol{\theta}^*)'\boldsymbol{F}(\boldsymbol{\theta}^*)(\boldsymbol{\theta}-\boldsymbol{\theta}^*)$.

Notons $\boldsymbol{\theta}_k^*$ la projection de $\boldsymbol{\theta}^*$ sur le sous-espace E_k et $\hat{\boldsymbol{\theta}}_k$ l'estimateur du maximum de vraisemblance de $\boldsymbol{\theta}_k^*$ dans E_k. D'après B.3 nous avons

$$\begin{aligned} 2\mathcal{L}(\hat{\boldsymbol{\theta}}_k|\boldsymbol{\theta}^*) &\simeq \|\boldsymbol{\theta}^*-\hat{\boldsymbol{\theta}}_k\|_F^2 \\ &\simeq \|\boldsymbol{\theta}^*-\boldsymbol{\theta}_k^*\|_F^2 + \|\boldsymbol{\theta}_k^*-\hat{\boldsymbol{\theta}}_k\|_F^2 \end{aligned}$$

en utilisant le théorème de Pythagore. Se servant ensuite de B.2, on obtient

$$\begin{aligned} 2\mathcal{R}(\hat{\boldsymbol{\theta}}_k) &= n\mathbf{E}[2\mathcal{L}(\hat{\boldsymbol{\theta}}_k|\boldsymbol{\theta}^*)] \\ &\simeq n\mathbf{E}[\|\boldsymbol{\theta}^*-\boldsymbol{\theta}_k^*\|_F^2 + \|\boldsymbol{\theta}_k^*-\hat{\boldsymbol{\theta}}_k\|_F^2] \\ &\simeq n\|\boldsymbol{\theta}^*-\boldsymbol{\theta}_k^*\|_F^2 + \mathbf{E}[\|n^{1/2}(\boldsymbol{\theta}_k^*-\hat{\boldsymbol{\theta}}_k)\|_F^2] \\ &\simeq \delta + \mathbf{E}[\|n^{1/2}(\boldsymbol{\theta}_k^*-\hat{\boldsymbol{\theta}}_k)\|_F^2] \end{aligned}$$

avec $\delta = n\|\boldsymbol{\theta}^*-\boldsymbol{\theta}_k^*\|_F^2$. Comme le remarque Bozdogan, le terme δ correspond au *biais* introduit par la projection de $\boldsymbol{\theta}^*$ sur E_k et $\mathbf{E}[\|n^{1/2}(\boldsymbol{\theta}_k^*-\hat{\boldsymbol{\theta}}_k)\|_F^2]$ est une mesure de la *variance* de l'erreur aléatoire $(\boldsymbol{\theta}_k^*-\hat{\boldsymbol{\theta}}_k)$.

Il est bien connu que l'estimateur du maximum de vraisemblance $\hat{\boldsymbol{\theta}}_k$ suit asymptotiquement $(n\to\infty)$ une loi normale de moyenne $\boldsymbol{\theta}_k^*$ et de matrice de variance l'inverse de la matrice d'information de Fisher (pour l'échantillon complet) soit $(n\boldsymbol{F}(\boldsymbol{\theta}^*))^{-1}$. Ainsi, asymptotiquement on a :

$$\|n^{1/2}(\boldsymbol{\theta}_k^*-\hat{\boldsymbol{\theta}}_k)\|_F^2 \sim \chi_k^2$$

avec χ_k^2 la distribution du chi-deux à k degrés de liberté. Utilisant le fait que $\mathrm{E}[\chi_k^2]=k$ on obtient alors

$$2\mathcal{R}(\hat{\boldsymbol{\theta}}_k) \simeq \delta + k. \tag{B.4}$$

Il reste alors à estimer le biais δ qui est *inconnu* mais *déterministe*. Dans ce but, nous utilisons le rapport du maximum de vraisemblance suivant :

$$\begin{aligned} -2\ln r(\mathcal{X}) &= -2\ln\frac{f(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n|\hat{\boldsymbol{\theta}}_k)}{f(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n|\hat{\boldsymbol{\theta}}_\kappa)} \\ &= -2(\mathrm{L}(\hat{\boldsymbol{\theta}}_k)-\mathrm{L}(\hat{\boldsymbol{\theta}}_\kappa)). \end{aligned}$$

Sous l'hypothèse alternative que $\boldsymbol{\theta}^* = \hat{\boldsymbol{\theta}}_\kappa$, Wald [88] a montré que $-2\ln r(\boldsymbol{X}_1, \dots, \boldsymbol{X}_n)$ a pour distribution asymptotique $\chi^2_{\kappa-k}(\delta)$, un chi-deux décentré avec $(\kappa - k)$ degrés de liberté et paramètre de décentrage $\delta = \|n^{1/2}(\boldsymbol{\theta}^* - \boldsymbol{\theta}^*_k)\|^2_F$:

$$-2\ln r(\boldsymbol{X}_1, \dots, \boldsymbol{X}_n) \sim \chi^2_{\kappa-k}(\delta) \;\text{ asymptotiquement.} \tag{B.5}$$

En utilisant le fait que $\mathrm{E}[\chi^2_{\kappa-k}(\delta)] = \delta + \kappa - k$ puis en faisant l'approximation

$$\begin{aligned} -2\ln r(\mathcal{X}) &\simeq \mathrm{E}[-2\ln r(\boldsymbol{X}_1, \dots, \boldsymbol{X}_n)] \\ &\simeq \mathrm{E}[\chi^2_{\kappa-k}(\delta)] \\ &\simeq \delta + \kappa - k \end{aligned}$$

on obtient l'estimation suivante de δ

$$\delta \simeq -2\ln r(\mathcal{X}) + k - \kappa.$$

Utilisant l'équation B.4, nous avons alors

$$\begin{aligned} 2\mathcal{R}(\hat{\boldsymbol{\theta}}_k) &\simeq -2\ln r(\mathcal{X}) + 2k - \kappa \\ &\simeq -2\mathrm{L}(\hat{\boldsymbol{\theta}}_k) + 2k + 2\mathrm{L}(\hat{\boldsymbol{\theta}}_\kappa) - \kappa. \end{aligned}$$

Choisissant κ comme une constante arbitrairement grande, on ne tient compte que des deux premiers termes pour enfin déduire le critère AIC :

$$\mathrm{AIC}(\mathcal{M}_k) = -2\mathrm{L}(\hat{\boldsymbol{\theta}}_k) + 2k.$$

Extension au critère AIC3

Bozdogan [21] [22] [25] a proposé, dans le cadre des mélanges, un autre estimateur du risque moyen $\mathcal{R}(\hat{\boldsymbol{\theta}}_k)$ au travers du critère AIC3. Il s'agit en fait d'une variante de AIC. La création de ce nouveau critère a été motivée par le fait que la distribution asymptotique de l'équation B.5 n'est pas vérifiée lorsque l'on teste sur la frontière de l'espace des paramètres, ce qui est le cas lors du choix du nombre de classes. S'appuyant, entre autres, sur les travaux de Wolfe [91] [92], il établit la conjecture d'une nouvelle distribution asymptotique :

$$-2C(n)\ln r(\boldsymbol{X}_1, \dots, \boldsymbol{X}_n) \sim \chi^2_{2(\kappa-k)}(\delta) \;\text{ asymptotiquement.}$$

avec le biais $\delta = \|n^{1/2}(\boldsymbol{\theta}^* - \boldsymbol{\theta}^*_k)\|^2_F$ et le terme de correction

$$C(n) = \frac{1}{n}\left(n - 1 - d - \frac{\text{(nombre de classes dans l'hypothèse alternative)}}{2}\right).$$

Le cheminement d'obtention de AIC3 est rigoureusement identique à celui de AIC, excepté l'usage de cette nouvelle distribution à l'équation B.5. Ainsi on obtient :

$$2\mathcal{R}(\hat{\boldsymbol{\theta}}_k) \simeq -2C(n)\mathrm{L}(\hat{\boldsymbol{\theta}}_k) + 3k + 2C(n)\mathrm{L}(\hat{\boldsymbol{\theta}}_\kappa) - 2\kappa.$$

D'où AIC3

$$\mathrm{AIC3}(\mathcal{M}_k) = -2C(n)\mathrm{L}(\hat{\boldsymbol{\theta}}_k) + 3k,$$

soit en remarquant que $C(n) \to 1$ quand $n \to \infty$

$$\mathrm{AIC3}(\mathcal{M}_k) = -2\mathrm{L}(\hat{\boldsymbol{\theta}}_k) + 3k.$$

Annexe C

Estimation de la matrice de Fisher

L'objectif de cette partie est d'estimer la matrice d'information de Fisher pour chacun des 28 modèles gaussiens. L'estimation de cette matrice est nécessaire pour calculer le critère ICOMP de Bozdogan.

La définition de la matrice d'information de Fisher $\boldsymbol{F}(\boldsymbol{\psi})$ en une valeur de paramètre $\boldsymbol{\psi}$ pour une observation $\boldsymbol{X}$ est donnée par

$$\begin{aligned} \boldsymbol{F}(\boldsymbol{\psi}) &= \mathrm{E}_X\left[\frac{\partial}{\partial\boldsymbol{\theta}}\ln f(\boldsymbol{X}|\boldsymbol{\theta})\bigg|_{\boldsymbol{\psi}} \cdot \frac{\partial}{\partial\boldsymbol{\theta}}\ln f(\boldsymbol{X}|\boldsymbol{\theta})\bigg|_{\boldsymbol{\psi}}'\right] \\ &= \int_{(X)} \frac{\partial}{\partial\boldsymbol{\theta}}\ln f(\boldsymbol{x}|\boldsymbol{\theta})\bigg|_{\boldsymbol{\psi}} \cdot \frac{\partial}{\partial\boldsymbol{\theta}}\ln f(\boldsymbol{x}|\boldsymbol{\theta})\bigg|_{\boldsymbol{\psi}}' \cdot f(\boldsymbol{x}|\boldsymbol{\theta})d\boldsymbol{x}, \end{aligned}$$

avec $(\boldsymbol{X}) \subseteq \mathrm{IR}^d$ le domaine de définition du vecteur aléatoire $\boldsymbol{X}$ et indépendant de $\boldsymbol{\theta}$. Nous prenons l'estimateur suivant de cette matrice :

$$\begin{aligned} \hat{\boldsymbol{F}}(\boldsymbol{\psi}) &= \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\partial}{\partial\boldsymbol{\theta}}\ln f(\boldsymbol{x}_i|\boldsymbol{\theta})\bigg|_{\boldsymbol{\psi}} \cdot \frac{\partial}{\partial\boldsymbol{\theta}}\ln f(\boldsymbol{x}_i|\boldsymbol{\theta})\bigg|_{\boldsymbol{\psi}}'\right). \\ &= \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{f(\boldsymbol{x}_i|\boldsymbol{\psi})^2} \cdot \frac{\partial}{\partial\boldsymbol{\theta}} f(\boldsymbol{x}_i|\boldsymbol{\theta})\bigg|_{\boldsymbol{\psi}} \cdot \frac{\partial}{\partial\boldsymbol{\theta}} f(\boldsymbol{x}_i|\boldsymbol{\theta})\bigg|_{\boldsymbol{\psi}}'\right). \end{aligned}$$

Il est démontré par la loi forte des grands nombres (voir par exemple Bernardo et Smith [10] p. 287 et 288) que $\hat{\boldsymbol{F}}(\boldsymbol{\psi}) \to \boldsymbol{F}(\boldsymbol{\psi}))$ quand $n \to \infty$.

Utilisant la décomposition spectrale de la matrice de variance $\boldsymbol{\Sigma}_k$ (relation 1.12), on a

$$\boldsymbol{\Sigma}_k = \lambda_k \boldsymbol{D}_k \boldsymbol{A}_k \boldsymbol{D}_k' \;\Leftrightarrow\; \boldsymbol{\Sigma}_k^{-1} = \frac{1}{\lambda_k}\boldsymbol{D}_k \boldsymbol{A}_k^{-1} \boldsymbol{D}_k'.$$

car la matrice $\boldsymbol{\Sigma}_k$ qui est définie positive est toujours inversible et de plus $\boldsymbol{D}_k^{-1} = \boldsymbol{D}_k'$ car $\boldsymbol{D}_k$ est une matrice orthogonale. L'expression générale de la k^{e} composante $\phi_k(\boldsymbol{x}|\boldsymbol{a}_k)$ du mélange gaussien s'énonce ensuite (d'après l'équation 1.1)

$$\phi_k(\boldsymbol{x}|\boldsymbol{a}_k) = \frac{1}{(2\pi)^{d/2}\lambda_k^{d/2}} \exp\left(-\frac{1}{2\lambda_k}(\boldsymbol{x}-\boldsymbol{\mu}_k)'\boldsymbol{D}_k\boldsymbol{A}_k^{-1}\boldsymbol{D}_k'(\boldsymbol{x}-\boldsymbol{\mu}_k)\right).$$

Exprimons maintenant, avec ces notations, l'expression du vecteur $\frac{\partial}{\partial\theta}f(\boldsymbol{x}|\boldsymbol{\theta})$ de dimension $q(\mathcal{M})$. Il est composé de la concaténation de certains des *vecteurs dérivées partielles* suivants en fonction bien sûr des paramètres libres permis par $\mathcal{M}$. Par exemple, pour le modèle $[p\lambda I]$ avec 2 classes on aura :

$$\frac{\partial}{\partial\boldsymbol{\theta}}f(\boldsymbol{x}|\boldsymbol{\theta})' = \left[\frac{\partial}{\partial\boldsymbol{\mu}_1}f(\boldsymbol{x}|\boldsymbol{\theta})' \ , \ \frac{\partial}{\partial\boldsymbol{\mu}_2}f(\boldsymbol{x}|\boldsymbol{\theta})' \ , \ \frac{\partial}{\partial\lambda}f(\boldsymbol{x}|\boldsymbol{\theta})'\right].$$

Centres

$$\frac{\partial}{\partial\boldsymbol{\mu}_k}f(\boldsymbol{x}|\boldsymbol{\theta}) = p_k\phi_k(\boldsymbol{x}|\boldsymbol{a}_k)\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k) \quad , \quad 1 \leq k \leq K.$$

Proportions libres

$$\frac{\partial}{\partial p_k}f(\boldsymbol{x}|\boldsymbol{\theta}) = \phi_k(\boldsymbol{x}|\boldsymbol{a}_k) - \phi_k(\boldsymbol{x}|\boldsymbol{a}_k) \quad , \quad 1 \leq k \leq K-1.$$

Volumes

Volumes égaux

$$\frac{\partial}{\partial\lambda}f(\boldsymbol{x}|\boldsymbol{\theta}) = \frac{1}{2\lambda}\left\{\sum_{k=1}^{K}\Big(p_k\phi_k(\boldsymbol{x}|\boldsymbol{a}_k)(\boldsymbol{x}-\boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)\Big) - d\,f(\boldsymbol{x}|\boldsymbol{\theta})\right\},$$

Volumes libres $(1 \leq k \leq K)$

$$\frac{\partial}{\partial\lambda_k}f(\boldsymbol{x}|\boldsymbol{\theta}) = \frac{p_k\phi_k(\boldsymbol{x}|\boldsymbol{a}_k)}{2\lambda_k}\Big((\boldsymbol{x}-\boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k) - d\Big).$$

Matrices de variance diagonales

On a posé $\boldsymbol{B}_k = \boldsymbol{D}_k\boldsymbol{A}_k\boldsymbol{D}_k'$ avec $\boldsymbol{D}_k$ matrice permutation de la base canonique. Donc $\boldsymbol{B}_k$ est diagonale avec ses éléments diagonaux non nécessairement classés par ordre croissant mais avec toujours $|\boldsymbol{B}_k| = 1$. On note $(1 \leq k \leq K)$

$$\boldsymbol{B}_k^{-1} = \mathrm{diag}\left(b_k^1, \ldots, b_k^{d-1}, \frac{1}{\prod_{q=1}^{d-1} b_k^q}\right)$$

et

$$\frac{\partial}{\partial b_k^q}\boldsymbol{B}_k^{-1} = \mathrm{diag}\left(0, \ldots, 0, \underbrace{1}_{\text{ligne } q}, 0, \ldots, 0, -\frac{1}{\prod_{j=1}^{d-1} b_k^j \cdot b_k^d}\right).$$

Matrices diagonales égales $(1 \leq q \leq d-1)$

$$\frac{\partial}{\partial b^q}f(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{k=1}^{K}\left(-\frac{p_k\phi_k(\boldsymbol{x}|\boldsymbol{a}_k)}{2\lambda_k}(\boldsymbol{x}-\boldsymbol{\mu}_k)'\frac{\partial}{\partial b^q}\boldsymbol{B}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k)\right),$$

Matrices diagonales libres $(1 \leq q \leq d-1 \text{ et } 1 \leq k \leq K)$

$$\frac{\partial}{\partial b_k^q} f(\boldsymbol{x}|\boldsymbol{\theta}) = -\frac{p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k)}{2\lambda_k}(\boldsymbol{x}-\boldsymbol{\mu}_k)' \frac{\partial}{\partial b_k^q} \boldsymbol{B}_k^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_k).$$

Matrices forme

La matrice forme $\boldsymbol{A}_k$ s'écrit comme la matrice diagonale précédente $\boldsymbol{B}_k$ en notant a_k^q les éléments diagonaux de $\boldsymbol{A}_k^{-1}$.

Formes égales $(1 \leq q \leq d-1)$

$$\frac{\partial}{\partial a^q} f(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{k=1}^{K} \left(-\frac{p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k)}{2\lambda_k}(\boldsymbol{x}-\boldsymbol{\mu}_k)' D_k \frac{\partial}{\partial a^q} \boldsymbol{A}^{-1} D_k'(\boldsymbol{x}-\boldsymbol{\mu}_k) \right),$$

Formes libres $(1 \leq q \leq d-1 \text{ et } 1 \leq k \leq K)$

$$\frac{\partial}{\partial a_k^q} f(\boldsymbol{x}|\boldsymbol{\theta}) = -\frac{p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k)}{2\lambda_k}(\boldsymbol{x}-\boldsymbol{\mu}_k)' D_k \frac{\partial}{\partial a_k^q} \boldsymbol{A}_k^{-1} D_k'(\boldsymbol{x}-\boldsymbol{\mu}_k).$$

Matrices orientation

Nous nous limiterons ici à la dimension deux ($d = 2$). Posons $(1 \leq k \leq K)$

$$\boldsymbol{D}_k = \begin{pmatrix} \cos\alpha_k & -\sin\alpha_k \\ \sin\alpha_k & \cos\alpha_k \end{pmatrix}, \quad \frac{\partial}{\partial \alpha_k}\boldsymbol{D}_k = \begin{pmatrix} -\sin\alpha_k & -\cos\alpha_k \\ \cos\alpha_k & -\sin\alpha_k \end{pmatrix}.$$

Orientations égales

$$\frac{\partial}{\partial \alpha} f(\boldsymbol{x}|\boldsymbol{\theta}) = \sum_{k=1}^{K} \left\{ -\frac{p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k)}{2\lambda_k}(\boldsymbol{x}-\boldsymbol{\mu}_k)' \left[\frac{\partial}{\partial \alpha}\boldsymbol{D} A_k^{-1} D' + D A_k^{-1} \frac{\partial}{\partial \alpha}\boldsymbol{D}' \right] (\boldsymbol{x}-\boldsymbol{\mu}_k) \right\},$$

Orientations libres

$$\frac{\partial}{\partial \alpha_k} f(\boldsymbol{x}|\boldsymbol{\theta}) = -\frac{p_k \phi_k(\boldsymbol{x}|\boldsymbol{a}_k)}{2\lambda_k}(\boldsymbol{x}-\boldsymbol{\mu}_k)' \left[\frac{\partial}{\partial \alpha_k}\boldsymbol{D}_k A_k^{-1} D_k' + D_k A_k^{-1} \frac{\partial}{\partial \alpha_k}\boldsymbol{D}_k' \right] (\boldsymbol{x}-\boldsymbol{\mu}_k).$$

Annexe D

Formules d'actualisation

Soit un ensemble E formé de n points $\boldsymbol{x}$ de $\mathbb{R}^d$ avec

$$\left\{ \begin{array}{ll} \boldsymbol{g} = \frac{\sum_{x \in E} \boldsymbol{x}}{n} & \text{son centre de gravité} \\ \boldsymbol{Q} = \sum_{x \in E} (\boldsymbol{x} - \boldsymbol{g})(\boldsymbol{x} - \boldsymbol{g})' & \text{sa matrice d'inertie} \end{array} \right. .$$

On ajoute à E (respectivement on enlève de E) un point $\boldsymbol{y}$ de $\mathbb{R}^d$ ce qui conduit à un nouvel ensemble E^* de cardinal n^* et définit par $E^* = E \cup \{\boldsymbol{y}\}$ (respectivement $E^* = E \setminus \{\boldsymbol{y}\}$). On a alors $n^* = n + \epsilon$ avec $\epsilon = +1$ pour l'ajout et $\epsilon = -1$ pour le retrait. On pose aussi $\boldsymbol{g}^*$ et $\boldsymbol{Q}^*$ le centre de gravité et la matrice d'inertie de E^*.

L'objectif est ici de produire des formules d'actualisation séquentielles pour le centre de gravité $\boldsymbol{g}$, la matrice d'inertie intraclasse $\boldsymbol{Q}$, ainsi que les termes plus complexes que sont $\boldsymbol{Q}^{-1}$, $|\boldsymbol{Q}|$ et $\mathrm{tr}(\boldsymbol{Q})$.

Actualisation de $\boldsymbol{g}$

Proposition 1 *Le centre de gravité $\boldsymbol{g}^*$ de E^* s'écrit*

$$\boldsymbol{g}^* = \frac{n\boldsymbol{g} + \epsilon \boldsymbol{y}}{n + \epsilon}.$$

Preuve Le calcul est direct à partir de la définition du centre de gravité :

$$\begin{aligned} \boldsymbol{g}^* &= \frac{\sum_{x \in E^*} \boldsymbol{x}}{n + \epsilon} \\ &= \frac{\sum_{x \in E} \boldsymbol{x} + \epsilon \boldsymbol{y}}{n + \epsilon} \\ &= \frac{n\boldsymbol{g} + \epsilon \boldsymbol{y}}{n + \epsilon}. \end{aligned}$$

□

Actualisation de $\boldsymbol{Q}$

Proposition 2 *La matrice d'inertie $\boldsymbol{Q}^*$ de E^* s'écrit*

$$\boldsymbol{Q}^* = \boldsymbol{Q} + \epsilon \boldsymbol{h}\boldsymbol{h}'$$

avec $\boldsymbol{h} = \omega(\boldsymbol{y} - \boldsymbol{g})$ et $\omega^2 = \frac{n}{n+\epsilon}$.

Preuve On a

$$\begin{aligned}
\boldsymbol{Q}^* &= \sum_{x\in E^*}(\boldsymbol{x}-\boldsymbol{g}^*)(\boldsymbol{x}-\boldsymbol{g}^*)' \\
&= \sum_{x\in E}(\boldsymbol{x}-\boldsymbol{g}+\boldsymbol{g}-\boldsymbol{g}^*)(\boldsymbol{x}-\boldsymbol{g}+\boldsymbol{g}-\boldsymbol{g}^*)' + \epsilon(\boldsymbol{y}-\boldsymbol{g}^*)(\boldsymbol{y}-\boldsymbol{g}^*)' \\
&= \underbrace{\sum_{x\in E}(\boldsymbol{x}-\boldsymbol{g})(\boldsymbol{x}-\boldsymbol{g})'}_{Q} + \underbrace{\sum_{x\in E}(\boldsymbol{x}-\boldsymbol{g})(\boldsymbol{g}-\boldsymbol{g}^*)'}_{0} + \underbrace{\sum_{x\in E}(\boldsymbol{g}-\boldsymbol{g}^*)(\boldsymbol{x}-\boldsymbol{g})'}_{0} \\
&\quad + \sum_{x\in E}(\boldsymbol{g}-\boldsymbol{g}^*)(\boldsymbol{g}-\boldsymbol{g}^*)' + \epsilon(\boldsymbol{y}-\boldsymbol{g}^*)(\boldsymbol{y}-\boldsymbol{g}^*)' \\
&= \boldsymbol{Q} + n(\boldsymbol{g}-\boldsymbol{g}^*)(\boldsymbol{g}-\boldsymbol{g}^*)' + \epsilon(\boldsymbol{y}-\boldsymbol{g}^*)(\boldsymbol{y}-\boldsymbol{g}^*)'.
\end{aligned}$$

D'autre part, on déduit de la formule de mise à jour du centre de gravité que

$$\left\{\begin{array}{l} \boldsymbol{g}-\boldsymbol{g}^* = -\frac{\epsilon}{n+\epsilon}(\boldsymbol{y}-\boldsymbol{g}) \\ \boldsymbol{y}-\boldsymbol{g}^* = \frac{n}{n+\epsilon}(\boldsymbol{y}-\boldsymbol{g}) \end{array}\right. .$$

D'où

$$\begin{aligned}
\boldsymbol{Q}^* &= \boldsymbol{Q} + n\frac{\epsilon^2}{(n+\epsilon)^2}(\boldsymbol{y}-\boldsymbol{g})(\boldsymbol{y}-\boldsymbol{g})' + \epsilon\frac{n^2}{(n+\epsilon)^2}(\boldsymbol{y}-\boldsymbol{g})(\boldsymbol{y}-\boldsymbol{g})' \\
&= \boldsymbol{Q} + \epsilon\frac{n}{n+\epsilon}(\boldsymbol{y}-\boldsymbol{g})(\boldsymbol{y}-\boldsymbol{g})'.
\end{aligned}$$

□

Actualisation de $\boldsymbol{Q}^{-1}$

Proposition 3 *Si* $\boldsymbol{Q}^* = \boldsymbol{Q} + \epsilon\boldsymbol{h}\boldsymbol{h}'$, *alors*

$$\boldsymbol{Q}^{*-1} = \boldsymbol{Q}^{-1} - \epsilon\frac{(\boldsymbol{Q}^{-1}\boldsymbol{h})(\boldsymbol{Q}^{-1}\boldsymbol{h})'}{1+\epsilon\boldsymbol{h}'\boldsymbol{Q}^{-1}\boldsymbol{h}}.$$

Preuve Il suffit de montrer que $\boldsymbol{Q}^*\boldsymbol{Q}^{*-1} = \boldsymbol{I}$:

$$\begin{aligned}
\boldsymbol{Q}^*\boldsymbol{Q}^{*-1} &= (\boldsymbol{Q}+\epsilon\boldsymbol{h}\boldsymbol{h}')\left(\boldsymbol{Q}^{-1} - \epsilon\frac{(\boldsymbol{Q}^{-1}\boldsymbol{h})(\boldsymbol{Q}^{-1}\boldsymbol{h})'}{1+\epsilon\boldsymbol{h}'\boldsymbol{Q}^{-1}\boldsymbol{h}}\right) \\
&= \boldsymbol{I} + \epsilon\frac{\overbrace{\boldsymbol{h}\boldsymbol{h}'\boldsymbol{Q}^{-1} - \boldsymbol{h}\boldsymbol{h}'\boldsymbol{Q}^{-1}}^{0} + \epsilon\overbrace{(\boldsymbol{h}\boldsymbol{h}'\boldsymbol{Q}^{-1}\boldsymbol{h}'\boldsymbol{Q}^{-1}\boldsymbol{h} - \boldsymbol{h}\boldsymbol{h}'\boldsymbol{Q}^{-1}\boldsymbol{h}\boldsymbol{h}'\boldsymbol{Q}^{-1})}^{0 \text{ car } h'Q^{-1}h\in\mathbb{R}}}{1+\epsilon\boldsymbol{h}'\boldsymbol{Q}^{-1}\boldsymbol{h}}.
\end{aligned}$$

□

Actualisation de $|\boldsymbol{Q}|$

Lemme 1 *Pour tout* $\boldsymbol{h} \in \mathbb{R}^d$, *on a*

$$|\boldsymbol{I} + \epsilon\boldsymbol{h}\boldsymbol{h}'| = 1 + \epsilon\boldsymbol{h}'\boldsymbol{h}.$$

Preuve Si $\boldsymbol{h} = (x_1, \ldots, x_d)'$, on a

$$
\begin{aligned}
|\boldsymbol{I} + \epsilon \boldsymbol{h}\boldsymbol{h}'| &= \epsilon^d \begin{vmatrix} \frac{1}{\epsilon} + x_1^2 & x_1 x_2 & \ldots & x_1 x_d \\ x_1 x_2 & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ x_1 x_d & \ldots & \ldots & \frac{1}{\epsilon} + x_d^2 \end{vmatrix} \\
&= \epsilon^d (x_1 \ldots x_d)^2 \begin{vmatrix} \frac{1}{\epsilon x_1^2} + 1 & 1 & \ldots & 1 \\ 1 & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ 1 & \ldots & \ldots & \frac{1}{\epsilon x_d^2} + 1 \end{vmatrix} \\
&= \epsilon^d (x_1 \ldots x_d)^2 \begin{vmatrix} \frac{1}{\epsilon x_1^2} + 1 & 1 & 1 & \ldots & 1 \\ -\frac{1}{\epsilon x_1^2} & \frac{1}{\epsilon x_2^2} & 0 & \ldots & 0 \\ -\frac{1}{\epsilon x_1^2} & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ -\frac{1}{\epsilon x_1^2} & 0 & \ldots & 0 & \frac{1}{\epsilon x_d^2} \end{vmatrix} \\
&= \begin{vmatrix} 1 + \epsilon x_1^2 & \epsilon x_2^2 & \epsilon x_3^2 & \ldots & \epsilon x_d^2 \\ -1 & 1 & 0 & \ldots & 0 \\ -1 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ -1 & 0 & \ldots & 0 & 1 \end{vmatrix} \\
&= \begin{vmatrix} 1 + \epsilon(x_1^2 + \ldots + x_d^2) & \epsilon x_2^2 & \epsilon x_3^2 & \ldots & \epsilon x_d^2 \\ 0 & 1 & 0 & \ldots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \ldots & 0 & 1 \end{vmatrix} \\
&= 1 + \epsilon(x_1^2 + \ldots + x_d^2) \\
&= 1 + \epsilon \boldsymbol{h}'\boldsymbol{h}.
\end{aligned}
$$

□

Proposition 4 *Si* $\boldsymbol{Q}^* = \boldsymbol{Q} + \epsilon \boldsymbol{h}\boldsymbol{h}'$, *alors*

$$|\boldsymbol{Q}^*| = |\boldsymbol{Q}|(1 + \epsilon \boldsymbol{h}'\boldsymbol{Q}^{-1}\boldsymbol{h}).$$

Preuve On a

$$|\boldsymbol{Q}^*| = |\boldsymbol{Q}||\boldsymbol{I} + \epsilon \boldsymbol{Q}^{-1}\boldsymbol{h}\boldsymbol{h}'|, \tag{D.1}$$

or $\boldsymbol{Q}$ symétrique donc $\boldsymbol{Q}^{-1}$ également. D'où $\boldsymbol{Q}^{-1}$ est diagonalisable et peut s'écrire avec $\boldsymbol{D}$ matrice diagonale des valeurs propres et $\boldsymbol{P}$ matrice de changement de base orthogonale ($\boldsymbol{P}' = \boldsymbol{P}^{-1}$) :

$$\boldsymbol{Q}^{-1} = \boldsymbol{P}\boldsymbol{D}\boldsymbol{P}'.$$

Le fait que $\boldsymbol{Q}$ soit définie positive implique que $\boldsymbol{Q}^{-1}$ le soit aussi et donc que ses valeurs propres soient positives. Notons $\boldsymbol{A}$ la matrice telle que $\boldsymbol{D} = \boldsymbol{A}^2$. On peut écrire

$$\begin{aligned}
\boldsymbol{Q}^{-1} &= \boldsymbol{PAAP}' \\
&= (\boldsymbol{PA})(\boldsymbol{PA})' \quad \text{car} \quad \boldsymbol{A} = \boldsymbol{A}' \\
&= \boldsymbol{BB}' \quad \text{avec} \quad \boldsymbol{B} = \boldsymbol{PA}.
\end{aligned}$$

Ainsi, en remplaçant $\boldsymbol{Q}^{-1}$ par $\boldsymbol{BB}'$ dans l'équation D.1

$$\begin{aligned}
|\boldsymbol{Q}^*| &= |\boldsymbol{Q}||\boldsymbol{I} + \epsilon\boldsymbol{BB}'\boldsymbol{hh}'| \\
&= |\boldsymbol{Q}||\boldsymbol{B}||\boldsymbol{B}^{-1} + \epsilon\boldsymbol{B}'\boldsymbol{hh}'| \\
&= |\boldsymbol{Q}||\boldsymbol{I} + \epsilon\boldsymbol{B}'\boldsymbol{hh}'\boldsymbol{B}| \\
&= |\boldsymbol{Q}||\boldsymbol{I} + \epsilon\boldsymbol{B}'\boldsymbol{h}(\boldsymbol{B}'\boldsymbol{h})'| \\
&= |\boldsymbol{Q}||\boldsymbol{I} + \epsilon\boldsymbol{kk}'| \quad \text{avec} \quad \boldsymbol{k} = \boldsymbol{B}'\boldsymbol{h} \\
&= |\boldsymbol{Q}|(1 + \epsilon\boldsymbol{k}'\boldsymbol{k}) \quad \text{car} \quad |\boldsymbol{I} + \epsilon\boldsymbol{kk}'| = 1 + \epsilon\boldsymbol{k}'\boldsymbol{k} \quad \text{(lemme précédent)} \\
&= |\boldsymbol{Q}|(1 + \epsilon\boldsymbol{h}'\underbrace{\boldsymbol{BB}'}_{Q^{-1}}\boldsymbol{h}) \\
&= |\boldsymbol{Q}|(1 + \epsilon\boldsymbol{h}'\boldsymbol{Q}^{-1}\boldsymbol{h}).
\end{aligned}$$

□

Actualisation de tr($\boldsymbol{Q}$)

Proposition 5 *Si* $\boldsymbol{Q}^* = \boldsymbol{Q} + \epsilon\boldsymbol{hh}'$, *alors*

$$tr(\boldsymbol{Q}^*) = tr(\boldsymbol{Q}) + \epsilon\boldsymbol{h}'\boldsymbol{h}.$$

Preuve La trace étant un opérateur linéaire, on a

$$\mathrm{tr}(\boldsymbol{Q}^*) = \mathrm{tr}(\boldsymbol{Q}) + \epsilon\,\mathrm{tr}(\boldsymbol{hh}').$$

Si $\boldsymbol{A}$ et $\boldsymbol{B}$ sont deux matrices de dimension respective $(p \times q)$ et $(q \times p)$, il est bien connu que $\mathrm{tr}(\boldsymbol{AB}) = \mathrm{tr}(\boldsymbol{BA})$. Donc

$$\mathrm{tr}(\boldsymbol{Q}^*) = \mathrm{tr}(\boldsymbol{Q}) + \epsilon\,\mathrm{tr}(\boldsymbol{h}'\boldsymbol{h}).$$

On conclut en remarquant que la quantité $\boldsymbol{h}'\boldsymbol{h}$ est réelle et donc que $\mathrm{tr}(\boldsymbol{h}'\boldsymbol{h}) = \boldsymbol{h}'\boldsymbol{h}$. □

Annexe E

Données réelles utilisées

Cette annexe regroupe les données réelles qui ont été utilisées.

Données d'hémophilie

- Source : Habbema et al. [55] ;
- Données : voir le tableau E.1.

Les papillons

- Source : Celeux et Robert [39] ;
- Données : voir la figure E.1 et le tableau E.2.

Les Iris de Fisher

- Source : Anderson [5] ;
- Données : voir le tableau E.3.

The Old Faithful Geyser

- Source : Azzalini et Bowman [6] ;
- Données : voir le tableau E.4.

n°	logarithme base 10 de Activité AHF	logarithme base 10 de Antigène AHF	Classe
1	-0.0056	-0.1657	1
2	-0.1698	-0.1585	1
3	-0.3469	-0.1879	1
4	-0.0894	0.0064	1
5	-0.1679	0.0713	1
6	-0.0836	0.0106	1
7	-0.1979	-0.0005	1
8	-0.0762	0.0392	1
9	-0.1913	-0.2123	1
10	-0.1092	-0.1190	1
11	-0.5268	-0.4773	1
12	-0.0842	0.0248	1
13	-0.0225	-0.0580	1
14	0.0084	0.0782	1
15	-0.1827	-0.1138	1
16	0.1237	0.2140	1
17	-0.4702	-0.3099	1
18	-0.1519	-0.0686	1
19	0.0006	-0.1153	1
20	-0.2015	-0.0498	1
21	-0.1932	-0.2293	1
22	0.1507	0.0933	1
23	-0.1259	-0.0669	1
24	-0.1551	-0.1232	1
25	-0.1951	-0.1007	1
26	0.0291	0.0442	1
27	-0.2228	-0.1710	1
28	-0.0997	-0.0733	1
29	-0.1972	-0.0607	1
30	-0.0867	-0.0560	1
31	-0.4986	-0.0860	2
32	-0.5015	-0.2984	2
33	-0.1326	0.0097	2
34	-0.3479	-0.1721	2
35	-0.3755	-0.1865	2
36	-0.2447	-0.0407	2
37	-0.2205	0.0046	2
38	-0.2154	-0.0219	2
39	-0.2540	-0.0573	2
40	-0.3778	-0.2682	2
41	-0.0639	0.1569	2
42	-0.3351	-0.1368	2
43	-0.0149	0.1539	2
44	-0.0312	0.1400	2
45	-0.1740	-0.0776	2
46	-0.0964	0.0531	2
47	-0.0234	0.0804	2
48	-0.4055	-0.2418	2
49	-0.3478	0.1151	2
50	-0.3618	-0.2008	2
51	-0.6911	-0.3390	2
52	-0.3608	0.1237	2
53	-0.4535	-0.1682	2
54	-0.3539	0.0722	2
55	-0.4719	-0.1079	2
56	-0.3610	-0.0399	2
57	-0.3226	0.1670	2
58	-0.4319	-0.0687	2
59	-0.2734	-0.0020	2
60	-0.5573	0.0548	2
61	-0.4950	-0.0153	2
62	-0.5107	-0.2483	2
63	-0.1652	0.2132	2
64	-0.4232	-0.0998	2
65	-0.2375	0.2876	2
66	-0.3447	0.0097	2
67	-0.4046	-0.1162	2
68	-0.1416	0.1642	2
69	-0.1508	0.1137	2
70	-0.2642	0.0867	2
71	-0.3352	0.0875	2
72	-0.1878	0.2510	2
73	-0.1744	0.1892	2
74	-0.2444	0.1614	2
75	-0.4784	0.0282	2

Légende des classes
1 – Femmes saines, 2 – Femmes hémophiles A

TAB. E.1 – *Donnée d'hémophilie.*

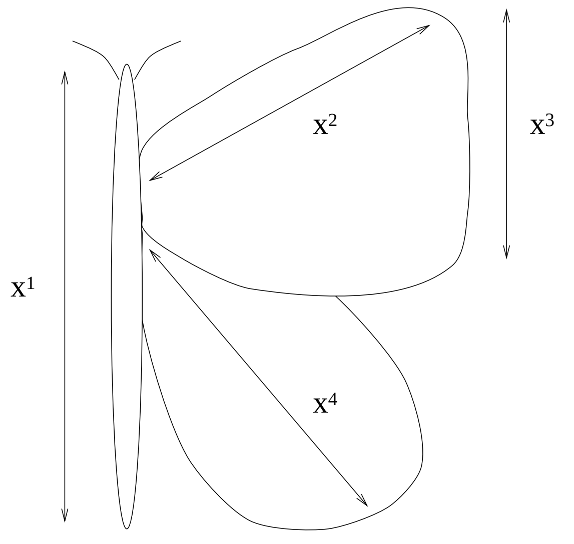


FIG. E.1 – *Mesures d'un papillon.*

n°	$\boldsymbol{x}^1$ (mm)	$\boldsymbol{x}^2$ (mm)	$\boldsymbol{x}^3$ (mm)	$\boldsymbol{x}^4$ (mm)
1	22	35	24	19
2	24	31	21	22
3	27	36	25	15
4	27	36	24	23
5	21	33	23	18
6	26	35	23	32
7	27	37	26	15
8	22	30	19	20
9	25	33	22	22
10	30	41	28	17
11	24	39	27	21
12	29	39	27	17
13	29	40	27	17
14	28	36	23	24
15	22	36	24	20
16	23	30	20	20
17	28	38	26	16
18	25	34	23	14
19	26	35	24	15
20	23	37	25	20
21	31	42	29	18
22	26	34	22	21
23	24	38	26	21

TAB. E.2 – *Les papillons.*

n°	Longueur (cm)		Largeur (cm)		Classe
	Sépale	Pétale	Sépale	Pétale	
1	5.1	3.5	1.4	0.2	1
2	4.9	3.0	1.4	0.2	1
3	4.7	3.2	1.3	0.2	1
4	4.6	3.1	1.5	0.2	1
5	5.0	3.6	1.4	0.2	1
6	5.4	3.9	1.7	0.4	1
7	4.6	3.4	1.4	0.3	1
8	5.0	3.4	1.5	0.2	1
9	4.4	2.9	1.4	0.2	1
10	4.9	3.1	1.5	0.1	1
11	5.4	3.7	1.5	0.2	1
12	4.8	3.4	1.6	0.2	1
13	4.8	3.0	1.4	0.1	1
14	4.3	3.0	1.1	0.1	1
15	5.8	4.0	1.2	0.2	1
16	5.7	4.4	1.5	0.4	1
17	5.4	3.9	1.3	0.4	1
18	5.1	3.5	1.4	0.3	1
19	5.7	3.8	1.7	0.3	1
20	5.1	3.8	1.5	0.3	1
21	5.4	3.4	1.7	0.2	1
22	5.1	3.7	1.5	0.4	1
23	4.6	3.6	1.0	0.2	1
24	5.1	3.3	1.7	0.5	1
25	4.8	3.4	1.9	0.2	1
26	5.0	3.0	1.6	0.2	1
27	5.0	3.4	1.6	0.4	1
28	5.2	3.5	1.5	0.2	1
29	5.2	3.4	1.4	0.2	1
30	4.7	3.2	1.6	0.2	1
31	4.8	3.1	1.6	0.2	1
32	5.4	3.4	1.5	0.4	1
33	5.2	4.1	1.5	0.1	1
34	5.5	4.2	1.4	0.2	1
35	4.9	3.1	1.5	0.2	1
36	5.0	3.2	1.2	0.2	1
37	5.5	3.5	1.3	0.2	1
38	4.9	3.6	1.4	0.1	1
39	4.4	3.0	1.3	0.2	1
40	5.1	3.4	1.5	0.2	1
41	5.0	3.5	1.3	0.3	1
42	4.5	2.3	1.3	0.3	1
43	4.4	3.2	1.3	0.2	1
44	5.0	3.5	1.6	0.6	1
45	5.1	3.8	1.9	0.4	1
46	4.8	3.0	1.4	0.3	1
47	5.1	3.8	1.6	0.2	1
48	4.6	3.2	1.4	0.2	1
49	5.3	3.7	1.5	0.2	1
50	5.0	3.3	1.4	0.2	1
51	7.0	3.2	4.7	1.4	2
52	6.4	3.2	4.5	1.5	2
53	6.9	3.1	4.9	1.5	2
54	5.5	2.3	4.0	1.3	2
55	6.5	2.8	4.6	1.5	2
56	5.7	2.8	4.5	1.3	2
57	6.3	3.3	4.7	1.6	2
58	4.9	2.4	3.3	1.0	2
59	6.6	2.9	4.6	1.3	2
60	5.2	2.7	3.9	1.4	2
61	5.0	2.0	3.5	1.0	2
62	5.9	3.0	4.2	1.5	2
63	6.0	2.2	4.0	1.0	2
64	6.1	2.9	4.7	1.4	2
65	5.6	2.9	3.6	1.3	2
66	6.7	3.1	4.4	1.4	2
67	5.6	3.0	4.5	1.5	2
68	5.8	2.7	4.1	1.0	2
69	6.2	2.2	4.5	1.5	2
70	5.6	2.5	3.9	1.1	2
71	5.9	3.2	4.8	1.8	2
72	6.1	2.8	4.0	1.3	2
73	6.3	2.5	4.9	1.5	2
74	6.1	2.8	4.7	1.2	2
75	6.4	2.9	4.3	1.3	2
76	6.6	3.0	4.4	1.4	2
77	6.8	2.8	4.8	1.4	2
78	6.7	3.0	5.0	1.7	2
79	6.0	2.9	4.5	1.5	2
80	5.7	2.6	3.5	1.0	2
81	5.5	2.4	3.8	1.1	2
82	5.5	2.4	3.7	1	2
83	5.8	2.7	3.9	1.2	2
84	6.0	2.7	5.1	1.6	2
85	5.4	3.0	4.5	1.5	2
86	6.0	3.4	4.5	1.6	2
87	6.7	3.1	4.7	1.5	2
88	6.3	2.3	4.4	1.3	2
89	5.6	3.0	4.1	1.3	2
90	5.5	2.5	4.0	1.3	2
91	5.5	2.6	4.4	1.2	2
92	6.1	3.0	4.6	1.4	2
93	5.8	2.6	4.0	1.2	2
94	5.0	2.3	3.3	1.0	2
95	5.6	2.7	4.2	1.3	2
96	5.7	3.0	4.2	1.2	2
97	5.7	2.9	4.2	1.3	2
98	6.2	2.9	4.3	1.3	2
99	5.1	2.5	3.0	1.1	2
100	5.7	2.8	4.1	1.3	2
101	6.3	3.3	6.0	2.5	3
102	5.8	2.7	5.1	1.9	3
103	7.1	3.0	5.9	2.1	3
104	6.3	2.9	5.6	1.8	3
105	6.5	3.0	5.8	2.2	3
106	7.6	3.0	6.6	2.1	3
107	4.9	2.5	4.5	1.7	3
108	7.3	2.9	6.3	1.8	3
109	6.7	2.5	5.8	1.8	3
110	7.2	3.6	6.1	2.5	3
111	6.5	3.2	5.1	2.0	3
112	6.4	2.7	5.3	1.9	3
113	6.8	3.0	5.5	2.1	3
114	5.7	2.5	5.0	2.0	3
115	5.8	2.8	5.1	2.4	3
116	6.4	3.2	5.3	2.3	3
117	6.5	3.0	5.5	1.8	3
118	7.7	3.8	6.7	2.2	3
119	7.7	2.6	6.9	2.3	3
120	6.0	2.2	5.0	1.5	3
121	6.9	3.2	5.7	2.3	3
122	5.6	2.8	4.9	2.0	3
123	7.7	2.8	6.7	2.0	3
124	6.3	2.7	4.9	1.8	3
125	6.7	3.3	5.7	2.1	3
126	7.2	3.2	6.0	1.8	3
127	6.2	2.8	4.8	1.8	3
128	6.1	3.0	4.9	1.8	3
129	6.4	2.8	5.6	2.1	3
130	7.2	3.0	5.8	1.6	3
131	7.4	2.8	6.1	1.9	3
132	7.9	3.8	6.4	2.0	3
133	6.4	2.8	5.6	2.2	3
134	6.3	2.8	5.1	1.5	3
135	6.1	2.6	5.6	1.4	3
136	7.7	3.0	6.1	2.3	3
137	6.3	3.4	5.6	2.4	3
138	6.4	3.1	5.5	1.8	3
139	6.0	3.0	4.8	1.8	3
140	6.9	3.1	5.4	2.1	3
141	6.7	3.1	5.6	2.4	3
142	6.9	3.1	5.1	2.3	3
143	5.8	2.7	5.1	1.9	3
144	6.8	3.2	5.9	2.3	3
145	6.7	3.3	5.7	2.5	3
146	6.7	3.0	5.2	2.3	3
147	6.3	2.5	5.0	1.9	3
148	6.5	3.0	5.2	2.0	3
149	6.2	3.4	5.4	2.3	3
150	5.9	3.0	5.1	1.8	3

Légende des classes
1 – Iris Setosa, 2 – Iris Virginica, 3 – Iris Versicolor

TAB. E.3 – *Les Iris de Fisher.*

n°	Durée (mn)	n°	Durée (mn)	n°	Durée (mn)	n°	Durée (mn)	n°	Durée (mn)
1	4.017	61	1.950	121	1.817	181	1.783	241	1.917
2	2.150	62	5.267	122	4.100	182	4.917	242	4.417
3	4.000	63	2.000	123	4.650	183	1.817	243	3.000
4	4.000	64	4.000	124	4.000	184	4.000	244	4.000
5	4.000	65	2.000	125	2.000	185	4.000	245	2.000
6	2.000	66	4.000	126	4.000	186	4.000	246	4.000
7	4.383	67	2.000	127	4.000	187	3.867	247	3.283
8	4.283	68	4.000	128	4.217	188	1.850	248	1.833
9	2.033	69	3.533	129	4.133	189	4.700	249	4.617
10	4.833	70	2.167	130	3.933	190	2.017	250	1.833
11	1.833	71	4.500	131	3.750	191	4.467	251	4.617
12	5.450	72	2.017	132	4.417	192	1.867	252	4.600
13	1.617	73	4.150	133	2.467	193	4.167	253	4.250
14	4.867	74	4.200	134	4.167	194	1.900	254	1.933
15	4.383	75	4.333	135	3.800	195	4.250	255	4.983
16	1.767	76	1.933	136	4.317	196	3.250	256	1.967
17	4.667	77	4.650	137	3.867	197	4.217	257	4.300
18	2.000	78	3.817	138	4.683	198	1.883	258	4.200
19	4.733	79	4.033	139	1.700	199	4.983	259	4.533
20	4.217	80	4.167	140	4.967	200	1.850	260	4.400
21	1.900	81	4.667	141	4.267	201	4.000	261	4.617
22	4.967	82	1.817	142	4.583	202	1.967	262	2.000
23	2.000	83	4.000	143	4.000	203	4.767	263	4.000
24	4.000	84	3.000	144	4.000	204	4.000	264	4.000
25	2.000	85	4.000	145	4.000	205	2.000	265	3.917
26	4.000	86	2.000	146	4.000	206	4.000	266	2.000
27	2.833	87	4.450	147	1.983	207	4.000	267	4.500
28	4.500	88	2.050	148	4.600	208	2.383	268	1.800
29	4.067	89	4.250	149	0.833	209	4.417	269	4.000
30	3.717	90	1.917	150	4.917	210	4.217	270	2.750
31	3.517	91	4.667	151	1.733	211	4.367	271	4.733
32	4.467	92	1.733	152	4.583	212	2.000	272	3.967
33	2.217	93	4.383	153	1.700	213	4.450	273	1.950
34	4.883	94	1.767	154	4.750	214	1.750	274	4.967
35	2.600	95	4.600	155	1.833	215	4.500	275	1.850
36	4.150	96	1.867	156	4.500	216	1.617	276	4.800
37	2.200	97	4.450	157	1.867	217	4.700	277	4.000
38	4.767	98	1.633	158	4.450	218	2.567	278	4.000
39	1.833	99	5.033	159	4.450	219	3.700	279	4.000
40	4.600	100	1.817	160	4.000	220	4.233	280	4.000
41	2.267	101	5.100	161	4.800	221	1.933	281	4.000
42	4.133	102	1.633	162	4.000	222	4.350	282	4.000
43	2.000	103	4.283	163	4.000	223	4.000	283	4.000
44	4.000	104	2.000	164	2.000	224	4.000	284	2.000
45	2.000	105	4.000	165	4.000	225	4.000	285	4.000
46	4.000	106	2.000	166	1.933	226	4.217	286	1.933
47	1.883	107	4.533	167	4.583	227	4.000	287	4.333
48	4.267	108	2.000	168	2.000	228	4.133	288	1.667
49	2.083	109	4.000	169	3.700	229	1.883	289	4.767
50	4.467	110	2.933	170	2.867	230	4.467	290	1.950
51	2.500	111	4.733	171	4.833	231	1.950	291	4.683
52	4.000	112	3.900	172	3.450	232	4.217	292	1.933
53	1.767	113	1.950	173	4.383	233	1.717	293	4.417
54	4.333	114	4.117	174	1.800	234	4.450	294	2.133
55	2.183	115	1.800	175	4.400	235	4.250	295	4.083
56	4.483	116	4.667	176	2.483	236	3.967	296	2.067
57	3.883	117	1.833	177	4.517	237	4.383	297	4.000
58	3.333	118	4.700	178	2.100	238	1.967	298	4.000
59	3.733	119	2.117	179	4.350	239	4.450	299	2.000
60	4.000	120	4.783	180	4.367	240	4.267		

TAB. E.4 – *Durée déruption de* The Old FaithFul Geyser.

Bibliographie

[1] M. Aitkin and D.B. Rubin. Estimation and Hypothesis Testing in Finite Mixture models. *Journal of the Royal Statistical Society, Series B*, 47:67–75, 1985.

[2] H. Akaike. Information theory as an extension of the maximum likelihood principle. In B. Petrov and F. Csaki, editors, *Second International Symposium on Information Theory*, pages 267–281, Budapest, Akademiai Kiado, 1973.

[3] H. Akaike. A new look at the statistical identification model. *IEEE Transactions on Automatic Control*, 19:716–723, 1974.

[4] H. Akaike. Information Measures and Model Selection. *Bulletin of the International Statistical Institute*, 50:277–290, 1983.

[5] E. Anderson. The irises of the gaspé peninsula. *Bulletin of the American Iris Society*, 59:2–5, 1935.

[6] A. Azzalini and A.W. Bowman. A Look at Some Data on the Old Faithful Geyser. *Applied Statistics*, 39:357–365, 1990.

[7] J.D. Banfield and A.E. Raftery. Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49:803–821, 1993.

[8] H. Bensmail and G. Celeux. Regularized Gaussian Discriminant Analysis through Eigenvalue Decomposition. *Journal of the American Statistical Association*, 91:1743–1748, 1996.

[9] H. Bensmail, G. Celeux, A.E. Raftery, and C.P. Robert. Inference in Model-Based Analysis. Unpublished, 1995.

[10] J.M. Bernardo and A.F.M. Smith. *Bayesian theory.* John Wiley & Sons, England, 1994.

[11] J.D. Bezdeck. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum, New York, 1981.

[12] C. Biernacki. Classification à partir de gaussiennes à liberté restreinte. Technical report, Université de Technologie de Compiègne, Compiègne, France, 1993. Rapport de DEA.

[13] C. Biernacki and G. Govaert. Detecting the Number of Clusters using Parsimonious Models with a Penalized Likelihood in Clustering. In *Congrès International sur les Mélanges de Distribution*, Aussois, France, September 17-21 1995. Poster.

[14] C. Biernacki and G. Govaert. Choix du nombre de classes par la vraisemblance classifiante, 16 décembre 1996. Séminaire du CREST-INSEE, Paris, France.

[15] C. Biernacki and G. Govaert. Comparaison de critères en classification pour la détection de modèles gaussiens parcimonieux. In *Quatrièmes journées de la Société Francophone de Classification*, Vannes, France, 19-20 septembre 1996.

[16] C. Biernacki and G. Govaert. Choosing Gaussian Models in Discriminant Analysis. In *IV International Meeting of Multidimentional Data Analysis*, Bilbao, Spain, September 10-12 1997.

[17] C. Biernacki and G. Govaert. Using the classification likelihood to choose the number of clusters. In *The second World Conference of the International Association for Statistical Computing*, Pasadena, USA, February 19-22 1997.

[18] S. Binder. Bayesian cluster analysis. *Biometrika*, 65:31–38, 1978.

[19] H.H. Bock. Statistical Testing and Evaluation Methods in Cluster Analysis. In *Proceedings of the Indian Statistical Institute Golden Jubilee International Conference on Statistics: Applications and New Directions*, pages 116–146, Calcutta, December 16-19 1981.

[20] H.H. Bock. On Tests Concerning the Existence of the Classification. *Journal of Classification*, 2:77–108, 1985.

[21] H. Bozdogan. *Multi-Sample Cluster Analysis and Approaches to Validity Studies in Clustering Individuals.* PhD thesis, Department of Mathematics, University of Illinois, Chicago, IL 60680, 1981.

[22] H. Bozdogan. Determining the number of clusters in the standart multivariate normal mixture model using model-selection criteria. Technical report no. uic/dqm/a83-1, Quantitative Methods Department, University of Illinois, Chicago, IL 60680, June 16 1983. ARO contract DAAGL90820K-0155.

[23] H. Bozdogan. Model Selection and Akaike's Information Criterion (aic): The General Theory and Its Analytical Extensions. *Psychometrika*, 52(3):345–370, 1987. Special Section (invited paper).

[24] H. Bozdogan. On the information-based measure of covariance complexity an its application to the evaluation of the multivariate linear models. *Communication in Statistics, Theory and Methods*, 19:221–278, 1990.

[25] H. Bozdogan. *Choosing the Number of Component Clusters in the Mixture-Model Using a New Informational Complexity Criterion of the Inverse-Fisher Information Matrix.* Information and Classification. Springer-Verlag, Heidelberg, 1993.

[26] H. Bozdogan. Mixture-Model Cluster Analysis using Model Selection Criteria and a New Informational Measure of Complexity. In *Congrès International sur les mélanges de distributions*, Aussois, France, 17-21 septembre 1995.

[27] H. Bozdogan and S.L. Sclove. Multi-Sample Cluster Analysis using Akaike's Information Criterion. *Annals of Institute of Statistical Mathematics*, 36:163–180, 1984.

[28] P.G. Bryant. Large-Sample Results for Optimization Based Clustering Methods. *Journal of Classification*, 8:31–44, 1991.

[29] P.G. Bryant. On Detecting the Number of Clusters Using the MDL Principle. Unpublished, 1993.

[30] P.G. Bryant and J.A. Williamson. Asymptotic Behaviour of Classification Maximum Likelihood Estimates. *Biometrika*, 65:273–281, 1978.

[31] P.G. Bryant and J.A. Williamson. Maximum Likelihood and Classification: a Comparison of Three Approaches. In W. Gaul and M. Schader, editors, *Classification as a tool of research*, pages 33–45. North-Holland, 1986.

[32] G. Celeux, C. Biernacki, and G. Govaert. Choosing models in model-based clustering and discriminant analysis. In *Joint Statistical Meetings of the American Statistical Association*, Anaheim/Orange County, USA, August 10-14 1997.

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

[34] G. Celeux and J. Diebolt. L'algorithme SEM : Un algorithme d'apprentissage probabiliste pour la reconnaissance de mélange de densités. *Revue de Statistique Appliquée*, 34(2):35–52, 1986.

[35] G. Celeux and J. Diebolt. Une version de type recuit simulé de l'algorithme EM. *Notes aux Comptes Rendus de l'Académie des Sciences*, 310:119–124, 1990.

[36] G. Celeux and G. Govaert. A classification EM algorithm for clustering and two stochastic versions. *Computational Statistics & Data Analysis*, 14:315–332, 1992.

[37] G. Celeux and G. Govaert. Comparison of the Mixture and the Classification Maximum Likelihood in Cluster Analysis. *Journal of Statistical Computation and Simulation*, 47:127–146, 1993.

[38] G. Celeux and G. Govaert. Gaussian Parsimonious Models. *Pattern Recognition*, 28(5):781–793, 1995.

[39] G. Celeux and C. Robert. Uune histoire de discrétisation (with Discussion). *La revue de Modulad*, 11:7–44, 1993.

[40] G. Celeux and G. Soromenho. An entropy criterion for assessing the number of clusters in a mixture model. *Journal of Classification*, 13(2):195–212, 1996.

[41] A. Cutler and M.P. Windham. Information-Based Validity Functionals for Mixture Analysis. In H. Bozdogan, editor, *Proceedings of the first US-Japan Conference on the Frontiers of Statistical Modeling*, pages 149–170, Amsterdam, 1993. Kluever (Academic Publishers).

[42] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum Likelihood from Incomplete Data with the EM Algorithm (with discussion). *Journal of the Royal Statistical Society, Series B*, 39:1–38, 1977.

[43] J. Diebolt and C. Robert. Bayesian estimation of finite mixture distributions, Part i: theoretical aspects. Technical Report 110, LSTA, Université Paris VI, 1990.

[44] J. Diebolt and C. Robert. Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society, Series B*, 1992.

[45] B. Dubuisson. *Diagnostic et reconnaissance des formes*. Diagnostic et Maintenance. Hermès, Paris, 1990.

[46] B.S. Everitt. A Monte-Carlo investigation of the likelihood ratio test for the number of components in a mixture of normal distributions. *Multivariate Behavioral Research*, 16:171–180, 1981.

[47] B.S. Everitt and D.J. Hand. *Finite Mixture Distributions*. Chapman and Hall, London, 1981.

[48] B.W. Flury, M.J. Schmid, and A. Narayanan. Error rates in quadratic discrimination with constraints on the covariance matrices. *Journal of Classification*, 11:101–120, 1994.

[49] H.P. Friedman and J. Rubin. On some invariant criteria for grouping data. *Journal of the American Statistical Association*, 62:1159–1178, 1967.

[50] J. Friedman. Regularized Discriminant Analysis. *Journal of the American Statistical Association*, 84:165–175, 1989.

[51] S. Ganesalingam. Classification and Mixture Approaches to Clustering via Maximum Likelihood. *Applied Statistics*, 38:455–466, 1989.

[52] A. Gelfand and A.F. Smith. Sampling based approaches to calculating marginal densities. *J. Amer. Statist. Assoc.*, 85:398–409, 1990.

[53] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the bayesian restoration of images. *IEEE Trans. Patt. Anal. Mach. Intel.*, 6:721–740, 1984.

[54] J. Geweke. Bayesian inference in econometric models using Monte-Carlo integration. *Econometrica*, 57:1317–1340, 1989.

[55] J. Habbema, J. Hermans, and K. Van Den Broek. A stepwise discriminant analysis program using density estimation. In *Compstat 1974, Proceedings in Computational Statistics*, pages 101–110, Wien, 1974. Physica Verlag.

[56] J.A. Hartigan. *Clustering Algorithms.* John Wiley & Sons, New York, 1975.

[57] W.K. Hastings. Monte-Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57:97–109, 1970.

[58] R.J. Hathaway. Another Interpretation of the EM Algorithm for Mixtures Distributions. *Statistics and Probability Letters*, 4:53–56, 1986.

[59] H. Jeffreys. Some tests of significance, treated by the theory of probability. *Proc. Camb. Phil. Soc.*, 31:203–222, 1935.

[60] H. Jeffreys. *Theory of Probability.* Oxford University Press, third edition, 1961.

[61] R.E. Kass. Bayes factors in practice. *Statistician*, 42:551–560, 1993.

[62] R.E. Kass and J.B. Greenhouse. comment on "Investigating therapies of potentially great benefit: ECMO". *Statist. Science*, 4:310–317, 1989. By Ware.

[63] R.E. Kass and A.E. Raftery. Bayes factors. *Journal of the American Statistical Association*, 90, 1995. To appear.

[64] N.M. Kiefer. Discrete parameter variation: efficient estimation of a switching regression model. *Econometrica*, 46:427–434, 1978.

[65] S.M. Lewis and A.E. Raftery. Estimating Bayes factors via posterior simulation with the Laplace-Metropolis estimator. Technical Report 279, Department of Statistics, University of Washington, 1994.

[66] H. Linhart and W. Zucchini. *Model Selection.* Wiley, New York, 1986.

[67] F.H.C. Marriott. Separating Mixtures of Normal Distributions. *Biometrics*, 31:767–769, 1975.

[68] G.J. Mclachlan. On Bootstraping the Likelihood Ratio Test Statistic for the Number of Components in a Normal Mixture. *Applied Statistics*, 36:318–324, 1987.

[69] G.J. McLachlan. *Discriminant Analysis and Statistical Pattern Recognition.* Wiley, New York, 1992.

[70] G.J. Mclachlan and K.E. Basford. Likelihood estimation with normal mixture models. *Applied Statistics*, 34:282–289, 1985.

[71] M.A. Newton and A.E. Raftery. Approximate Bayesian inference by the weighted likelihood bootstrap (with discussion). *Journal of the Royal Statistical Society, series B*, 56:3–48, 1994.

[72] K. Pearson. Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society*, 185:71–110, 1894.

[73] B.G. Quinn, G.J. Mclachlan, and N.L. Hjort. A note on the Aitkin-Rubin Approach to Hypothesis Testing in Mixture Models. *JRSS B*, 49:311–314, 1987.

[74] A.E. Raftery. *Hypothesis testing and model selection via posterior simulation.* Practical Markov Chain Monte Carlo. Chapman and Hall, London, 1995. To appear.

[75] H. Raiffa and R. Schlaifer. *Applied Statistical Decision Theory.* Division of Research, Graduate School of Business Administration, Harvard University, 1961.

[76] R. Redner and H. Walker. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, 26:195–239, 1984.

[77] B.D. Ripley. *Pattern Recognition and Neural Networks.* Cambridge University Press, 1996.

[78] J. Rissanen. *Stochastic Complexity in Statistical Inquiry.* World Scientific, Teaneck, New Jersey, 1989.

[79] C. Robert. *L'analyse Statistique Bayésienne.* Economica, Paris, 1992.

[80] G. Saporta. *Probabilités, Analyse des Données et Statistique*, chapter 12, pages 241–260. Eds Technip, Paris, 1990.

[81] G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6:461–464, 1978.

[82] A.J. Scott and M.J. Symons. Clustering methods based on likelihood ratio criteria. *Biometrics*, 27:387–397, 1971.

[83] G.S. Sebestyen. *Decision Making Process in Pattern Recognition.* Macmillan, New York, 1962.

[84] G. Soromenho. Comparing Approaches for Testing the Number of Components in a Finite Mixture Model. *Computational Statistics*, 9:65–78, 1994.

[85] M. Stone. An Asymptotic Equivalence of Choice of Model by Cross-Validation and Akaike's Criterion. *Journal of Royal Statistical Society, Series B*, 39:44–47, 1977.

[86] M.J. Symons. Clustering criteria and multivariate normal mixtures. *Biometrics*, 37:35–43, 1981.

[87] M. Tanner and W. Wong. The calculation of posterior distributions by data augmentation. *J. Amer. Statist. Assoc.*, 82:528–550, 1987.

[88] A. Wald. Tests of statistical hypotheses concerning several parameters when the number of observations is large. *Transactions of the American Mathematical Society*, 54:426–482, 1943.

[89] J.H. Ward. Hierarchical groupings to optimize an objective function. *Journal of the American Statistical Association*, 58:236–244, 1963.

[90] M.P. Windham and A. Cutler. Information Ratios for Validating Cluster Analysis. *Journal of the American Statistical Association*, 87:1188–1192, 1992.

[91] J.H. Wolfe. Pattern Clustering by Multivariate Mixture Analysis. *Multivariate Behavioral Research*, 5:329–350, 1970.

[92] J.H. Wolfe. A Monte-Carlo Study of the Sampling Distribution of the Likelihood Ratio for Mixtures of Multinormal Distributions. Technical Report Technical Bulletin STB 72-2, US Naval Personnel Research activity, San Diego, California, 1971.

[93] C. Wu. On the convergence properties of the EM algorithm. *Annals of Statistics*, 1983.