EM : complexifier le modèle ou estimer de manière robuste

Christophe Saint-Jean, Carl Frélicot, Bertrand Vachon

EM : complexifier le modèle ou estimer de manière robuste

EM : complex models vs robust estimation

C. Saint-Jean* C. Frélicot B. Vachon

L3I - UPRES EA 1216

Avenue de Marillac
17042 La Rochelle Cedex 1
{csaintje,cfrelico,bvachon}@univ-lr.fr

Résumé

Notre étude compare deux stratégies liées à la maximisation de la vraisemblance via l'algorithme EM. La première consiste à utiliser des modèles théoriques plus complexes qu'à l'habitude, la seconde en l'intégration d'estimateurs robustes dans l'étape de maximisation de l'algorithme. Nous avons appliqué ces deux techniques séparement puis conjointement dans le cadre de données bruitées pour en tester la robustesse. Nous proposons également un modèle hybride et en présentons les résultats.

Mots Clef

Classification, Caractérisation et évaluation de performances, EM, Robustesse, M-estimateur.

Abstract

Our study compares two strategies related to the likehood maximization via the EM algorithm. The first one consists in using theorical models which are more complex than usual ones. The second is based on integration of robust estimators in the maximization step of the algorithm. We have applied these two techniques separately then jointly within the framework of noisy data in order to test theirs robustnesses. Finally, we introduce a hybrid model for a class and present the results in this paper.

Keywords

Clustering, Characterization and evaluation of performances, EM, robustness, M-estimator.

*Ce travail est financé par le Conseil Général de la Charente-Maritime

1 Introduction

La statistique connait aujourd'hui un vif succès dans des domaines très divers. Elle trouve aussi bien son application dans des domaines traditionnels tels que l'analyse économique que dans des processus industriels (Ex : détection automatisée de défauts de fabrication par analyse d'images). De nombreux logiciels issus de la recherche sont maintenant accessibles par Internet. Parmi ceux-ci, citons EMMIX de G. McLachlan et D. Peel [MP98] ou encore MCLUST de C. Fraley et A. Raftery [FR98b].

La classification (appelée encore catégorisation) a pour objectif de mettre en évidence des relations entre des objets décrits par un certain nombre de variables et de les regrouper en entités homogènes suivant une certaine mesure de similarité.
Dans notre cas, les attributs décrivant les objets sont uniquement de type numérique et l'on notera :

$$x_i = (x_i^1, \ldots, x_i^P)^T$$

le vecteur attribut pour le i-ème objet.
Notre échantillon χ est constitué de N descriptions :

$$\chi = \{x_1, x_2, \ldots, x_N\}$$

Lorsqu'on se place du point de vue modèle de mélange, on considère les éléments de l'échantillon χ comme les réalisations d'un vecteur aléatoire X de dimension P issues de C composantes de paramètres Θ_i $(i \in 1, C)$. La densité de mélange en x s'écrit dès lors :

$$f(x;\Theta) = \sum_{i=1}^{C} \pi_i f(x;\Theta_i)$$

où π_i désigne la probabilité a priori de la i-ème composante ($\sum_{i=1}^{k} \pi_i = 1$) et $\Theta = (\pi_1, \ldots, \pi_C, \Theta_1^T, \ldots, \Theta_C^T)^T$ les paramètres du modèle.
Dans le cas d'une composante normale indicée par i, les paramètres sont $\Theta_i = (\mu_i, \Sigma_i)^T$ où μ_i représente le vecteur moyenne et Σ_i la matrice de covariance.
On souhaite modéliser la densité de probabilité de X à l'aide d'un modèle de paramètre Θ en se servant de ses réalisations (et éventuellement de connaissances a priori supplémentaires).
Une démarche classique consiste à trouver une estimation $\hat{\Theta}$ de Θ qui maximise la vraisemblance

$$\mathcal{L}(\Theta) = P(\chi|\Theta) = \prod_{i=1}^{N} \sum_{j=1}^{C} \pi_j f(x_i; \Theta_j)$$

en supposant les réalisations de X indépendantes.
Malheureusement, il arrive souvent que les données que nous ayons à traiter soient bruitées.
On définit généralement le bruit comme une distortion d'un modèle hypothétique théorique. Il provient aussi bien de pertubations dans l'acquisition des données (défaillance d'un capteur ou conditions extérieures) que de pertubations liées au transport et au stockage des données (Ex. : Compression pour les images). Un algorithme est robuste s'il est peu sensible au bruit.
Jusqu'à récemment, les algorithmes de classification n'intégraient directement que peu ou pas de techniques robustes pour cause de coût de calcul. Aujourd'hui, l'augmentation de la puissance des machines permet leur utilisation.

Dans ce papier, nous nous demandons s'il est préferable de complexifier le modèle théorique d'une classe ou de robustifier l'estimation des paramètres de modèles plus simples pour tenir compte des données bruitées.

A la section 2, nous rappelons les fondements de l'algorithme EM. Nous donnerons les bases de l'estimation robuste à l'aide des M-estimateurs à la section 3. Nous présenterons ensuite les résultats que nous avons obtenus sur divers jeux de données (section 4) avant de conclure en section 5.

2 EM

2.1 Cadre général

L'algorithme EM est une technique itérative de maximisation de la vraisemblance en présence de données incomplètes. On l'attribut généralement à Dempster, Laird et Rubin [DLR77] même s'il y a eu antérieurement quelques travaux connexes [DH73].

Comme nous l'avons dit plus haut, la vraisemblance des données relativement au modèle de paramètre Θ s'écrit :

$$\mathcal{L}(\Theta) = P(\chi|\Theta)$$

Cette maximisation étant difficile à réaliser directement, on introduit une variable aléatoire Z correspondant aux données cachées ou manquantes. L'idée de cet algorithme est faciliter le processus d'optimisation en utilisant une estimation de ces données manquantes.

Au lieu de maximiser $\mathcal{L}(\Theta)$, on maximise itérativement l'espérance conditionnelle de la vraisemblance complète qui s'écrit :

$$\mathcal{L}_c(\Theta) = P(X, Z|\Theta)$$

L'algorithme EM alterne successivement deux phases :

-E-Step
Calcul de l'espérance conditionnelle de la vraisemblance complète :

$$Q(\Theta|\Theta^{(t)}) = E[\mathcal{L}_c(\Theta)|\chi, \Theta^{(t)}]$$

où χ est un ensemble de réalisations de X et $\Theta^{(t)}$ l'estimation des paramètres à l'instant t. Cette étape revient à engendrer une distribution de probabilité pour Z.

-M-Step
Maximisation de $Q(\Theta|\Theta^{(t)})$:

$$Q^{(t+1)} = \arg\max_{\Theta} Q(\Theta|\Theta^{(t)})$$

On cherche à maximiser l'estimation de la vraisemblance obtenue dans l'étape précedente.

L'une des proprietés de cet algorithme est d'améliorer la vraisemblance $L(\Theta)$ après chaque itération jusqu'à stabilité. Le lecteur interessé trouvera des détails sur la convergence de EM dans [DLR77] et [Wu83].
Cet algorithme possède deux inconvénients majeurs. Il est fortement dépendant de l'initialisation $\Theta^{(0)}$ et converge vers un extremum local qui risque d'être éloigné de l'extremum global. De plus, il peut se revèler coûteux en temps machine pour des applications de taille importante.

2.2 Utilisation en classification

L'algorithme EM est abondamment utilisé en classification dans le cadre d'un modèle de mélange ([MP98], [FR98a]).

Sous cette hypothèse, la log-vraisemblance $\log \mathcal{L}(\Theta)$ s'écrit :

$$\log \mathcal{L}(\Theta) = \sum_{i=1}^{N} \log \sum_{j=1}^{C} \pi_j f(x_i; \Theta_j)$$

On considère la donnée incomplète comme la classe d'appartenance de chacun des éléments à classifier. On écrit pour l'élement x_i :

$$z_{ij} = \begin{cases} 1 & \text{si } x_i \text{ appartient à la classe } C_j \\ 0 & \text{sinon} \end{cases}$$

Généralement, les $z_i = (z_{i1}, \ldots, z_{iC})$ sont pris comme des réalisations indépendantes suivant une distribution multinomiale $Mult_k(1, \pi)$ où $\pi = (\pi_1, \ldots, \pi_C)^T$. Sous ces conditions, la log-vraisemblance complète s'écrit :

$$\log \mathcal{L}_c(\Theta) = \sum_{i=1}^{N} \sum_{j=1}^{C} z_{ij} \log(\pi_j f(x_i; \Theta_j)) \qquad (1)$$

L'algorithme EM repète deux phases :

E-Step (Expectation Step) :
On calcule l'espérance conditionnelle de $\log \mathcal{L}_c$:

$$Q(\Theta|\Theta^{(t)}) = E[Log\mathcal{L}_c(\Theta)|\chi, \Theta^{(t)}] \qquad (2)$$

Comme $Log\mathcal{L}_c$ est une fonction linéaire des z_{ij}, ce calcul se limite à remplacer les z_{ij} par leur espérance conditionnelle :

$$E[z_{ij}|\chi, \Theta^{(t)}] = \frac{\pi_j f(x_i; \Theta_j^{(k)})}{\sum_{l=1}^{C} \pi_l f(x_i; \Theta_l^{(k)})}$$

Les z_{ij} sont les probabilités a posteriori d'appartenance de l'élément x_i à la classe C_j.

M-Step (Maximization Step) : Dans cette phase, on recherche la valeur de Θ qui maximise (2). Cette étape dépend complètement du modèle de classe choisi.

2.3 Cas d'un mélange gaussien

Nous allons expliciter le calcul dans le cas d'un mélange gaussien.
Rappellons tout d'abord l'expression d'une densité gaussienne multi-dimensionnelle de dimension P :

$$f(x; \Theta_j) = \frac{1}{(2\pi)^{P/2}\sqrt{|\Sigma_j|}} e^{-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1}(x-\mu_j)} \qquad (3)$$

où μ_j et Σ_j sont respectivement le vecteur moyenne et la matrice de covariance de la j-ème gaussienne.

Le calcul de l'espérance conditionnelle revient à remplacer les z_{ij} par leurs estimations :

$$\hat{z}_{ij} = \frac{\pi_j f(x_i; \Theta_j)}{\sum_{l=1}^{C} \pi_l f(x_i; \Theta_l)}$$

La phase de maximisation quant à elle relève de l'estimation de :

- $\hat{\pi}_j$: Estimation de la probabilité a priori de la j-ème classe

$$\hat{\pi}_j^{(k+1)} = \frac{\sum_{i=1}^{N} \hat{z}_{ij}}{N} \qquad (4)$$

- $\hat{\mu}_j$: Estimation de la moyenne de la j-ème classe

$$\hat{\mu}_j^{(k+1)} = \frac{\sum_{i=1}^{N} \hat{z}_{ij} x_i}{\sum_{i=1}^{N} \hat{z}_{ij}} \qquad (5)$$

- $\hat{\Sigma}_j$: Estimation de la matrice de covariance de la j-ème classe

$$\hat{\Sigma}_j^{(k+1)} = \frac{\sum_{i=1}^{N} \hat{z}_{ij}(x_i - \hat{\mu}_j^{(k+1)})(x_i - \hat{\mu}_j^{(k+1)})^T}{\sum_{i=1}^{N} \hat{z}_{ij}} \qquad (6)$$

3 Robustesse et m-estimateur

3.1 Cadre général

On souhaite estimer un paramètre $a = (a_1, \ldots, a_m)$ représentant un échantillon. Soit e_i l'écart entre la donnée observée g_i et la prévision de cette donnée $\hat{g}_i$:

$$e_i = g_i - \hat{g}_i(a)$$

Cette variable e suit une loi de distribution J. Notre objectif est de minimiser l'erreur sur l'ensemble des données. Dans le cas où l'échantillon est indépendant, on peut utiliser la méthode du maximum de vraisemblance qui revient à maximiser :

$$\prod_{i=1}^{N} J(e_i) \qquad (7)$$

Dans le cas mono-dimensionnel et si l'on fait l'hypothèse que e suit une loi normale, on aboutit à la méthode des moindres carrés de Legendre. Par contre, si l'on suppose que e suit une loi exponentielle alors on retrouve l'estimateur médian.
On peut transformer la maximisation de (7) par la minimisation d'une fonction de coût :

$$C(a) = \sum_{i=1}^{N} \rho(\frac{g_i - \hat{g}_i(a)}{\sigma_i}) \qquad (8)$$

avec $\rho = \log(J^{-1})$ et σ_i une pondération de l'erreur (incertitude). La minimisation de (8) s'effectue par la résolution d'un système de m équations différentielles:

$$\frac{\partial C(a)}{\partial a_k} = \sum_{i=1}^{N} \frac{1}{\sigma_i} \psi(\frac{g_i - \hat{g}_i(a)}{\sigma_i}) \frac{\partial \hat{g}_i(a)}{\partial a_k} = 0 \qquad (9)$$

avec $\psi = \frac{d\rho}{dx}(x)$. Ce système n'a pas de solution générale et il convient de l'étudier selon la fonction ρ. Par la suite, nous noterons $w(x) = \frac{\psi(x)}{x}$ la fonction de poids.

3.2 Quelques estimateurs

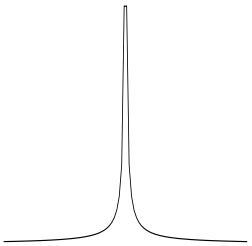
Nous allons maintenant présenter quelques M-estimateurs. Le graphique associé à chacun des modèles représente la fonction de poids en relation avec l'erreur.

Modèle de Legendre

$\rho(x) = x^2$
$\psi(x) = 2x$
$w(x) = 2$

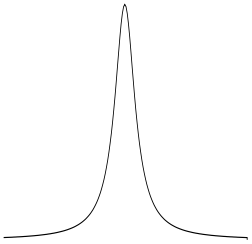
L-estimateur (Médiane)

$\rho(x) = |x|$
$\psi(x) = sgn(x)$
$w(x) = \frac{1}{|x|}$



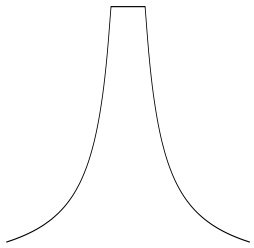
Modèle de Cauchy/Lorentz

$\rho(x) = \frac{c^2}{2}\log(1 + (\frac{x}{c})^2)$
$\psi(x) = \frac{x}{1+(\frac{x}{c})^2}$
$w(x) = \frac{1}{1+(\frac{x}{c})^2}$



Modèle de Huber

$$\rho(x) = \begin{cases} \frac{x^2}{2} \text{ si } |x| \leq c \\ c|x| - \frac{c^2}{2} \text{ sinon} \end{cases}$$
$$\psi(x) = \begin{cases} x \text{ si } |x| \leq c \\ c\ sgn(x) \text{ sinon} \end{cases}$$
$$w(x) = \begin{cases} 1 \text{ si } |x| \leq c \\ \frac{c}{|x|} \text{ sinon} \end{cases}$$



3.3 Estimation robuste des paramètres d'un modèle gaussien

Certains des estimateurs que nous venons de présenter possèdent une fonction de poids dont la forme rappelle celle d'une gaussienne. Ainsi les données sur lesquelles l'erreur est faible influeront d'autant plus sur le calcul. Au contraire, une donnée avec une erreur forte ne jouera pas un grand rôle.
Nous allons maintenant évoquer le cas de l'estimation robuste de la moyenne et de la matrice de covariance dans le cadre de l'étape de maximisation dans l'algorithme EM.

Estimation robuste de la moyenne. La procédure décrite ici effectue une estimation itérative de la moyenne m. En voici le fonctionnement général. Dans un premier temps, on calcule la moyenne arithmétique sur les données. Ensuite, on affecte un poids p_i à chaque x_i en fonction de la distance de celui-ci à la moyenne précédemment calculée. On réestime la moyenne en tenant compte des poids affectés par la formule :

$$\hat{m} = \frac{\sum_{i=1}^{N} p_i x_i}{\sum_{i=1}^{N} p_i} \qquad (10)$$

On recommence l'opération tant que l'estimateur n'a pas convergé. On note $m^{(t)}$ l'estimation de la moyenne à l'itération t.

t $\longleftarrow$ 1;
$m^{(0)} = \frac{\sum_{i=1}^{N} z_i x_i}{\sum_{i=1}^{N} z_i}$;
répéter
  pour *i allant de 1 à N* **faire**
    $e_i \longleftarrow d(x_i, m^{(t-1)})$;
    $w_i \longleftarrow \frac{\psi(e_i)}{e_i}$;
  $m^{(t)} \longleftarrow \frac{\sum_{i=1}^{N} w_i\ z_i\ x_i}{\sum_{i=1}^{N} w_i\ z_i}$;
  t $\longleftarrow$ t+1;
jusqu'à *Condition d'arrêt*;

Algorithme 1: Estimation robute de la moyenne

Dans l'algorithme, le choix de ψ se fait parmi les modèles que nous avons présentés précédemment. Le calcul du résidu e_i fait intervenir une fonction de distance entre la donnée x_i et la moyenne estimée $\hat{m}^{(t)}$. On prend généralement la distance de Mahalanobis:

$$d_{\mathcal{M}}(x_i, \Omega_j) = \sqrt{(x_i - m_j)^T \Sigma_j^{-1} (x_i - m_j)}$$

avec Ω_j une classe gaussienne de paramètres (μ_j, Σ_j).

3.4 Estimation robuste de la matrice de covariance

On peut imaginer coupler le procédé ci-dessus avec le calcul de la matrice de covariance Σ. En effet, on prendrait la même pondération que celle obtenue dans la moyenne qui donnerait :

$$\hat{\Sigma}^{(t)} = \frac{\sum_{i=1}^{N} w_i \; z_i (x_i - \hat{m}^{(t)})(x_i - \hat{m}^{(t)})^T}{\sum_{i=1}^{N} w_i \; z_i}$$

On permet de cette façon une remise à jour à la volée de la matrice de covariance dont on tient compte pour le calcul de la distance de Mahalanobis.

3.5 Remarques

La complexité du l'algorithme de calcul de la moyenne passe de O(n) à O(k×n) où k désigne le nombre d'itérations effectuées.
On peut également remarquer que les fonctions de poids sont symétriques et monotones décroissantes sur $[0;+\infty[$. Dans le cas de notre estimation itérative, cela implique que plus l'on effectue d'itérations, plus l'estimation repose sur un petit nombre de valeurs. Ainsi, ce que l'on gagne en robustesse est perdu en précision. Pour cette raison, on borne habituellement le nombre d'itérations. Le test de convergence peut être une combinaison de plusieurs types :

1. $t < t_{max}$
2. $\frac{|\hat{m}^{(t)} - \hat{m}^{(t+1)}|}{\hat{m}^{(t)}} < \epsilon$
3. Taux maximal d'élimination $\alpha < \alpha_{max}$ où α est le pourcentage des données qui ont un poids quasi-nul.

Nous avons utilisé une combinaison de 1 et 3 avec $\alpha_{max} = 50\%$.

4 Comparaison des deux stratégies

Nous souhaitons déterminer s'il convient d'employer des modèles plus complexes, ou s'il est préferable de robustifier le processus de maximisation.

L'un des problèmes de la densité gaussienne est que la base est relativement étroite. Ainsi, une donnée située à 3 fois l'écart-type a une probabilité a priori quasi-nulle (0.0044) d'être issue de cette celle-ci. De fait, si après un certain nombre d'itérations de EM, le modèle est perturbé par des données bruitées, l'algorithme risque de se retrouver piégé dans un extremum local éloigné de extremum global. Notre idée est d'utiliser des modèles plus souples.

4.1 Modèles testés

Voici les modèles testés :

1. Mélange de gaussiennes.
   Chaque classe est modélisée par une loi normale. Les paramètres sont estimés par les formules (3) à (5).
   $$f(x; \Theta) = \sum_{i=1}^{C} \pi_i \; \mathcal{N}(x; \mu_i, \Sigma_i)$$
   $\mathcal{N}$ représente la densité gaussienne multi-dimensionnelle (3).

2. Mélange de gaussiennes + une uniforme.
   On rajoute à 1 une loi uniforme pour modéliser le bruit de manière globale.
   $$f(x; \Theta) = \epsilon \; \mathcal{U}(x; H) + \sum_{i=1}^{C} \pi_i \; \mathcal{N}(x; \mu_i, \Sigma_i)$$
   $\mathcal{U}$ représente la loi uniforme appliquée dans l'hyper-cube H englobant les données.

3. Mélange de gaussiennes avec estimation robuste.
   $$f(x; \Theta) = \sum_{i=1}^{C} \pi_i \; \mathcal{N}(x; \bar{\mu_i}, \bar{\Sigma_i})$$
   où $\bar{\mu_i}$ et $\bar{\Sigma_i}$ sont les estimations robustes respectives des vecteurs moyennes et la matrice de covariance de la i-ème composante.

4. Mélange de gaussiennes avec estimation robuste + uniforme.
   $$f(x; \Theta) = \epsilon \; \mathcal{U}(x; H) + \sum_{i=1}^{C} \pi_i \; \mathcal{N}(x; \bar{\mu_i}, \bar{\Sigma_i})$$

5. Nous proposons une modélisation intra-classe du bruit. Chaque classe est modélisée par :
   $$f(x; \Theta) = \sum_{i=1}^{C} [(1-\epsilon) \; \mathcal{N}(x; \bar{\mu_i}, \bar{\Sigma_i}) + \epsilon \; \mathcal{N}(x; \mu_i, c\Sigma_i)]$$
   On choisira de préference ϵ petit et c grand (cf. Fig. 1)

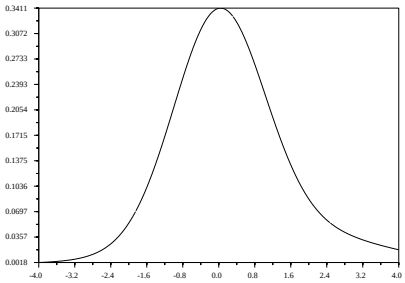


Figure 1: Modèle 5

	Echantillon 1	Echantillon 2
μ_1	7 9	4 6
Σ_1	3 −2 −2 1.5	1 0 0 3
ρ_1	-0.9428	0
μ_2	6 3	7 10
Σ_2	3 0 0 1	2 1.5 1.5 2
ρ_2	0.	0.75
μ_3	3 7	7 6
Σ_3	1 0 0 1	3.25 −3 −3 3.25
ρ_3	0.	-0.923
Uniforme	0,12;0,12	0,12;0,12
Tailles	100,150,100,150	200,150,150,100

Table 1 : Paramètres théoriques des classes

Ces modèles sont des versions modifiées des distorsions proposées par Huber [Hub81] et Y. Kharin [Kha97].

4.2 Méthodologie des tests

Jeux de données. Nous avons géneré deux jeux de données de manière à tester la robustesse des différents modèles proposés. Le premier échantillon (cf Fig. 2) consiste en trois classes bien séparées auxquelles on a ajouté un bruit uniforme dans une proportion de 30%.

Le second échantillon (cf Fig. 3) contient trois classes, mais celles-ci se chevauchent. On a également ajouté un bruit uniforme sur la zone englobante des données dans une proportion de 16,6%.
Le tableau 1 résume les paramètres théoriques de génération des données.

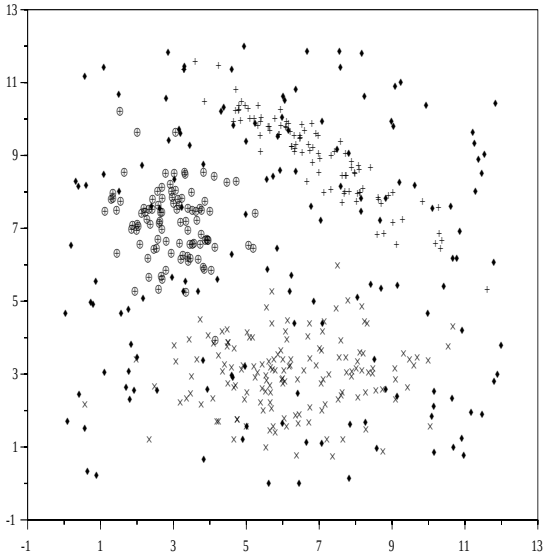


Figure 2: Échantillon 1

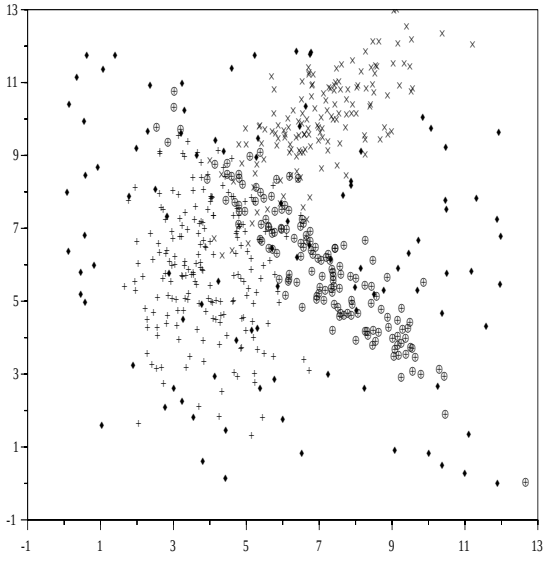


Figure 3: Échantillon 2

Paramétrage des modèles. L'utilisation de l'algorithme EM nécessite la connaissance du nombre de classes du modèle. Ce paramètre peut être estimé par des critères de validation tels que le "Normalized Entropy Criterion" (NEC) [BG98]. Pour notre part, nous avons fixé le nombre de classes à 3 dans chacun des cas.
Nous présentons ici les résultats correspondants à :

- Estimation robuste - Modèles 3 à 5 :
  M-estimateur de Cauchy-Lorentz avec c = $\sqrt{2}$.

- Modélisation du bruit - Modèles 2 et 4 :
  La probabilité a priori ϵ de la loi uniforme est fixée à 0.25.

- Modélisation du bruit - Modèle 5 :
  La proportion ϵ entre les deux gaussiennes est fixée à 0.25, tandis que le facteur d'échelle c de la seconde gaussienne est fixé à 5.

Conditions de comparaison. Comme nous le savons, l'algorithme EM est très dépendant des conditions d'initialisation. Pour pallier à ce problème, nous avons relancé l'algorithme 100 fois en choisissant à chaque fois une position aléatoire des centres μ_k des classes et une matrice de covariance $\Sigma_k = I$. Les résultats présentés sont ceux pour lesquels le taux de réussite est le meilleur : l'algorithme a affecté correctement un individu par rapport à sa classe d'origine selon le critère du Maximum A Posteriori (MAP) :

$$d(x, \hat{\Theta}) = \arg \max_{i \in [1,C]} (P(C_i|x))$$

4.3 Résultats

L'estimation robuste fait perdre la qualité de croissance monotone de l'algorithme. Ceci provient du fait que l'on a substitué l'estimation robuste à l'estimation issue de l'annulation de la différentielle $\frac{dQ(\Theta|\Theta^{(k)})}{d\Theta}$. La maximisation de $Q(\Theta|\Theta^{(k)})$ n'est ainsi pas assurée.

Le bruit dans les données pertube le calcul des matrices de covariance et des moyennes (dans une moindre mesure). Il nous a ainsi semblé plus intéressant de considérer l'erreur commise dans l'estimation du coefficient de corrélation que celle concernant la matrice de covariance.
Il permet donc de valider ou non l'orientation des classes engendrées par rapport aux axes par l'algorithme.

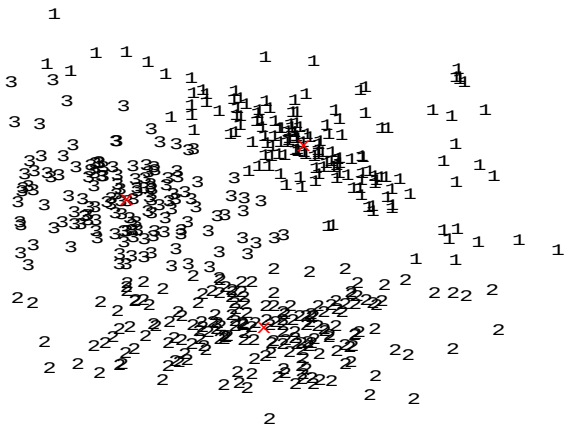


Figure 4: Modèle 3 - échantillon 1

Voici maintenant le résultat obtenu pour chacun des modèles sur E1 et E2.

Modèle 1 : Mélange de gaussiennes
Comme nous l'avons dit plus haut, l'estimation des paramètres de la loi normale est perturbée par les données extrêmes. Les classes engendrées se sont déplacés vers l'extérieur (cf. résultat sur les moyennes). Pour E1 et E2, l'erreur d'estimation des paramètres est souvent plus importante qu'avec les autres modèles. Néanmoins, le taux de classement demeure élevé malgré l'absence de modélisation ou d'estimation robuste. On peut expliquer ceci par l'importance du taux de bruit dans E1. Chaque élément issu de la loi uniforme perturbe le système de telle façon que finalement tout s'équilibre.

Modèle 2 : Mélange de gaussiennes + une uniforme
Ici, l'objectif est de modéliser le bruit par une loi uniforme. Comme cette loi a été également employé pour générer les perturbations, on peut s'attendre à obtenir de bons résultats. Dans E1, on retrouve quasiment la même erreur sur l'estimation des moyennes que dans le modèle 1. En revanche, on améliore significativement les résultats sur les coefficients de corrélation. La modélisation du bruit par la loi uniforme permet d'effectuer un filtrage des données. En effet, pour

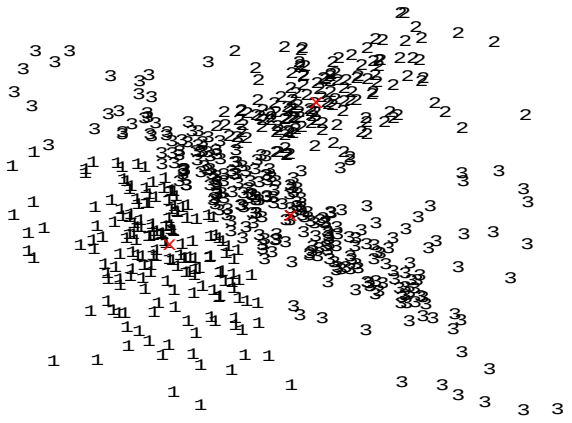


Figure 5: Modèle 1 - échantillon 2

un élément donné x et une classe gaussienne C, la probabilité a priori que x appartienne à C decroit rapidement alors que la probabilité a priori que x appartienne à la classe uniforme est constante. Du fait de la normalisation, les points éloignés de toute classe ont une probabilité a posteriori forte d'apppartenir à la classe uniforme et très faible de provenir d'une classe gaussienne. Ils ont ainsi un poids quasi-nul dans le processus d'estimation. Ici, le filtrage a été trop fort car plusieurs points du mélange ont été affectés à la classe uniforme.
Dans le cadre de l'échantillon E2, on a obtenu les meilleurs résultats grâce à ce modèle. En particulier, la classe 3 est bien retrouvée (Fig. 6).

Modèle 3 : Mélange de gaussiennes avec estimation robuste
Dans le modèle 3, on cherche à corriger les erreurs commises dans le modèle 1 du fait des données bruitées. Pour E1, on améliore l'estimation des paramètres. On remarque que le taux d'erreur d'affectation est plus faible ici que dans le modèle 2 alors l'estimation des moyennes et matrices de covariance est bien meilleure que dans le modèle 1. Cela laisse à supposer que la probabilité fixée a priori de la classe uniforme est trop élevée. Sur E2, les résultats sont identiques voire moins bons qu'avec le modèle 1. On trouve peut-être ici une limite de notre estimation robuste. A la fin du processus, le calcul repose sur un nombre réduit de valeurs. A la prochaine itération de l'algorithme, le système ne parviendra pas à sortir d'un minimum local.

Modèle 4 : Mélange de gaussiennes avec estimation robuste + uniforme
Ce modèle cumule à la fois les inconvénients et les avantages de la modèlisation du bruit par une loi uniforme et de l'estimation robuste. Le taux d'erreur d'affectation est très grand alors l'estimation des paramètres est très bonne (la meilleure des 5 modèles). Comme pour le modèle 2, la loi uniforme filtre une première fois les données. Ensuite, lors de l'estimation robuste, on refiltre les données si bien que très peu de données (mais suffisament) interviennent dans les itérations finales. Cela produit ainsi fréquemment des matrices singulières. On obtient une grande précision sur l'estimation des paramètres alors que les données doublement filtrées sont assignées à la classe de bruit.

Modèle 5 : Modélisation intra-classe du bruit
Ici, l'objectif est de modéliser le bruit à travers la densité de probabilité de chaque classe. Pour E1, on retrouve quasiment les résultats obtenus avec le modèle 3 si ce n'est un élément qui diffère. Dans le cas de l'échantillon 2, on améliore fortement les résultats par rapport à ce dernier. Comme pour le modèle 3, la classe 3 n'a pas été totalement retrouvée si bien que les éléments se trouvant dans la zone où le mélange est important ont été mal affectés.

Tous les modèles que nous avons présenté sont paramétrés à des degrés divers. Il est clair que les valeurs de ces paramètres sont déterminantes dans la réussite du processus de classification. Certaines valeurs, tel que le seuil c dans l'estimation de Cauchy-Lorentz, doivent pouvoir être reliées à la variance des éléments de la classe courante de manière à adapter la largeur du filtrage.

5 Conclusions et perspectives

Nous avons essayé de comparer deux approches pour la prise en compte du bruit dans l'algorithme EM : l'une consistant à complexifier le modèle théorique des classes ou du mélange, l'autre revenant à estimer de manière robuste les paramètres de modèles plus simples. Aucune des deux stratégies n'a montré sa supériorité sur nos jeux de données. Finalement, nous avons proposé un modèle hybride (n°5) qui nous semble intéressant pour les cas que nous avons testés.
Les valeurs des paramètres pour les modèles et les estimateurs étant primordiales, nous espérons pouvoir

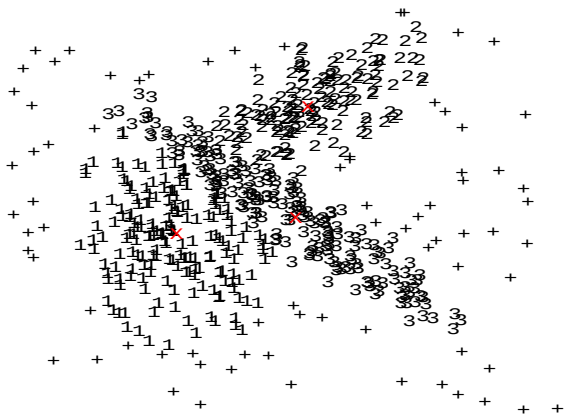


Figure 6: Modèle 2 - échantillon 2

automatiser et affiner leurs choix afin d'à améliorer nos résultats.

References

[BG98] Christophe Biernacki and Grard Govaert. Choosing models in model-based clustering and discriminant analysis. Technical Report RR-3509, Inria, Institut National de Recherche en Informatique et en Automatique, 1998.

[CD84] Gilles Celeux and Jean Diebolt. Reconnaissance de mlange de densit et classification. un algorithme d'apprentissage probabiliste: l'algorithme SEM. Technical Report RR-0349, Inria, Institut National de Recherche en Informatique et en Automatique, 1984.

[CLR95] N. A. Campbell, H. P. Lopuhad, and P. J. Rousseeuw. On the calculation of a robust S-estimator of a covariance matrix. Technical Report DUT-TWI-95-117, Delft University of Technology, Department of Technical Mathematics and Informatics, 1995.

[CS] Gilles Celeux and G. Soromenho. An entropy criterion for assessing the number of clusters in a mixture model. Technical Report RR-1874, Inria, Institut National de Recherche en Informatique et en Automatique.

[DH73] Richard Duda and Peter Hart. *Pattern Recognition and Scene Analysis*. John Wiley and Sons, 1973.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society series B*, 39:1–38, 1977.

[FR98a] Chris Fraley and Adrian Raftery. How many clusters ? which clustering method ? answers via model-based cluster analysis. Technical Report 329, Seattle: Department of Statistics, University of Washington, 1998.

[FR98b] Chris Fraley and Adrian E. Raftery. Mclust: Software for model-based clustering and discriminant analysis. Technical Report 342, Department of Statistics, University of Washington, 1998.

[Hub81] Peter J. Huber. *Robust Statistics*. John Wiley, New York, 1981.

[Kha97] Y. Kharin. Robustness of clustering under outliers. *Lecture Notes in Computer Science*, 1280, 1997.

[MP98] G. J. McLachlan and D. Peel. Robust cluster analysis via mixtures of multivariate t-distributions. *Lecture Notes in Computer Science*, 1451:658–??, 1998.

[WT90] Greg C. G. Wei and Martin A. Tanner. A Mont-Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *Journal of the American Statistical Association*, 85(411), September 1990.

[Wu83] C. F. J. Wu. On the convergence properties of the EM algorithm. *Annals of Statistics*, 11(1):95–103, 1983.

Table 2 : Résultats sur l'échantillon 1 (E1)

	Modèle 1	**Modèle 2**	**Modèle 3**	**Modèle 4**	**Modèle 5**
μ_1	7.07 8.91	7.12 8.84	7.15 8.88	7.18 8.89	7.13 8.90
Σ_1	5.35 −1.70 −1.70 2.29	1.70 −1.20 −1.20 1.14	0.74 −0.46 −0.46 0.49	0.35 −0.29 −0.29 0.29	0.76 −0.46 −0.46 0.48
ρ_1	-0.48	-0.86	-0.76	-0.90	-0.77
μ_2	6.07 2.74	6.07 2.74	6.14 2.73	6.25 2.75	6.14 2.74
Σ_2	4.48 0.00 0.00 1.14	3.06 −0.08 −0.08 0.89	1.98 −0.02 −0.02 0.47	0.98 0.04 0.04 0.14	2.00 −0.01 −0.01 0.49
ρ_2	0.00	0.05	-0.02	0.00	-0.02
μ_3	2.77 6.90	2.98 7.00	2.92 7.06	3.01 7.02	2.90 7.03
Σ_3	1.75 −0.39 −0.39 1.29	1.72 −0.20 −0.20 1.21	0.54 −0.09 −0.09 0.50	0.34 −0.02 −0.02 0.45	0.85 −0.13 −0.13 0.67
ρ_3	-0.26	-0.14	-0.17	-0.05	-0.17
Erreur	3.43 %	5.71%	2%	32.86 %	2.29%

Table 3 : Résultats sur l'échantillon 2 (E2)

	Modèle 1	**Modèle 2**	**Modèle 3**	**Modèle 4**	**Modèle 5**
μ_1	3.71 5.40	3.88 5.76	3.59 5.47	3.46 5.90	3.97 6.57
Σ_1	1.75 −0.3 −0.3 3.77	1.16 −0.21 −0.21 2.62	0.43 −0.22 −0.22 0.79	0.14 −0.04 −0.04 0.42	0.42 0.19 0.19 1.07
ρ_1	-0.13	-0.12	-0.38	-0.15	0.29
μ_2	7.10 10.04	6.91 9.91	7.03 10.05	6.78 9.90	7.04 10.06
Σ_2	2.32 1.04 1.04 1.47	1.58 0.93 0.93 1.28	0.52 0.25 0.25 0.39	0.39 0.21 0.21 0.25	0.37 0.21 0.21 0.30
ρ_2	0.57	0.65	0.56	0.68	0.64
μ_3	6.51 6.34	6.62 6.28	6.21 6.65	6.05 6.75	7.42 5.67
Σ_3	5.96 −3.80 −3.80 4.57	3.38 −2.99 −2.99 3.06	1.71 −1.40 −1.40 1.40	1.60 −1.30 −1.30 1.29	0.47 −0.48 −0.48 0.66
ρ_3	-0.73	-0.93	-0.91	-0.91	-0.86
Erreur	16.6 %	14.20%	24.20%	48.40%	20.4%