REVUE DE STATISTIQUE APPLIQUÉE

GILLES CELEUX
JEAN DIEBOLT

L'algorithme SEM : un algorithme d'apprentissage probabiliste pour la reconnaissance de mélange de densités

Revue de statistique appliquée, tome 34, n° 2 (1986), p. 35-52
<http://www.numdam.org/item?id=RSA_1986__34_2_35_0>

L'ALGORITHME SEM : UN ALGORITHME D'APPRENTISSAGE PROBABILISTE POUR LA RECONNAISSANCE DE MÉLANGE DE DENSITÉS

Gilles CELEUX ([1]), Jean DIEBOLT ([2])

(1) *(INRIA)*
(2) *(Paris 6)*

I. INTRODUCTION

1.1. Le problème traité

L'algorithme SEM (Stochastique, Estimation, Maximisation) a pour but de déterminer les composants d'un mélange fini de lois de probabilité, ainsi que le nombre lui-même de ces composants, par une approche d'apprentissage probabiliste.

Le problème de reconnaissance de mélanges est le suivant :

Soit un échantillon $E = (x_1, \ldots, x_N)$ d'une variable aléatoire X à valeurs dans $\mathbf{R}^d$, dont la loi est :

$$F = \sum \{p_k F_k \mid k = 1, K\} \text{ où :}$$

— Les F_k sont des mesures de probabilité sur $\mathbf{R}^d$;

— $\forall k = 1, K \quad 0 < p_k < 1 \quad \text{et} \quad \sum \{p_k \mid k = 1, K\} = 1.$

p_k est la probabilité qu'un point de l'échantillon suive la loi F_k.

Le problème consiste à estimer le nombre K de composants, les paramètres inconnus p_k ainsi que les lois inconnues F_k.

Ce problème n'a été étudié que dans le cas paramétrique :

Les F_k forment une famille paramétrée que nous noterons $\mathcal{F} = \{F(.\,, a), a \in \mathbf{R}^s\}$, l'application $a \longrightarrow F(.\,, a)$ étant injective. Nous nous plaçons dans ce cadre.

Ce problème ne peut se résoudre, dans toute sa généralité, que si le mélange de lois appartient à une famille de mélange identifiable.

Une famille $\mathcal{F}$ de mélange est dite identifiable si et seulement si :

$$\forall F \in \mathcal{F} \quad F = \sum \{p_k F(.\,, a_k) \mid k = 1, K\} = \sum \{p'_k F(.\, a'_k) \mid k' = 1, K'\}$$

implique :

$$K = K' \quad \text{et} \quad \forall k = 1, K \quad p_k = p'_k \text{ et } a_k = a'_k.$$

TEICHER [Te 63], YAKOWITZ et SPRAGINS [Ya Sp 68] ont caractérisé les mélanges identifiables. En particulier une condition nécessaire et suffisante est que la famille $\mathcal{F}$ soit un système libre sur $\mathbf{R}$.

Mots-clés : **Mélange de densités, Maximum de vraisemblance, Apprentissage probabiliste, Classification.**

Dans la plupart des cas les lois $F(.\,, a)$ admettent une densité $f(.\,, a)$. Le mélange s'écrit alors :

$$f(x) = \sum \{p_k f(x, a_k) \mid k = 1, K\} \text{ avec :}$$
$$\forall k = 1, K \quad 0 < p_k < 1 \quad \text{et} \quad \sum \{p_k \mid k = 1, K\} = 1$$

$f(x, a)$ est la densité de probabilité dépendant du paramètre vectoriel a de $\mathbf{R}^s$.

1.2. Plan de l'article

Au paragraphe 2, nous faisons une revue commentée des principales méthodes existantes.

Il en ressort que l'algorithme le plus efficace est l'algorithme EM qui cherche à maximiser la vraisemblance de l'échantillon relativement au modèle de mélange par un procédé itératif faisant intervenir deux étapes d'inférence statistique (estimation et maximisation). Nous le présentons au paragraphe 3 et détaillons ses caractéristiques :

- Il exige de connaître le nombre exact de composants du mélange.
- Les résultats obtenus dépendent de l'initialisation.
  - Il peut converger vers un point stationnaire de la vraisemblance de type col.
  - Il peut converger avec une lenteur rédhibitoire.

Du point de vue théorique, les nombreux travaux qui lui ont été consacrés n'ont fourni des preuves de convergence que sous des hypothèses difficiles à vérifier en pratique (cf. paragraphe 3).

Au paragraphe 4, nous présentons l'algorithme SEM qui s'inspire de l'algorithme EM en lui adjoignant une étape d'apprentissage probabiliste.

Nous précisons les caractéristiques de cet algorithme et la forme de ses résultats.

Au paragraphe 5 nous présentons des utilisations de l'algorithme SEM sur des données simulées.

Au paragraphe 6 nous présentons une note sur l'utilisation de l'algorithme SEM sur des données réelles en classification.

II. REVUE BIBLIOGRAPHIQUE

La méthode la plus ancienne est la méthode des moments ([Pe 94], [Co 64]).

Le principe de cette méthode consiste à résoudre les équations :

$$\int (x - E(x))^q f(x)\, dx = (1/N) \sum \{(x_i - \bar{x})^q \mid i = 1, N\}$$

q variant de 1 au nombre de moments nécessaires pour estimer les paramètres du mélange.

Dans le cas d'un mélange gaussien à deux composants QUANDT et RAMSEY [Qu Ra 78] ont proposé une méthode qui utilise les moments de la fonction génératrice E(exp(cx)) du mélange.

La méthode la plus usitée consiste à résoudre les équations de vraisemblance.

Les algorithmes les plus efficaces pour résoudre les équations de vraisemblance sont, à des variantes près, des algorithmes de type EM ([DLR 77]) (cf. [Sh 68], [Wo 70], [Re Wa 84]...).

Nous le présentons au paragraphe 3.

Cette dernière approche s'avère supérieure à la méthode des moments :

— La méthode des moments et ses variantes sont impraticables dans le cas multidimensionnel ou lorsque le nombre de composants est élevé.

— Des simulations ont montré que lorsque les composants du mélange étaient peu séparés les estimations fournies par la méthode du maximum de vraisemblance étaient meilleures que celles fournies par la méthode des moments.

Nous voulons aussi décrire deux approches assez récentes, intéressantes en elles-mêmes et relativement à notre approche.

Dans le cas où seules les proportions du mélange $p = (p_k, k = 1, K)$ sont inconnues, une approche bayésienne du problème a été proposée par SMITH et MAKOV ([Sm Ma 78]).

Partant d'une distribution a priori $P^o(p) = (p_k^o, k = 1, K)$ pour les p_k, l'approche consiste à résoudre de manière séquentielle (c'est-à-dire en actualisant l'estimation des p_k par une prise en compte séquentielle des points de l'échantillon) la formule de BAYES :

$$Pr(p/x_n) = f(x_n/p)\ Pr(p/x_{n-1}) / \int f(x_n/p)\ Pr(p/x_{n-1})\ dp$$

Malheureusement, il n'existe pas de statistique exhaustive pour p et les calculs deviennent inextricables.

Les auteurs proposent une approximation de la formule itérative en partant d'une distribution de DIRICHLET pour $P^o(p)$. Leur procédure converge p.s. vers la vraie valeur du paramètre.

Cette approche peut également s'appliquer lorsque seuls les paramètres $(a_k, k = 1, K)$ sont inconnus ([Ma Sm 76]).

Une autre voie pour contourner la difficulté de résolution de la formule de BAYES consiste à utiliser un algorithme d'apprentissage probabiliste.

AGRAWALA ([Ag 70]) a construit un algorithme séquentiel pour estimer une valeur a parmi les a_k, les autres étant connues ainsi que les proportions du mélange.

Partant d'une distribution a priori $P^o(a)$ pour a, il procède ainsi : x_n étant un nouveau point observé de l'échantillon, il l'affecte à l'un des composants par tirage aléatoire suivant la loi a posteriori d'appartenance de x_n aux composants. Soit l_n le numéro du composant obtenu par ce tirage aléatoire.

Il recalcule une nouvelle probabilité a posteriori pour a, sous l'hypothèse que x_n est effectivement issu du composant l_n.

Par un procédé analogue, SILVERMAN ([Si 80]) traite le problème d'estimation de p_1 pour un mélange à deux composants dont les paramètres a sont connus.

Partant d'une loi a priori $P^o(p_1) = \beta(b_0, c_0)$ (loi bêta de paramètres b_0 et c_0), il procède ainsi :

Un nouveau point observé x_n de l'échantillon est affecté au composant 1 ou 2 par tirage aléatoire suivant sa loi a posteriori d'appartenance à ces composants.

Si x_n est affecté au composant 1 alors $b_n = b_{n-1} + 1$.

Si x_n est affecté au composant 2 alors $c_n = c_{n-1} + 1$.

La loi a posteriori de p_1 est une loi $\beta(b_n, c_n)$ et le rapport $p_n = b_n/(b_n + c_n)$ est un estimateur de p_1.

En pratique, des simulations semblent montrer que le schéma d'apprentissage probabiliste donne des résultats meilleurs que le schéma quasi bayésien ([Ma Sm 76]).

Une approche radicalement différente ([Sc Sy 71], [Sch 76], [Sy 81]) consiste à rechercher une partition $P = (P_1, \ldots, P_K)$ telle que chaque classe P_k soit assimilable à un sous-échantillon suivant la loi $f(x, a_k)$.

Dans ce cadre, les algorithmes utilisés sont de type Nuées Dynamiques ([Di 80]).

Ils utilisent un critère de vraisemblance classifiante de la forme :

$$W(a, P) = \sum \{\text{Log } L(P_k, a_k) \mid k = 1, K\}$$

où $L(P_k, a_k)$ est la vraisemblance du sous-échantillon P_k pour la loi de densité $f(x, a_k)$.

Cette approche présente avant tout l'intérêt d'être rapide.

Du point de vue théorique, la méthode peut être vue ainsi : chaque point est issu de l'un des K composants du mélange.

Considérons les N paramètres inconnus suivants $(o(i); i = 1, N)$ avec : $o(i)$ = numéro du composant dont x_i est issu.

La méthode revient à rechercher les estimateurs du maximum de vraisemblance de ces paramètres. Or le nombre de paramètres à estimer augmente indéfiniment avec la taille N de l'échantillon. En conséquence les estimateurs du maximum de vraisemblance de ces paramètres ne sont pas convergents ([Br Wy 78]).

De plus, l'approche classification induit, en général, un biais dans l'estimation des paramètres du fait de la connexité des classes ([Ma 75]).

III. L'ALGORITHME EM

Cette méthode consiste à résoudre itérativement les équations de vraisemblance; le Log de la vraisemblance étant :

$$L(x_1, \ldots, x_N, a_1, \ldots, a_K, p_1, \ldots, p_K) = \sum \{\text{Log} \{ \sum (p_k f(x_i, a_k) | k = 1, K)\} | i = 1, N\}.$$

A partir d'une solution initiale $(p_k^0, a_k^0 ; k = 1, K)$, l'algorithme est le suivant :

Etape E (estimation) :

Pour $k = 1, K$; $i = 1, N$

calcul des $t_k^n(x_i) = p_k^n f(x_i, a_k^n) / \sum \{p_k^n f(x_i, a_k^n) | k = 1, K\}$

Etape M (maximisation) :

Pour $k = 1, K$ calcul de $p_k^{n+1} = (1/N) \sum \{t_k^n(x_i) | i = 1, N\}$

et résolution des équations pour $k = 1, K$; $j = 1, s$:

$$\sum (t_k^n(x_i) \, \partial \text{Log } f(x_i, a_k^{n+1}) / \partial a_{jk} | i = 1, N) = 0$$

où $a_k = (a_{jk} ; j = 1, s)$.

L'algorithme EM présente les caractéristiques suivantes :

— Il fonctionne pour un grand nombre de composants et dans le cas multidimensionnel;

— Il fournit, en général, de bons résultats si le nombre de composants est connu;

— Malheureusement, il converge extrêmement lentement. Cette lenteur peut rendre son utilisation rédhibitoire. C'est en particulier le cas lorsque la solution initiale est éloignée de la solution limite accessible.

Un problème de l'approche par le maximum de vraisemblance vient de ce que la fonction de vraisemblance n'est pas bornée. En conséquence, il se peut que l'algorithme EM dégénère vers une solution singulière (cf. [Ev Ha 81]).

Ces solutions interviennent avant tout lorsque l'un des composants a une probabilité d'apparition p petite. Plus précisément les singularités ont lieu lorsque pour ce composant $Np < d$ (d étant la dimension de l'espace contenant l'échantillon) ou plus généralement lorsque les points engendrés par l'un des composants sont tous situés dans un sous espace de codimension strictement positive.

L'apparition de tels phénomènes peut être liée à une surestimation du nombre de composants. Malheureusement, dans de tels cas, l'algorithme EM ne dégénère que très rarement et a plutôt tendance à rechercher une solution pour le nombre de composants surestimé.

Enfin, notons que REDNER et WALKER ([Re Wa 84]) ont montré que, sous des conditions assez générales :

— Pour N assez grand, l'unique solution convergente q^N des équations de vraisemblance existe p.s..

— Il existe une norme sur l'espace des paramètres pour laquelle la suite des itérés de l'algorithme EM converge vers q^N si la solution initiale q^o est suffisamment proche de q^N.

IV. NOTRE APPROCHE : UN ALGORITHME D'APPRENTISSAGE PROBABILISTE (ALGORITHME SEM)

Tous les algorithmes évoqués ci-dessus présentent les limitations suivantes :

— Le nombre K de composants est supposé connu.
— La solution obtenue dépend de la position initiale de l'algorithme.

L'algorithme que nous proposons ici répond en grande partie à ces deux limitations. Il utilise de manière complémentaire la construction de partitions et les étapes de l'algorithme EM.

Il s'agit en fait d'un algorithme EM auquel nous avons adjoint une étape d'apprentissage probabiliste. D'où son nom, algorithme SEM : Stochastique, Estimation, Maximisation.

4.1. Présentation

Au départ, on fixe le paramètre K majorant supposé du nombre de composants du mélange et un seuil c(N, d) compris entre 0 et 1.

Initialisation

En chaque point x_i, i = 1, N on choisit (en général au hasard) les probabilités initiales d'appartenance à l'un des composants :

Soient $t_k^o(x_i)$, k = 1, K avec :

$$0 < t_k^o(x_i) < 1 \quad \text{et} \quad \sum \{t_k^o(x_i) \mid k = 1, K\} = 1$$

Itération n (n > 0) :

Etape S (stochastique) :

On tire en chaque point x_i la v.a multinomiale

$$e^n(x_i) = (e_k^n(x_i);\ k = 1, K)$$

d'ordre un et de paramètres $(t_k^n(x_i);\ k = 1, K)$.

Les réalisations $e^n(x_i)$ définissent une partition $P^n = (P_1^n, \ldots, P_K^n)$ de l'échantillon avec :

$$P_k^n = \{x_i / e_k^n(x_i) = 1\}.$$

Si pour un certain k, card (P_k^n) est plus petit que Nc(N, d) l'algorithme est re-initialisé.

Sinon

Etape M (maximisation)

On calcule les estimations du maximum de vraisemblance $q_k^{n+1} = (p_k^{n+1}, a_k^{n+1})$ des paramètres du mélange sur la base des sous-échantillons $(P_k^n, k = 1, K)$.

On a :

$$p_k^{n+1} = (1/N) \sum \{e_k^n(x_i) | i = 1, N\}$$

L'estimation des a_k^{n+1} dépend bien sûr de la famille paramétrée, posée a priori, des composants du mélange.

Remarque

Si les paramètres a_k, $k = 1, K$ n'admettent pas d'estimateur du maximum de vraisemblance, on calculera des estimateurs qui améliorent la vraisemblance comme il est fait dans [Sch 76] pour un mélange de lois gamma ou plus généralement dans l'algorithme GEM ([DLR 78]).

Dans le cas où l'espérance m_k ou la matrice de variance $\sum_k$ sont des constituants des paramètres (cas de mélanges gaussiens, de Poisson, d'exponentielles,..) les estimations à l'itération n sont les suivantes :

$$m_k^{n+1} = \sum \{e_k^n(x_i)\, x_i | i = 1, N\} / \sum \{e_k^n(x_i) | i = 1, N\}$$

$$\sum_k^{n+1} = \sum \{e_k^n(x_i)\, (x_i - m_k^{n+1})\, (x_i - m_k^{n+1})' | i = 1, N\} / \sum \{e_k{}^n(x_i) | i = 1, N\}.$$

Etape E (estimation)

A partir des $q_k^{n+1} = (p_k^{n+1}, a_k^{n+1})$, on calcule :

Pour $k = 1, K$; $i = 1, N$

$$t_k{}^{n+1}(x_i) = p_k{}^{n+1} f(x_i, a_k{}^{n+1}) / \sum \{p_k{}^{n+1} f(x_i, a_k{}^{n+1}) | k = 1, K\}.$$

4.2. Caractéristiques de cette approche

A la stabilité de l'algorithme, on obtient non pas une seule partition mais une classe de partitions statistiquement admissibles pour les estimations des paramètres du mélange. Ces estimations sont précises (cf. les simulations du paragraphe 5) et asymptotiquement sans biais (cf. [Ce Di 84]).

Le type de convergence obtenue est une convergence en loi correspondant à la stationnarité de la suite des estimés (p^n, a^n) (cf. [Ce Di 84]).

Par ailleurs, les perturbations introduites à chaque itération par les tirages aléatoires empêchent la convergence vers un maximum local instable de la vraisemblance comme cela peut être le cas pour l'algorithme EM.

Cet algorithme fournit en général le nombre exact de composants pourvu que le paramètre K en soit bien un majorant. Ce point sera illustré au paragraphe 5.

Enfin il converge notablement plus rapidement que l'algorithme EM quelle que soit la configuration initiale. Les tirages aléatoires l'empêchent de « stationner » trop longtemps loin de la solution limite.

4.3. Mise en œuvre de l'algorithme SEM

Dans les applications présentées aux paragraphes 5 et 6, nous avons simplifié la procédure :

— D'une part, on a pris $c(N, d) = \frac{d}{N}$.

— Lorsque pour un certain k on a

$$\text{card}\,(P_k^n) \leqslant Nc(N, d) = d$$

on supprime le composant numéro k et l'algorithme continue sur la base de $(K-1)$ composants.

En pratique, après que le nombre de classes se soit stabilisé, on édite à la stationnarité la moyenne et la variance de chacune des lois marginales de $q^n = (p^n, a^n)$.

Nous ne disposons pas actuellement de procédure statistique pour déterminer le nombre minimum d'itérations à partir duquel on peut considérer que la suite des paramètres acquiert son comportement stationnaire (test de début d'enregistrement des résultats pour le calcul de la moyenne et de l'écart-type des lois marginales des q^n).

Pour l'instant, nous faisons tourner l'algorithme suffisamment longtemps pour être assuré d'avoir atteint l'état stationnaire (phase d'apprentissage). Nous faisons ensuite tourner l'algorithme à partir de cet état stationnaire durant r itérations et nous enregistrons les résultats (phase d'exploitation) :

A chaque itération, on obtient des valeurs $((p^i, a^i), i = 1, r)$ pour les paramètres et on calcule pour chaque paramètre sa moyenne empirique et sa variance empirique sur la base des r réalisations.

4.4. Note sur le comportement asymptotique de l'algorithme SEM

Dans le cas d'un mélange à deux composants où seul le paramètre p est inconnu, nous avons établi (cf [Ce Di 84] théorèmes 1 et 2) le résultat suivant :

Supposons que le seuil $c(N, d)$ est constant ou bien tend vers zéro assez lentement lorsque N tend vers l'infini.

Soit Ψ_N la loi limite de l'algorithme SEM et X_N une v.a. suivant cette loi.

Dans ce cas, il existe un unique point fixe de l'algorithme EM que nous noterons p_N.

Alors, il existe un nombre σ strictement positif fonction explicite des valeurs exactes des paramètres tel que :

$$\text{Pour tout a, lim } P\,[N^{1/2}\,(X_N - p_N) < a)] = \Phi_{0,\sigma}(a)$$

où $\Phi_{0,\sigma}$ est la fonction de répartition de la loi normale centrée et d'écart-type σ.

Nous conjecturons que ce résultat peut être étendu au cas général, sous les hypothèses introduites par Redner et Walker (cf. [Re Wa 84]] dans le but de prouver leurs principaux résultats.

5. SIMULATIONS

Dans toutes les simulations présentées, les algorithmes ont été initialisés en tirant au hasard suivant la loi uniforme les probabilités conditionnelles d'appartenance de chaque point de l'échantillon à l'un des composants du mélange.

Nous présentons les résultats sous forme d'un tableau. Chaque ligne de ce tableau est associée à l'un des paramètres du mélange. La première colonne (MOY) donne la moyenne des estimés à la stationnarité et la deuxième colonne (E.T.) donne son écart-type. Il s'agit de la moyenne et de l'écart-type des r réalisations après avoir atteint la stationnarité (cf. 4.3). Ces statistiques ont été calculées sur une suite de r = 100 réalisations. L'écart-type représente l'incertitude relative à l'estimation de chacun des paramètres. Les troisième (MOY-E.T.) et quatrième (MOY+E.T.) colonnes donnent les bornes de l'intervalle moyenne-écart-type, moyenne + écart-type. La cinquième colonne (VAL. EMP.) donne les valeurs empiriques obtenues sur l'échantillon tiré au départ des paramètres du mélange. Ces valeurs sont bien sûr connues et à découvrir par l'algorithme SEM. Elles sont accompagnées d'une étoile lorsque l'intervalle [MOY-E.T., MOY+E.T.] ne les contient pas. Enfin on a noté s la variance de chaque composant.

5.1. Dispersion des estimations et imbrication des composants

Dans ce paragraphe, nous avons considéré le mélange gaussien unidimensionnel suivant :

$$K=2,\ p_1=p_2=0.5,\ \sigma_1=\sigma_2=1,\ m_1=0 \text{ et } m_2=2,3,4.$$

La taille de l'échantillon est N=200. L'algorithme a été initialisé avec K=3.

L'observation des trois tableaux associés à ces trois valeurs de m_2 montre la diminution de l'écart-type (E.T.) lorsque l'écart entre les composants augmente.

$m_2 = 2$	MOY	E.T.	MOY-E.T.	MOY+E.T.	VAL-EMP
p_1	0.525	0.084	0.441	0.609	0.500
p_2	0.475	0.084	0.391	0.559	0.500
m_1	-0.322	0.179	-0.501	-0.143	-0.176
m_2	1.968	0.192	1.776	2.160	1.884
s_1	1.022	0.167	0.855	1.189	1.042
s_2	0.993	0.176	0.817	1.169	1.101

$m_2 = 3$	MOY	E.T.	MOY-E.T.	MOY+E.T.	VAL-EMP.
p_1	0.492	0.042	0.450	0.534	0.500
p_2	0.508	0.042	0.466	0.550	0.500
m_1	-0.201	0.127	-0.328	-0.174	-0.176
m_2	2.857	0.133	2.724	2.990	2.884
s_1	1.025	0.155	0.860	1.180	1.042
s_2	1.122	0.170	0.952	1.292	1.101

$m_2 = 4$	MOY	E.T.	MOY-E.T.	MOY+E.T.	VAL.EMP.
p_1	0.485	0.017	0.468	0.502	0.500
p_2	0.515	0.017	0.498	0.532	0.500
m_1	-0.223	0.064	-0.287	-0.159	-0.176
m_2	3.810	0.075	3.735	3.885	3.884
s_1	0.937	0.108	0.829	1.045	1.042
s_2	1.323	0.163	1.160	1.486	1.101 (*)

5.2. Mélanges gaussiens unidimensionnels

La taille de l'échantillon est $N = 200$ pour les trois premiers exemples et $N = 100$ pour le quatrième.

Exemple 1

Valeurs théoriques des paramètres :

$K = 2$, $p_1 = 0.25$, $p_2 = 0.75$, $m_1 = 0$, $\sigma_1 = 1$, $m_2 = 3$, $\sigma_2 = 1$

Initialisation : $K = 2$

Résultats

Ici les proportions sont très différentes, ce qui augmente la difficulté pour une bonne estimation (cf [Ev Ha 81]). L'essai présenté a été initialisé avec le bon nombre de composants, mais d'autres essais montrent que même dans ce cas il suffit de partir d'un majorant de ce nombre de composants.

	MOY	E.T.	MOY.-E.T.	MOY.+E.T.	VAL.EMP.
p1	0.225	0.032	0.193	0.257	0.250
p2	0.775	0.032	0.743	0.807	0.750
m1	-0.322	0.134	-0.456	-0.188	-0.199
m2	2.808	0.055	2.753	2.863	2.871
s1	0.887	0.170	0.717	1.057	1.008
s2	1.202	0.095	1.107	1.297	1.093

Exemple 2

Valeurs théoriques des paramètres :

$$K=2,\ p_1=0.8,\ p_2=0.2,\ m_1=0,\ \sigma_1=1,\ m_2=0,\ \sigma_2=3$$

Initialisation : $K=3$

Résultats

2 composants obtenus. Dans ce cas les composants sont très imbriqués et de plus les proportions sont très différentes et le nombre de composants initiaux n'est pas bon.

Naturellement, l'incertitude portant sur les estimations des paramètres est importante.

	MOY.	E.T.	MOY.-E.T.	MOY.+E.T.	VAL.EMP.
p1	0.223	0.055	0.168	0.278	0.200
p2	0.777	0.055	0.722	0.832	0.800
m1	0.086	0.045	0.041	0.131	0.062
m2	0.036	0.187	-0.151	0.223	0.133
s1	0.965	0.114	0.851	1.079	0.990
s2	6.635	1.145	5.490	7.780	6.964

Exemple 3

Valeurs théoriques des paramètres :

$$K=1,\ m=0,\ \sigma=1$$

Initialisation $K=2$

Résultats

Un seul composant obtenu. Ici il n'y a pas en fait de mélange mais une seule loi normale. L'algorithme SEM permet de découvrir ce fait.

Exemple 4

Valeurs théoriques des paramètres :

$K=4$, $p_1=p_2=p_3=p_4=0.25$

$m_1=2$, $\sigma_1=0.5$, $m_2=5$, $\sigma_2=0.5$, $m_3=0$, $\sigma_3=1$, $m_4=15$, $\sigma_4=2$

Cet exemple est tiré de [Ev Ha 81].

Initialisation $K=5$

Résultats

4 composants obtenus. Notons que cinq valeurs empiriques sont légèrement à l'extérieur de l'intervalle [MOY-E.T., MOY+E.T.]. Ce fait s'explique par le degré d'imbrication des composants et la taille assez faible des échantillons pour cet exemple (25 par composant).

	MOY.	E.T.	MOY.-E.T.	MOY.+E.T.	VAL.EMP.
p1	0.249	0.003	0.246	0.252	0.250
p2	0.260	0.006	0.254	0.264	0.250(*)
p3	0.250	0.017	0.233	0.267	0.250
p4	0.240	0.014	0.226	0.254	0.250
m1	1.900	0.006	1.888	1.912	1.904
m2	4.937	0.035	4.902	4.972	4.897(*)
m3	9.013	0.040	8.863	9.053	8.835(*)
m4	14.831	0.398	14.433	15.129	14.674
s1	0.161	0.009	0.152	0.170	0.164
s2	0.424	0.063	0.361	0.487	0.340(*)
s3	0.750	0.155	0.595	0.905	0.928(*)
s4	4.476	1.323	3.153	5.799	4.893

5.3. Mélanges gaussiens multi-dimensionnels

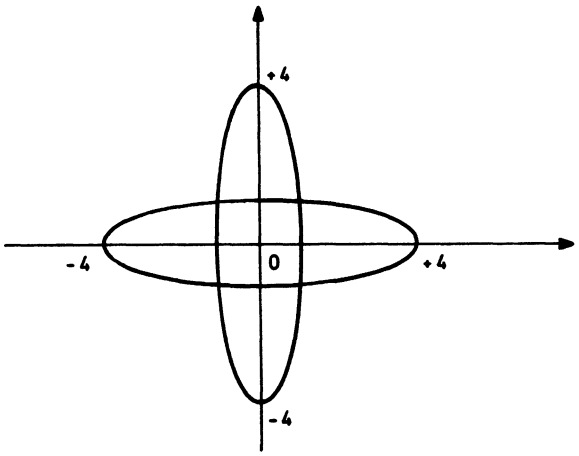
Un exemple dans R^2 :

Valeurs théoriques des paramètres :

$$K=2,\ p_1=p_2=0.5,\ m_1=(0,0),\ m_2=(0,0)$$

$$\Gamma_1=\begin{bmatrix}1 & 0\\ 0 & 4\end{bmatrix};\ \Gamma_2=\begin{bmatrix}4 & 0\\ 0 & 1\end{bmatrix}$$

La taille de l'échantillon étant $N=200$.



Ellipsoïdes d'inertie à 95 %

Initialisation $K=2$

Résultats

Les moyennes sont identiques, malgré cela les paramètres sont correctement estimés. Du point de vue classification par contre, il n'est pas possible de distinguer deux classes.

Pour ne pas allonger l'article, nous en restons là pour la présentation de simulations. Le lecteur pourra se reporter à [Ce Di 84] pour trouver d'autres exemples, notamment pour un mélange gaussien en dimension cinq et un mélange de lois de POISSON.

	MOY.	E.T.	MOY.-E.T.	MOY.+E.T.	VAL.EMP.
p1	0.526	0.071	0.455	0.597	0.500
p2	0.474	0.071	0.403	0.545	0.500
m_1^x	0.110	0.071	0.039	0.181	0.089
m_1^y	0.103	0.076	0.026	0.180	0.117
Γ_1^x	0.868	0.138	0.730	1.006	0.823
Γ_1^{xy}	0.221	0.084	0.137	0.305	0.020(*)
Γ_1^y	4.362	0.365	3.999	4.725	4.575
m_2^x	0.186	0.095	-0.281	-0.091	-0.212
m_2^y	0.000	0.084	-0.084	0.084	-0.022
Γ_2^x	4.572	0.460	4.112	5.032	4.405
Γ_2^{xy}	-0.301	0.114	-0.415	-0.187	-0.067(*)
Γ_2^y	1.015	0.232	0.783	1.247	0.967

5.4. Comparaison des résultats avec ceux obtenus par l'algorithme EM

Dans les exemples du paragraphe 5.1 et 1, 2, 4 du paragraphe 5.2, l'algorithme EM donne des résultats proches de ceux obtenus par l'algorithme SEM à condition que :

— L'algorithme EM soit initialisé à l'aide du nombre exact de composants.
— Les valeurs initiales des paramètres soient choisies assez près des valeurs exactes.

Ces deux conditions sont cruciales :

Partant d'une valeur inexacte de K

Dans l'exemple du paragraphe 5.1 avec $m_2=2$ en posant $K=3$, on obtient pour l'algorithme EM :

$p_1 = 0.322$, $p_2 = 0.329$, $p_3 = 0.349$
$m_1 = 0.660$, $m_2 = 1.008$, $m_3 = 0.887$
$s_1 = 2.070$, $s_2 = 2.128$, $s_3 = 2.136$

Et dans l'exemple 3 du paragraphe 5.2 en posant $K=2$, on obtient :

$p_1 = 0.857$, $p_2 = 0.143$
$m_1 = -1.523$, $m_2 = 0.130$
$s_1 = 0.611$, $s_2 = 0.283$

Ainsi l'algorithme EM ne permet pas de déceler l'erreur faite sur le nombre de composants.

Partant de valeurs initiales éloignées des vraies valeurs

Dans l'exemple du paragraphe 5.1 avec $m_2=2$, en ayant posé $K=2$, nous avons initialisé l'algorithme EM en tirant au hasard les probabilités conditionnelles d'appartenance de chaque point de l'échantillon à l'un des composants du mélange comme nous l'avons fait systématiquement pour l'algorithme SEM. (Usuellement, nous avions initialisé l'algorithme EM par l'estimation des paramètres sur la base d'une partition obtenue après une itération de l'algorithme des centres mobiles).

Résultats

A l'itération 1 les valeurs des paramètres étaient (à 10^{-3} près):

$p_1=0.493$, $p_2=0.507$, $m_1=0.845$, $\sigma_1=2.006$, $m_2=0.862$, $\sigma_2=2.256$

La valeur de la vraisemblance était de -359.5672.

A l'itération 1000 les valeurs des paramètres étaient (à 10^{-3} près) :

$p_1=0.493$, $p_2=0.507$, $m_1=0.810$, $\sigma_1=2.125$, $m_2=0.896$, $\sigma_2=2.136$

La valeur de la vraisemblance était de -359.5158.

On voit donc qu'au bout de 1 000 itérations, les estimations des paramètres (loin des valeurs réelles) n'ont presque pas été modifiées. Dans un tel cas, l'algorithme EM demandera un nombre d'itérations immense pour atteindre des estimations correctes des paramètres.

En revanche, il donne des résultats plus fiables pour de petits échantillons. Ainsi pour l'exemple 4 du paragraphe 5.2, si l'on fait passer la taille de l'échantillon de 100 à 60, les résultats par l'algorithme EM ne sont pas notablement modifiés. Mais par l'algorithme d'apprentissage probabiliste, plusieurs essais ont montré qu'environ une fois sur trois l'un des quatre composants disparaissait. Dans les autres cas les résultats sont analogues à ceux obtenus par l'algorithme EM.

Ce fait vient de ce que, pour de petits échantillons, les aléas introduits prennent trop d'importance.

Dans le même ordre d'idées, si les moyennes des composants sont très proches relativement à la dispersion, les caractéristiques de ces dispersions étant les mêmes pour tous les composants, alors l'algorithme SEM ne permet de discerner des composants différents que si la taille N de l'échantillon est assez grande. Ainsi, nous avons repris l'exemple du paragraphe 5.1 avec $m_2=1$ et toujours $N=200$; l'algorithme SEM a conclu à l'existence d'un seul composant.

6. UTILISATION DE L'ALGORITHME SEM EN CLASSIFICATION

L'exposé d'applications à des problèmes concrets de classification demanderait de trop longs développements pour trouver place ici. Le lecteur intéressé trouvera deux applications dans [Ce Di 84]. D'autres applications sont disponibles auprès des auteurs.

Dans ce paragraphe, nous nous contentons d'indiquer quelques observations sur l'utilisation de l'algorithme SEM en classification.

Notons, tout d'abord, que les algorithmes usuels de partitionnement peuvent être vus comme des algorithmes de reconnaissance de mélanges gaussiens (cf. [Sc Sy 71], [Ce Di 84]).

Habituellement, en classification on recherche des groupes homogènes et bien séparés. Les enveloppes convexes des classes obtenues sont disjointes. Si la structure réelle en classes des données ne correspond pas à ce modèle, les résultats obtenus par les méthodes usuelles de classification sont peu fiables.

L'algorithme SEM est utile dans les cas où la structure en classes est faible, dans le sens où les enveloppes convexes des classes à découvrir empiètent les unes sur les autres, même de manière importante.

D'une part, comme le montrent les simulations, il permet d'analyser de telles structures, d'autre part, il en fournit un mode de représentation plus pertinent qu'une partition.

Il s'agit d'une classification partielle au seuil α ($1/2 \leqslant \alpha \leqslant 1$) obtenue ainsi :

Nous sélectionnons les probabilités $t_k(x_i)$ d'affectation des points x_i à chaque composant k, associées aux valeurs des paramètres qui, en régime stationnaire, optimisent la fonction de vraisemblance. Nous définissons la classe partielle C_k associée au $k^{ième}$ composant par :

$$C_k = \{x_i / t_k(x_i) \geqslant \alpha\}$$

Une telle classification partielle n'est pas en général une partition car il peut exister des points x_i n'appartenant à aucune classe C_k.

Dans les cas que nous avons traités, ou bien la structure en classes des données était suffisamment forte pour que les classifications partielles soient proches d'une partition, ou bien au contraire la structure en classes des données était faible, auquel cas les classifications partielles à différents seuils restituaient ce fait en faisant apparaître des ilôts caractéristiques de chaque composant.

RÉFÉRENCES BIBLIOGRAPHIQUES

[Ag 70] AGRAWALA. — "Learning with a probabilistic teacher" IEEE. Information theory, Vol. 16, nu. 4.

[Br Wy 78] BRYANT — WYLLIAMSON. — "Asymptotic behaviour of classification maximum likelihood estimates". *Biometrika 78,* Vol. 68.

[Ce Di 84] CELEUX, DIEBOLT. — « Reconnaissance de mélange de densité et classification, un algorithme d'apprentissage probabiliste : l'algorithme SEM ». *Rapport de recherche INRIA n° 349.*

[Co 64] COOPER. — "Non supervised adaptative signal detection and pattern recognition". *Information and Control,* Vol. 7.

[Da 69] DAY. — "Estimating the components of a mixture of normal distributions". *Biometrika 69,* Vol. 56.

[Di 80] DIDAY et collaborateurs. — *Optimisation en classification automatique.* Editeur : INRIA.

[DLR 77] DEMPSTER — LAIRD — RUBIN. — "Maximum likelihood from incomplete data via the EM algorithm". *JRSS.B.,* Vol. 39.

[Ev Ha 81] EVERITT — HAND. — *Finite mixture distributions.* Chapman and Hall.

[Ma 75] MARRIOTT. — "Separating mixtures of normal distributions". *Biometrika,* 31.

[Ma Sm 76] MAKOV — SMITH. — "Quasi Bayes procedures for unsupervised learning". *Proc. IEEE. Conf. on Decision and Control.*

[Pe 94] PEARSON. — "Contribution to the mathematic theory of evolution". *Philos. Trans. Soc.,* nu. 185 (1894).

[Qu Ra 78] QUANDT, RAMSEY. — "Estimating mixtures of normal distributions and switching regression". *JASA,* Vol. 73.

[Re Wa 84] REDNER — WALKER. — "Mixture densities, maximum likelihood and the EM algorithm". *SIAM Review,* Vol. 26, No. 2, April 84.

[Sch 76] SCHROEDER. — "Analyse d'un mélange de distribution de probabilité de même type". *RSA,* Vol. 24, nu. 1.

[Sc Sy 71] SCOTT — SYMONS. — "Clustering methods based on likelihood ratio criteria". *Biometrics,* Vol. 27.

[Sh 68] SHLEZINGER. — "An algorithm for solving the selforganization problem". *Cybernetics,* nu. 2.

[Si 80] SILVERMAN. — "Some asymptotic properties of the probabilistic teacher" IEEE. *Information theory,* Vol. 26, nu. 2.

[Sm Ma 78] SMITH — MAKOV. — "A quasi Bayes sequential procedure for mixtures". *JRSS. B.,* Vol. 40, nu. 1.

[Te 62] TEICHER. — "Identifiability of finite mixture". *Ann. Math. Statist.,* Vol. 34.

[Sy 81] SYMONS. — "Clustering criteria and multivariate normal mixtures". *Biometrics,* Vol. 37.

[Wo 70] WOLFE. — "Pattern clustering by multivariate mixture analysis". *Multiv. Behav. Res.,* Vol. 5.

[Ya Sp 68] YAKOWITZ — SPRAGINS. — "On the identifiability of finite mixtures". *Ann. Math. Statist.,* Vol. 39.