Objectif

Montrer le fonctionnement de la classification (typologie) avec l'algorithme EM de TANAGRA.

Les **modèles de mélanges** traduisent une fonction de densité régissant la distribution de données à l'aide d'une combinaison linéaire de fonctions de densité élémentaires. L'approche la plus connue est le **modèle de mélange gaussien** où les densités élémentaires sont des lois normales multidimensionnelles.

Cette technique peut être utilisée pour décrire la distribution des données en classification automatique. Chaque classe (groupe, cluster, etc.) est décrite par une loi de distribution normale, paramétrée par son centre de gravité et sa matrice de variance covariance. Pour estimer les paramètres des distributions élémentaires, l'algorithme EM (**Expectation-Maximization**) est certainement le plus connu. L'objectif est de maximiser la log-vraisemblance de l'échantillon de données compte tenu d'un nombre de cluster défini au préalable.

Fichier

Pour illustrer le fonctionnement du composant, nous utilisons des données synthétiques[1] décrites dans le plan. Nous distinguons nettement les deux lois de distributions distinctes, l'enjeu de la typologie est de réussir à les circonscrire au mieux.

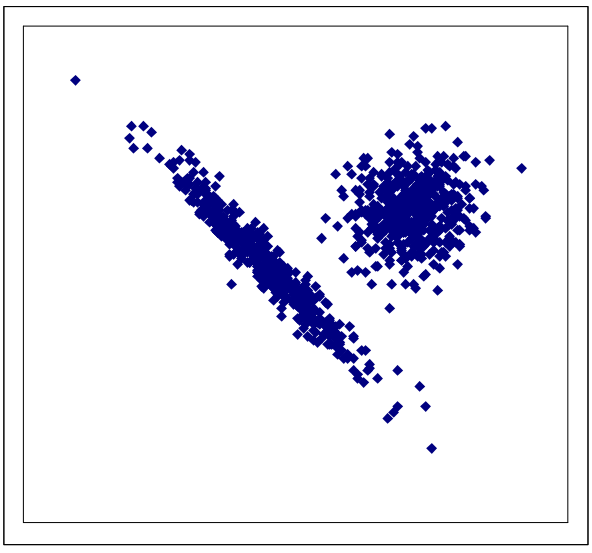


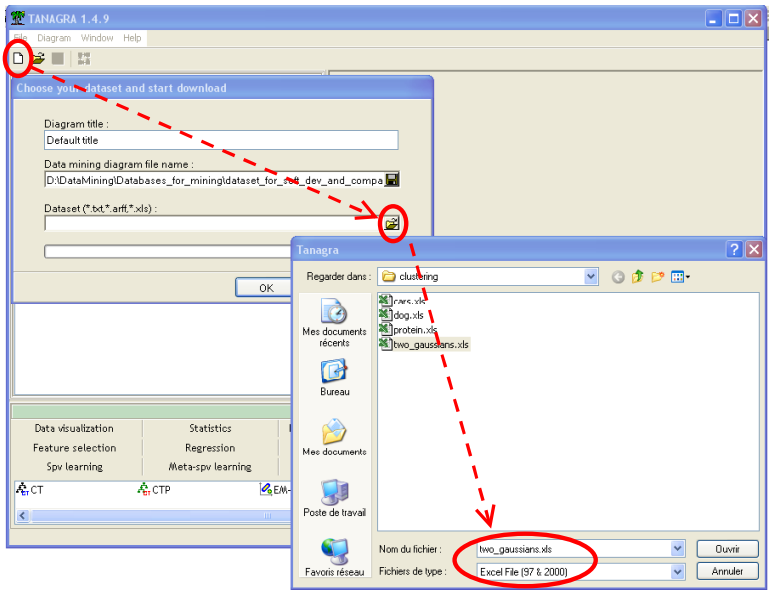
Figure 1 : Deux lois de distributions normales (distinctes et de formes très différentes) dans le plan

[1] Ces données proviennent de la distribution gratuite « FAST EM Clustering » de AUTONLAB (http://www.autonlab.org/autonweb/10466.html). Il sera ainsi possible de comparer les résultats par la suite.

Classification avec les modèles de mélange

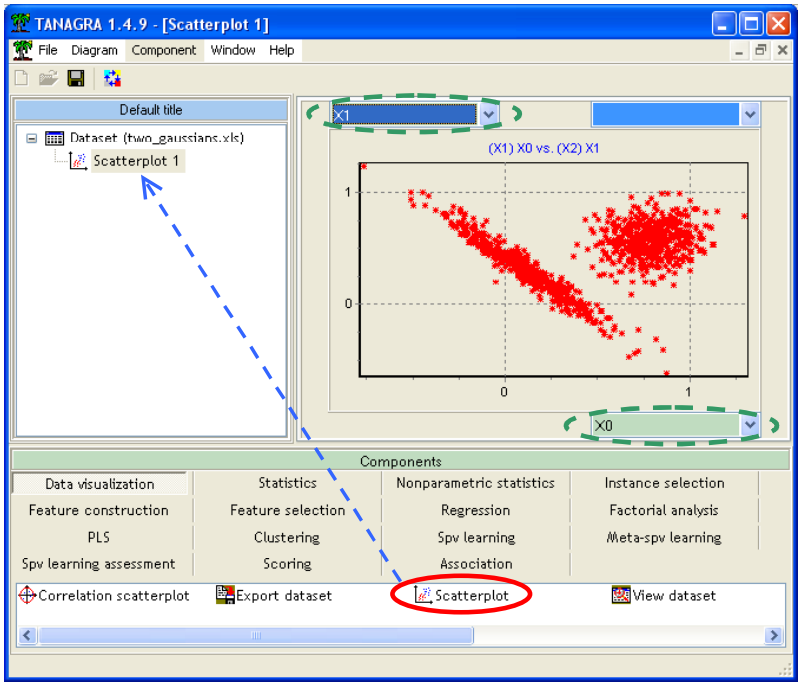
Charger les données

Nous créons un nouveau diagramme (FILE/NEW) et importons le fichier TWO_GAUSSIANS.XLS.



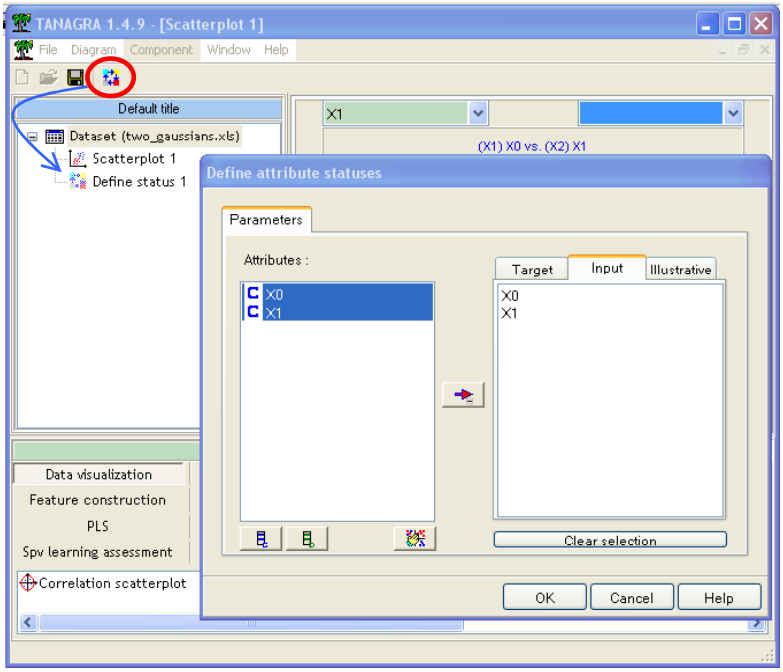
Représenter le nuage de points

Pour obtenir la représentation du nuage de points dans le plan dans TANAGRA, nous ajoutons le composant SCATTERPLOT (onglet DATA VISUALIZATION). Nous plaçons en abscisse la variable X0, et en ordonnée X1. Nous distinguons nettement deux blocs de points que la classification devrait mettre en exergue.



Sélectionner les variables

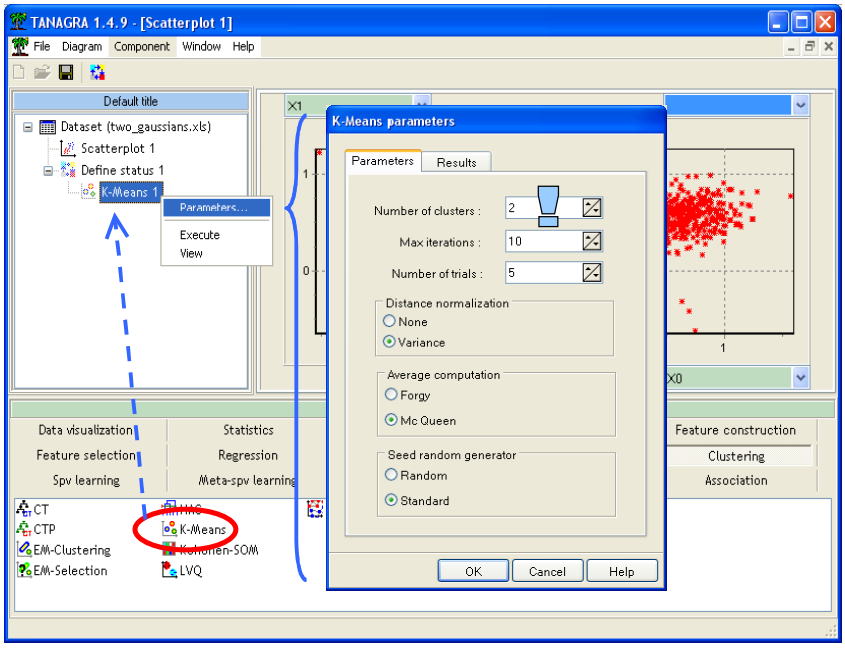
A l'aide du composant DEFINE STATUS, nous définissons comme variables d'études (INPUT) les deux variables qui composent le fichier.



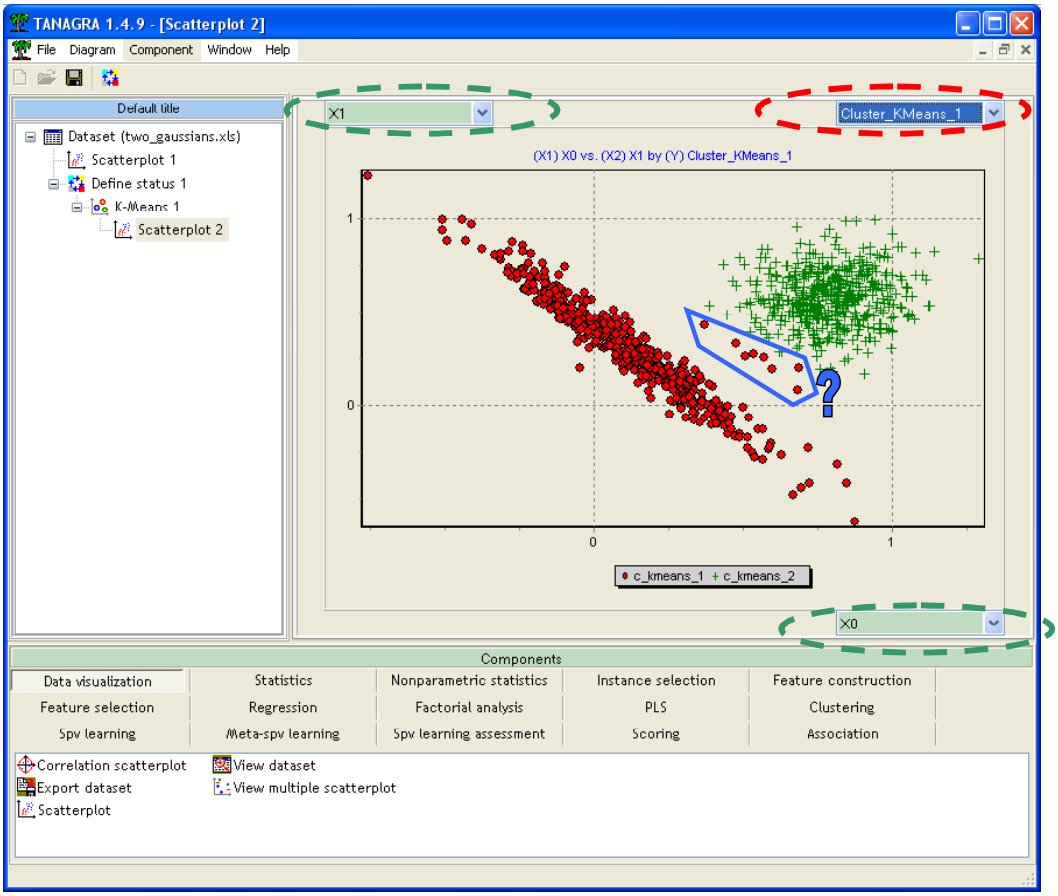
Typologie avec les K-MEANS

Dans un premier temps, nous construisons une typologie avec la méthode des K-MEANS. Elle nous servira de référence pour évaluer les résultats de l'algorithme EM.

Nous insérons le composant K-MEANS (onglet CLUSTERING) dans le diagramme. Nous le paramétrons de manière à produire 2 classes. Les autres paramètres ne sont pas modifiés.



Pour visualiser le regroupement, nous insérons de nouveau le composant SCATTERPLOT, et cette fois-ci nous illustrons les points à l'aide des groupes que K-MEANS a attribué à chaque observation.



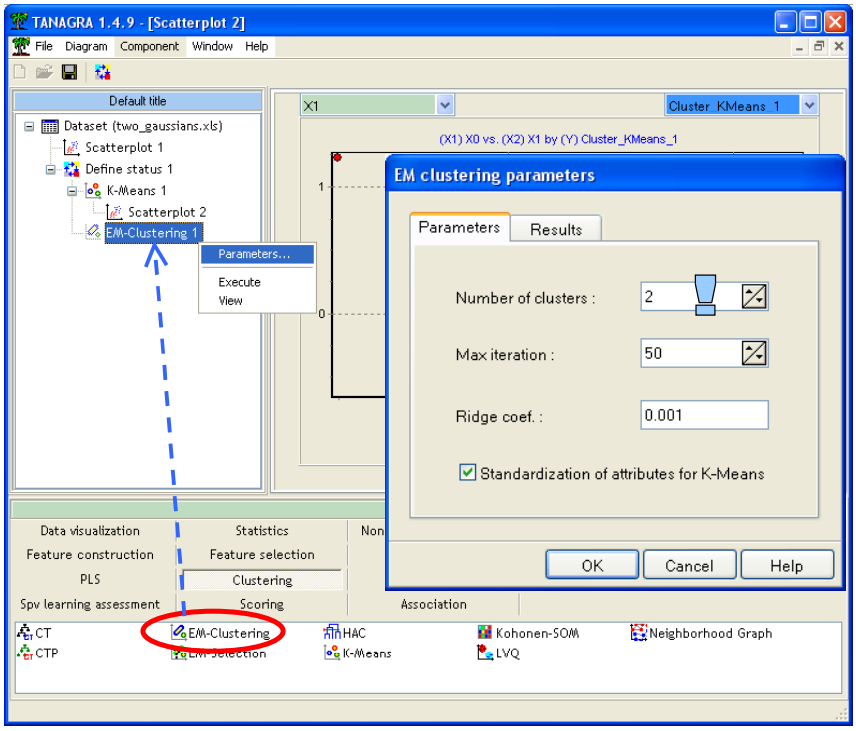
La méthode des K-MEANS trouve *grosso modo* les deux classes. Elle isole les deux nuages de points. Nous constatons néanmoins qu'une partie des observations est mal classée dans le nuage de droite. En tous les cas, leur affectation ne correspond pas à l'impression visuelle dans le plan.

Connaissant la méthode, ce résultat n'est pas étonnant. L'approche K-MEANS est uniquement paramétrée par les centres de gravité des classes, elle ne tient pas compte de leur dispersion. Elle s'appuie en fait sur l'hypothèse selon laquelle les nuages de points ont la même forme sphérique. Ce qui est manifestement erroné dans notre exemple.

Typologie avec le modèle de mélange

L'algorithme EM permet de calculer les paramètres d'un modèle de mélange gaussien dans le cadre de la typologie. Nous plaçons en dessous du composant DEFINE STATUS 1 le

composant EM-CLUSTERING (onglet CLUSTERING). Nous le paramétrons de manière à produire 2 groupes.



Techniquement, le composant initialise les groupes à l'aide des K-MEANS, puis optimise itérativement la vraisemblance à l'aide de l'algorithme Expectation – Maximisation. La recherche est stoppée lorsqu'il y a convergence c.-à.-d. lorsque la vraisemblance n'est plus améliorée ou lorsque l'on atteint la limite maximale du nombre d'itérations[2].

TANAGRA affiche les effectifs dans chaque cluster, les centres de classes et les indicateurs de qualité du partitionnement.

[2] Pour plus de détails sur les calculs, nous conseillons les sites http://fr.wikipedia.org/wiki/Algorithme_esp%C3%A9rance-maximisation et http://en.wikipedia.org/wiki/Expectation-maximization_algorithm

EM-Clustering 1

Parameters

EM parameters	
Clusters	2
Max Iteration	50
Ridge	0.001000
Seed random generator	Standard

Results

Clustering results

Clusters	2	
Cluster	**Description**	**Size**
cluster n°1	c_em_1	491
cluster n°2	c_em_2	509

Clustering quality criterion

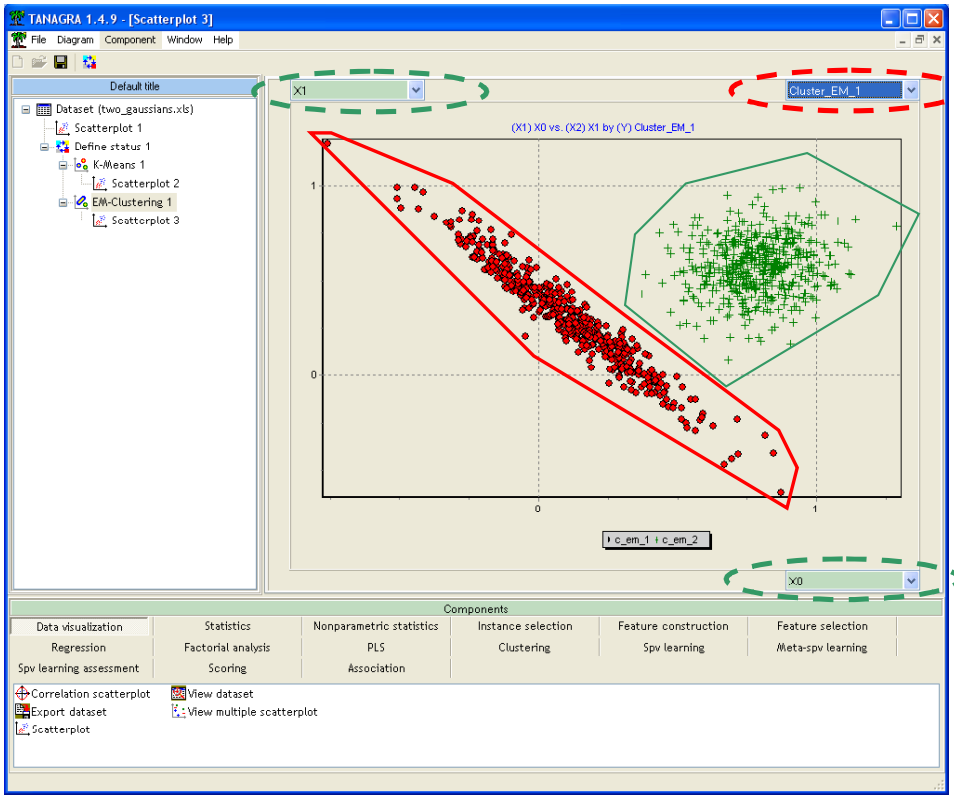
Criterion	Value
Log-likelihood	510.4171
AIC	-998.8341
BIC	-944.8488

Mean of clusters

Attribute	Cluster_1	Cluster_2
X0	0.1121	0.7862
X1	0.3013	0.5745

Computation time : 16 ms.

Pour évaluer la typologie proposée, nous plaçons de nouveau un composant SCATTERPLOT, nous illustrons cette fois-ci les points à l'aide de la nouvelle variable définie par la classification EM.



Le résultat est en accord avec notre intuition visuelle cette fois-ci. Le calcul tient compte des centres de classes toujours, mais également de la forme des nuages de points à travers la matrice de variance co-variance.

En conclusion, nous dirons que le modèle de mélange gaussien est certainement plus puissant que les K-MEANS. Mais cela se paie : les calculs sont plus complexes, l'occupation mémoire est plus importante et, le nombre de paramètres augmentant très vite avec le nombre de classes et la dimension de représentation, le sur-apprentissage nous guette.

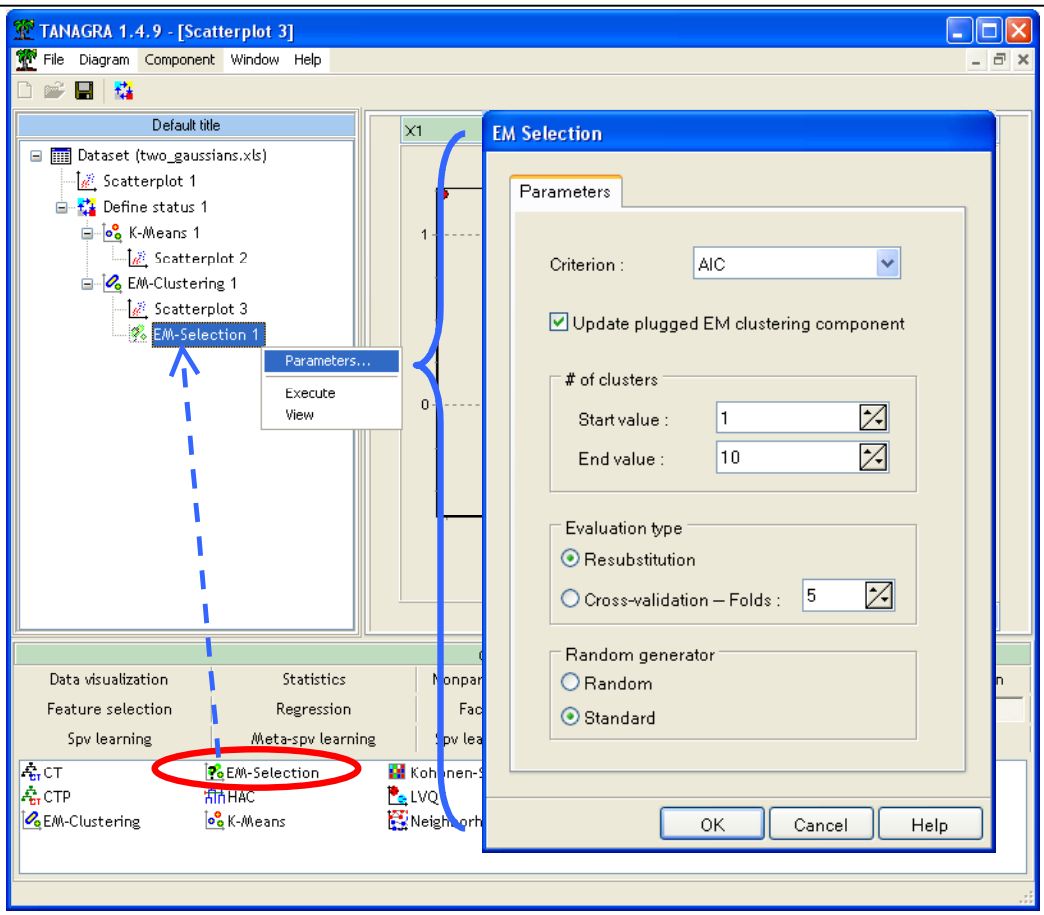
Détermination automatique du nombre de classes

Une question cruciale est récurrente en classification automatique : comment déterminer le bon nombre de classes ?

Avec le modèle de mélange gaussien, nous disposons d'une grandeur à optimiser : la vraisemblance (le logarithme de la vraisemblance dans la pratique). Il est possible de rechercher la solution « optimale » en testant différents nombres de groupes, par exemple en testant un nombre de clusters allant de 1 à 10.

Cette technique, assez simple à mettre en œuvre, présente un inconvénient rédhibitoire : la vraisemblance augmente mécaniquement avec le nombre de classes. De fait la solution « optimale » est connue d'avance, c'est celle qui correspond au nombre de classes le plus élevé dans les solutions testées. Pour palier cet écueil, nous pouvons introduire deux variantes : utiliser un critère qui tient compte de la complexité du modèle, les critères AIC (Akaike) et BIC (Bayesian Information Criterion de Schwartz) semblent tout à fait indiqués ; utiliser la validation croisée pour obtenir une évaluation plus réaliste de la vraisemblance.

Le composant EM-SELECTION permet de tester différentes valeurs du nombre de clusters. Nous devons le placer à la suite du composant EM-CLUSTERING 1 dans le diagramme. Il va alors exécuter plusieurs fois ce composant en comparant les résultats obtenus. Plusieurs paramètres sont disponibles.



Dans cet exemple, nous cherchons à optimiser le critère AIC en resubstitution (sur le fichier d'apprentissage). Les valeurs testées vont de 1 à 10. Une option importante « UPDATE PLUGGED EM CLUSTERING COMPONENT », si elle est cochée, permet de mettre à jour le composant EM-CLUSTERING associé en lui affectant le nombre de classes optimal détecté.

L'exécution permet d'obtenir le tableau de résultats suivant. Les calculs rejoignent l'impression visuelle, la partition en deux classes semble la plus appropriée dans notre exemple, elle minimise bien le critère AIC.

EM-Selection 1

Parameters

Parameter	Value
Criterion	AIC
# clusters -- Start value	1
# clusters -- End value	10
Evaluation type	Resubstitution
Folds for CV	5
Random generator	Standard

Results

Criterion values w.r.t. # of clusters

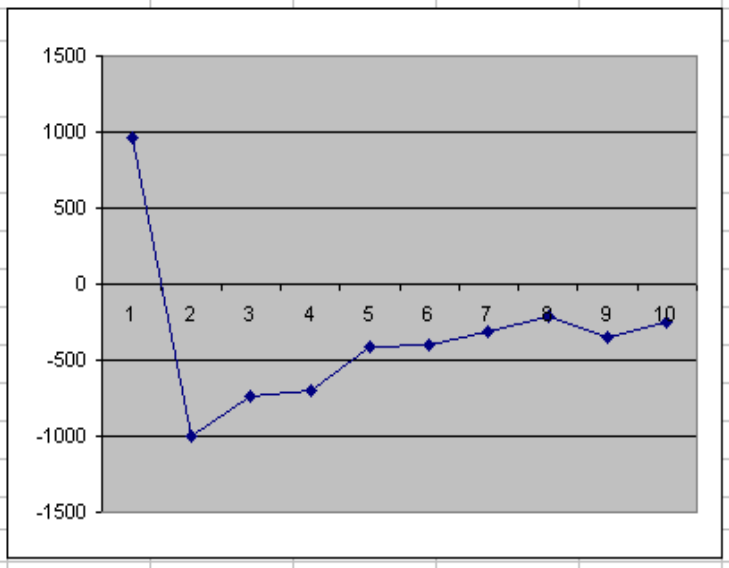
# of clusters	Criterion
1	966.711776
2	-998.834106
3	-733.068350
4	-702.453791
5	-413.147625
6	-395.647989
7	-316.241112
8	-211.967653
9	-350.100880
10	-249.646101

Computation time : 359 ms.
Created at 28/08/2006 10:03:10

Dans la pratique, il est conseillé de tester différents critères et surtout d'utiliser la validation croisée. Nous pouvons également nous référer au graphique reliant le critère calculé avec le nombre de classes. Cela permet de visualiser l'évolution de critère et de choisir sur un « plateau » le nombre de classes le plus faible en vertu du principe de parcimonie (Rasoir d'Occam). Dans notre exemple, nous avons copié le tableau de résultats dans un tableur (menu principal COMPONENT/COPY RESULTS), la partition en deux classes est indiscutablement la plus adéquate dans cet exemple.

Criterion values w.r.t. # of clusters

# of clusters	Criterion
1	966.711776
2	-998.834106
3	-733.06835
4	-702.453791
5	-413.147625
6	-395.647989
7	-316.241112
8	-211.967653
9	-350.10088
10	-249.646101



Computation time : 359 ms.

Conclusion

Les modèles de mélanges sont particulièrement puissants pour la classification. L'hypothèse de normalité n'est pas une limitation, l'approche est assez robuste et couvre en réalité une variété plus étendue de distributions. Assez curieusement pourtant, cette technique de classification est rarement disponible dans les logiciels de Data Mining.