PARTITIONS OPTIMISÉES SELON DIFFÉRENTS CRITÈRES : ÉVALUATION ET COMPARAISON

Alain GUÉNOCHE[1]

RÉSUMÉ - *Dans cet article, nous étudions et comparons des méthodes de partitionnement d'un ensemble d'éléments muni d'une distance, méthodes qui opèrent à partir de cette seule donnée. On cherche à construire une partition à nombre de classes fixé qui optimise un critère (séparation, diamètre ou inertie). Les méthodes étudiées fonctionnent sur le même principe : le nombre maximum de classes étant fixé, on construit une partition pour chaque valeur du nombre de classes variant de 2 au maximum. Tous les critères étudiés conduisent à des problèmes d'optimisation NP-difficiles. L'algorithme général combine des méthodes de descente et des métaheuristiques pour construire des partitions sous-optimales. Plusieurs façons d'évaluer la qualité des classes et de comparer ces partitions sont proposées ; elles sont indépendantes du critère optimisé et des cardinaux des classes. Elles permettent de choisir la partition la plus compatible avec une distance donnée. Par simulation de plusieurs types de distance (euclidienne, booléenne, ou distance d'arbre) on étudie les critères qui donnent en moyenne les meilleurs résultats.*

MOTS CLÉS - Méthodes de partitionnement, Optimisation combinatoire, Simulation.

SUMMARY - Partitions optimizing different criteria : Evaluation and Comparision

In this article, we study and compare partitionning methods applied to a distance matrix. Given the maximum number of classes and a criterion, we build one partition optimizing this criterion for each number of classes varying from 2 to this maximum. All the studied criteria lead to NP-hard problems. The general algorithm combines optimization and metaheuristic technics to build sub-optimal solutions. Several ways to evaluate the quality of the classes and to compare partitions corresponding to different criteria are proposed. They allow to chose the best partition fitting a distance matrix and, simulating several types of metric, to designate the criterion providing generally the best results.

KEYWORDS - Partitionning method, Combinatorial optimization, Simulations.

REMERCIEMENTS - Ce travail a été réalisé dans le cadre d'un contrat Bio-Informatique soutenu par les EPST. Il décrit plusieurs méthodes implémentées dans des programmes que l'on peut obtenir en s'adressant à l'auteur.

[1] Institut de Mathématiques de Luminy, 163 Av. de Luminy, 13288 Marseille Cedex 9, guenoche@iml.univ-mrs.fr

Cet article traite du partitionnement d'un ensemble X à N éléments muni d'une distance D. Indépendamment des espaces de représentation dans lesquels les éléments de X sont décrits, on peut toujours se ramener à cette donnée, même si le choix d'une distance appropriée n'est pas toujours évident. Rappelons qu'une partition de X est un ensemble de classes disjointes, dont l'union est X. On note $P = \{X_1, X_2, .. X_p\}$ une partition de X en p classes.

$$\forall\, i,j \text{ on a } X_i \cap X_j = \varnothing \qquad \text{et} \qquad \cup_{i=1,..p} X_i = X$$

On cherche à construire une partition à nombre de classes fixé qui optimise un certain critère. Ce problème n'est pas nouveau et, s'il fallait citer toutes les références, un numéro de la revue y suffirait à peine. S'il faut n'en citer qu'une, qui permettent en cascade d'accéder à toutes les autres, je renverrai à l'ouvrage de B. Mirkin [1996] dont un chapitre porte sur le partitionnement.

Il y a trois familles de critères "naturels" en Classification ; ce sont la séparation, l'homogénéité et la dispersion. Selon les premiers, une bonne partition présente des classes bien séparées ; on cherche à maximiser les *écarts* entre classes, qui sont fonctions des distances inter-classes. Selon les seconds, les classes sont les plus concises possible, on cherche à minimiser le *diamètre*, c'est-à-dire le maximum des distances intra-classes. Selon les troisièmes, on minimise une *fonction d'inertie*, la somme des carrés des écarts à un centre, qu'il soit réel ou virtuel.

Dans cet article, nous étudions un ou plusieurs critères spécifiques de chaque famille en appliquant systématiquement la même stratégie d'optimisation : un nombre maximum de classes est fixé et, à chaque étape, on construit une partition de X à k classes pour k variant de 2 au maximum indiqué. L'algorithme général combine une méthode de descente de type "k-means", suivie d'une méthode tabou, qui fait intervenir des choix aléatoires. Les partitions résultantes sont donc sous-optimales. Pour k plus grand que 2, les centres des nouvelles classes sont choisis à l'aide de la partition obtenue à k-1 classes.

Dans les trois premiers paragraphes, nous étudions respectivement les critères de séparation, d'homogénéité et de dispersion. Au total, dix critères sont retenus, et chacun permet de construire une partition. Pour les comparer, au paragraphe 4, nous définissons des indices de qualité, aussi bien pour les classes que pour les partitions. Ces derniers sont indépendants des critères optimisés ; ils permettent de choisir une des solutions proposées. Enfin au paragraphe 5 nous utilisons les indices de qualité des partitions pour déterminer, à l'aide de simulations, quels sont les critères d'optimisation les plus adaptés à différents types de distances.

Nous adoptons tout du long la terminologie de l'Analyse Combinatoire de Données, [Arabie & Hubert 1996] à savoir qu'une distance est considérée comme un graphe complet dont les arêtes sont pondérées par les valeurs de distance. Une arête entre deux éléments est aussi appelée un lien : il est dit *interne* si les éléments sont dans la même classe, *et externe* s'ils sont dans des classes différentes.

1 CRITÈRES DE SÉPARATION

On peut maximiser les écarts entre classes, selon plusieurs critères. Le plus simple est la *séparation*, c'est-à-dire la plus petite des distances inter-classes. On parcourt donc toutes les paires $\{X_k, X_l\}$ de classes et

$$\sigma(P) = \text{Min}_{x_i \in X_k,\, x_j \in X_l}\ D(x_i, x_j)$$

Il est bien connu que toutes les partitions de séparation maximum sont obtenues en supprimant les arêtes d'un arbre couvrant de poids minimum (minimum spanning tree) dans l'ordre décroissant des longueurs [Hubert, 1974] ; les classes sont alors les composantes connexes de ce graphe. Toutes les partitions de séparation maximum constituent une hiérarchie, celle que l'on obtient par la méthode ascendante du lien unique. Ces partitions sont optimales et, comme l'algorithme est de complexité polynomiale ($O(N^2)$), l'exécution est très rapide.

Mais pour un nombre de classes fixé, la partition optimale n'est pas nécessairement unique, notamment s'il y a des arêtes de l'arbre minimum qui sont de même longueur. De plus avec des valeurs de distance égales, il peut y avoir plusieurs arbres minimums et des partitions différentes ayant même valeur optimale. Pour éviter ce biais on considère les *composantes connexes des graphes seuils*. Un graphe seuil inférieur, pour une valeur s, est noté $G_{\leq s}$; il a pour sommets tous les éléments de X et pour arêtes les paires (x,y) dont la distance est inférieure ou égale à s. Au seuil de la plus longue arête d'un arbre minimum (quel que soit l'arbre, cette valeur est la même) le graphe seuil est connexe et il n'y a qu'une seule classe, X. Quand on fait décroître cette valeur, les différents graphes seuils ne sont plus connexes et font apparaître plusieurs classes et donc plusieurs partitions de séparation maximum pour ce nombre de classes.

De plus ces partitions sont construites à partir d'un très petit nombre de valeurs de distance (une seule valeur établit la séparation entre deux classes) et c'est pourquoi on maximise généralement d'autres critères qui utilisent un plus grand nombre de liens.

Appelons *cocycle*, l'ensemble des liens inter-classes.

$$\text{Cocycle} = \{ (x_i, x_j) \mid x_i \in X_k, x_j \in X_l \}$$

On cherche à maximiser les écarts entre classes, ces écarts étant calculés en norme L_1 ou L_2 au choix de l'utilisateur. Nous garderons le terme écart, noté $E(x,y)$, sachant qu'il peut être égal à $D(x,y)$ ou à son carré ; les notations des critères correspondent à la norme L_1.

Les critères les plus classiques sont :

La somme des écarts inter-classes.

$$\sum e\,(P) = \sum (x,y) \in \text{Cocycle}\ E(x,y)$$

Cette somme tend à faire intervenir le plus grand nombre de liens externes et produit donc des partitions en classes équilibrées ayant plus ou moins le même nombre d'éléments. En utilisant les carrés des distances, on atténue l'effet "classes équilibrées" qui reste néanmoins très sensible. Comme la somme des carrés des distances est une constante des données, ce critère est le complément de la somme des carrés des distances intra-classes qui représente une fonction d'inertie. Il revient au même de minimiser cette inertie (comme dans les méthodes de centre) que de maximiser la somme des carrés des distances inter-classes, comme nous le faisons.

La moyenne des écarts inter-classes.

$$\overline{\sum e}\ (P) = \frac{1}{|\,\text{Cocycle}\,|} \sum (x,y) \in \text{Cocycle}\ E(x,y)$$

Suivant ce critère, un singleton n'ayant que des distances fortes aux autres éléments réalisera le maximum. Il tend donc à séparer les éléments loin de tous les autres et produit souvent des classes très déséquilibrées. C'est donc une façon de mettre en évidence des éléments inclassables. On notera aussi que des partitions voisines, obtenues par des changements de classes de quelques éléments, produisent des variations infimes du critère de moyenne.

Pour le premier de ces critères, une partition en classes équilibrées fait intervenir $O(N^2)$ liens, alors que pour le second, une bipartition avec un singleton n'en fait intervenir que $O(N)$. Nous avons défini un nouveau critère qui utilise le même nombre de liens quels que soient les cardinaux des classes

La somme des plus petits écarts aux autres classes.

$$\overline{\sum \text{ppd}}\ (P) = \frac{1}{\text{N(p-1)}} \sum_{x \in X} \sum_{k=1,\,..\,p} [\ \text{Min}_{y \in X_k} E(x,y)\ ]$$

Pour chaque élément, on cherche son plus proche voisin dans chaque autre classe et l'on fait la moyenne de tous ces écarts. Dans une partition à p classes, on utilise p-1 valeurs de distance par élément, si bien que le nombre de liens utilisés est $N(p\text{-}1)$.

Quelle que soit la norme, tous les problèmes d'optimisation de ces critères, sur l'ensemble des partitions à nombre de classes fixé, sont des problèmes *NP*-difficile [Brucker 1978]. Nous avons adopté une méthode approchée qui commence comme une méthode de centres :

- On détermine un centre pour chaque classe. Initialement, on commence avec deux classes et les centres sont deux éléments dont la distance est maximum ; ils réalisent le diamètre. Au-delà ($p > 2$), on part de la meilleure partition trouvée à p-1 classes. Dans chacune, on retient l'élément dont la somme des distances aux éléments hors cette classe est maximum. On a donc p-1 centres et pour le dernier, on retient l'élément dont la plus petite distance à l'un de ces centres est maximum.
- On construit une partition initiale par une méthode de descente de type allocation-recentrage. A chaque itération, on affecte chaque élément à la classe du centre le plus proche ; puis, pour chaque classe, on détermine son "centre" comme une médiane, c'est-à-dire un élément dont la somme des distances aux autres éléments de cette classe est minimum. On effectue ces deux opérations tant que le critère est amélioré.
- Ensuite on applique une méthode de recherche Tabou [Glover, 1986] en deux parties.

  Affectations aléatoires : Pour chaque élément, on teste une affectation dans une autre classe choisie au hasard, et l'on calcule la variation du critère qui résulterait de ce seul changement. On effectue le changement de classe de l'élément qui maximise la nouvelle valeur, que le critère soit amélioré ou non. A partir de cette nouvelle affectation, on immobilise cet élément pendant un certain nombre d'itérations successives - le retour arrière est "tabou". Ce nombre d'itérations est un paramètre qui définit la longueur de la liste Tabou ; ici il est fixé à 3. Quand le critère est amélioré, on recommence les itérations à zéro. On teste jusqu'à $N(p\text{-}1)$ affectations aléatoires consécutives qui n'améliorent pas le critère.

  Meilleure affectation : Pour chaque élément, on cherche dans quelle autre classe il pourrait être placé au mieux et quelle serait la variation du critère. Puis, comme précédemment, on effectue le moins mauvais changement. Cette deuxième phase est appliquée jusqu'à N affectations après la dernière amélioration trouvée.

  A la fin, la meilleure partition trouvée au cours de ces explorations est éditée.

2. CRITÈRES D'HOMOGÉNÉITÉ

Il s'agit de construire des partitions dans lesquelles les classes sont les plus concises possible. Ce critère est tout à fait naturel quand elles correspondent à des zones

dans lesquelles on veut minimiser des temps d'accès ou des déplacements à l'intérieur de ces zones. Le paramètre important est :
- le diamètre d'une classe, égal à la plus grande dissimilarité intra-classe,

$$\delta(X_k) = \text{Max } D(x_i,x_j) \text{ pour } x_i \in X_k \text{ et } x_j \in X_k$$

- le diamètre d'une partition, égal au plus grand des diamètres de ses classes,

$$\Delta(P) = \text{Max}_{k=1,..\ p}\ \delta(X_k)$$

Les critères naturels sont le diamètre de la partition et la somme des diamètres des classes. Ce dernier tend à produire des classes réduites à un seul élément - qui ont un diamètre nul - et, bien que le problème à deux classes soit polynomial [Hansen & Jaumard 1987], ce critère n'est pas utilisé pour les problèmes réels. On cherche donc à minimiser $\Delta(P)$.

Construire une bipartition de diamètre minimum est un problème de complexité polynomiale. Par contre, dès que le nombre de classes est supérieur à 2, le problème est *NP*-difficile (Hansen & Delattre [1978]). Ces auteurs ont ramené le problème à celui d'une coloration optimale d'un graphe seuil supérieur (chaque couleur est une classe). Rappelons :
- qu'un graphe seuil supérieur (resp. supérieur ou égal) d'une distance D sur X a pour sommets X et pour arêtes les paires de sommets dont la distance est supérieure (resp. supérieure ou égale) au seuil;
- qu'une coloration des sommets d'un graphe affecte des couleurs différentes aux sommets adjacents ; cette coloration est optimale si le nombre de couleurs utilisées est minimum. Leur théorème peut être reformulé :

THÉORÈME [Hansen & Delattre, 1978] : Une partition en p classes est de diamètre minimum s, si et seulement si le graphe seuil $G_{>s}$ est p coloriable alors que le graphe $G_{\geq s}$ ne l'est pas.

On ne peut donc construire une partition optimale à plus de deux classes quand N est grand, et nous avons développé une stratégie également basée sur une méthode de descente suivie d'une phase d'optimisation stochastique.

2.1. Bipartition

Donc pour p=2, on construit une partition optimale (elle n'est généralement pas unique). Pour cela, nous utilisons :

THÉORÈME [Guénoche 1989, Monma & Suri 1989] : La partition en deux classes obtenue par coloration des sommets d'un arbre couvrant de poids maximum d'une distance est une bipartition de diamètre minimum.

On commence donc par construire un arbre maximum, avec le même algorithme de Prim que pour l'arbre minimum. Puis on affecte une couleur à un élément quelconque ; celles des autres en résultent. Cet algorithme est détaillé dans Guénoche et al. [1991].

2.2. Plus de 2 classes

Pour $p > 2$, on cherche une partition à p classes de petit diamètre. La méthode est en deux étapes :

- On part de la partition obtenue à l'étape précédente et l'on subdivise la classe de plus grand diamètre en deux sous-classes de diamètre minimum. On utilise la même procédure que ci-dessus, en construisant un arbre maximum de la classe considérée. Puis on détermine le diamètre de la partition initiale à p classes, réalisé par la paire (x,y) ; on a $\Delta = D(x,y)$.
- Ensuite on tente de minimiser ce diamètre par une procédure de transfert. On essaye de placer x ou y dans une autre classe, ce qui donne deux partitions. Soit Δ' le plus petit des diamètres obtenus, par exemple, en déplaçant x. On accepte les transformations pénalisantes, telles que $\Delta' > \Delta$, mais on interdit pendant un certain nombre d'étapes consécutives le déplacement de x, afin d'éviter les cycles de transferts. C'est donc encore une heuristique Tabou dans laquelle on ne déplace, à chaque itération, qu'une des extrémités d'un diamètre. Après chaque amélioration obtenue, on applique cette procédure au plus pN fois.

Du fait de cette procédure de transfert, les classes réalisées ne constituent pas une hiérarchie, alors que si elle était absente, ou toujours improductive, on retrouverait les mêmes résultats que la méthode de subdivision basée sur le critère du diamètre [Guénoche et al. 1991]

3. CRITÈRES DE DISPERSION

Ils tendent à minimiser l'inertie de la partition c'est-à-dire la somme des inerties des classes. Pour des données dans un espace euclidien, l'inertie fait référence au centre de gravité de chaque classe mais, en l'absence d'espace de représentation des éléments - on a juste une distance - il faut définir une notion de centre. Nous avons retenu trois fonctions d'inertie, l'une par rapport à un centre réel, et les deux autres par rapport à un centre virtuel, dont on estime les distances aux éléments de X.

3.1. Inertie par rapport à un centre réel

Pour chaque classe de plus de 2 éléments, le centre est déterminé par sa médiane, c'est-à-dire l'élément dont la somme des distances aux autres est minimum. Les centres sont donc choisis parmi les éléments à classer. Puis, classiquement, l'inertie de chaque classe est égale à la somme des carrés des distances à ce centre.

$$\imath(X_k) = \text{Min}_{\,x_i \in X_k} \sum_{x_j \; X_k} D^2(x_i,x_j)$$

$$I_r(P) = \sum_{k=1,..p} \imath(X_k)$$

3.2. Inertie par rapport à un centre virtuel

Pour calculer l'inertie d'une classe, il n'est pas nécessaire de définir un centre ; il suffit de connaître pour chaque élément x_i sa distance L_i à ce centre. Ceci revient à déterminer une approximation de la distance réduite à cette classe par une distance à centre comme sur la Figure 1 ci-dessous :

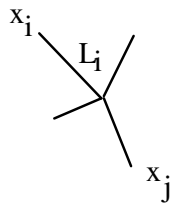

Figure 1 : Distance à centre représentée par un arbre en étoile

Cette distance *Dc* est déterminée par les longueurs des arêtes d'un arbre en étoile qui minimisent l'écart quadratique entre *D* et *Dc* :

$$\sum x_i, x_j \in X_k \; [\, D(x_i,x_j) - (L_i+L_j) \,]^2$$

La racine de cet arbre tient lieu de centre de la classe ; il est donc virtuel et la fonction d'inertie est la somme des carrés des longueurs de branches.

$$I_V(P) = \sum x_i \in X \, (L_i)^2$$

L'approximation, au sens des moindres carrés, d'une distance donnée par une distance à centre est un cas particulier de l'approximation par une distance d'arbre. Les méthodes de projection dans le cône des distances à centre sont itératives [Fichet 1986, Gascuel & Levy 1996], soit elles demandent la résolution d'un système linéaire sous contraintes positives [Guénoche 1986] ; c'est cette seconde approche que nous avons implémentée. Le système linéaire est très simple : tous ses coefficients sont égaux à 1 sauf sur la diagonale dont toutes les valeurs sont égales à *N*-1 ; il est donc à diagonale dominante et le schéma itératif de Gauss-Sidel converge toujours.

Cette nouvelle fonction d'inertie est plus "juste", puisqu'une classe à deux éléments {*x,y*} donnera $D^2(x,y)$ comme inertie réelle, mais la moitié comme inertie par rapport à un centre virtuel situé au "milieu" de cette arête. De même pour une classe composée de trois éléments équidistants à distance *d*, l'inertie réelle est égale à $2d^2$, alors que l'inertie virtuelle est égale $3\ d^2/4$.

3.3. Inertie moyenne

Une autre façon d'associer une longueur à chaque élément est de considérer sa distance moyenne aux éléments de sa classe. Si x_i est dans la classe X_k,

$$L_i = \frac{1}{|X_k|-1} \sum x \in X_k \, D(x_i,x).$$

La fonction d'inertie, notée I_m, est toujours la somme des carrés de ces longueurs. On notera qu'elle diffère de la somme des carrés des distances intra-classe, qui équivaut à la variance dans le cas d'un nuage euclidien, puisque cette inertie fait intervenir le carré d'une moyenne.

Pour ces trois fonctions d'inertie, la stratégie de calcul des partitions est la même que pour les partitions de séparation maximum. Dans une première phase, on applique une procédure d'allocation recentrage qui, par une suite de descentes, permet de construire une partition initiale. Puis on applique la méthode tabou, avec affectations aléatoires, dans laquelle on déplace un seul élément à chaque essai, pour aboutir à la meilleure partition trouvée. Le nombre maximum d'itérations est également fixé à *N*(*p*-1) et le compteur est remis à zéro après chaque amélioration du critère.

Pour l'inertie par rapport à un centre virtuel, les calculs, s'ils étaient répétés pour chaque essai de la procédure tabou, seraient assez longs si bien que le choix de l'élément déplacé n'est pas fait suite à l'évaluation exacte du critère. On déplace

l'élément dont la distance moyenne à une autre classe est minimum. Mais à chaque itération, on calcule la valeur exacte du critère.

4. EVALUATION DES CLASSES ET COMPARAISON DES PARTITIONS

Finalement, nous avons retenu dix critères d'optimisation pour calculer des partitions à nombre de classes fixé : il y a les critères de séparation, au nombre de six si l'on travaille en norme L_1 et L_2, notées $\sum e$ $\overline{\sum e}$ $\overline{\sum ppd}$ $\sum e^2$ $\overline{\sum e^2}$ $\overline{\sum ppd^2}$, le diamètre Δ et les trois critères d'inertie, *Ir*, *Iv* et *Im*. Chacun permet de construire une partition et, si le nombre d'éléments est assez grand, toutes les partitions optimisées peuvent être différentes. Excepté dans le cas où un critère est particulièrement justifié, comme le diamètre, on ne sait pas très bien laquelle retenir. C'est pourquoi nous nous sommes attachés à évaluer la qualité des différentes classes ainsi que celle des partitions. Cette qualité ne peut que se mesurer par rapport aux données ; il faut donc l'entendre comme une adéquation à la distance proposée. Pour les classes, on privilégie leur homogénéité dans l'ensemble *X*. Pour les partitions, on quantifie la part de grandes (resp. petites) valeurs de distance qui correspondent à des liens inter-classes (resp. intra-classe). Pour ne pas engendrer de confusion entre les critères d'optimisation des partitions et les critères de qualité des classes et de ces mêmes partitions, nous utiliserons pour ces derniers le terme d'*indice*.

4.1. Indices de qualité des classes

Pour chaque classe X_k à n_k éléments, on peut mesurer son homogénéité en calculant les indices suivants :

- Le seuil de connexité s_k. Chaque élément x_i a un plus proche voisin à distance d_i. Ce seuil est la plus grande de ces valeurs d_i pour x_i appartenant à X_k. C'est aussi la longueur de la plus longue arête d'un arbre minimum de la distance réduite à cette classe. Si l'on supprime toutes les arêtes dont la longueur est supérieure ou égale à ce seuil, il y a plusieurs composantes connexes et donc X_k n'existe pas (d'où le nom de cet indice).
- Le taux de liens intra-classe, dont la longueur est inférieure ou égale au seuil s_k. Plus il est faible, plus la classe est le résultat d'un effet de chaîne. Sa plus petite valeur est $2/n_k$ - il n'y a que les arêtes de l'arbre minimum de la classe - et la valeur 1 est atteinte s'il n'y a pas d'arête de valeur supérieure à s_k.
- Le diamètre δ_k, plus grande distance intra-classe. On indique le quotient δ_k/Δ qui doit être le plus faible possible. Tous les liens sont inférieurs ou égaux au diamètre et X_k est une clique du graphe $G_{\leq \delta_k}$. Plus le diamètre est proche de s_k et plus le pourcentage de liens est élevé, plus la classe est homogène. On remarquera que si la distance réduite à la classe vérifie la condition ultramétrique, le seuil de connexité est égal au diamètre.
- La taille maximale des cliques de $G_{\leq s_k}$ incluses dans X_k. Plus elle est grande, plus la classe est solide. Ce problème étant *NP*-difficile, nous ne déterminons qu'un encadrement du cardinal maximum d'une clique. La borne inférieure est obtenue en construisant une clique gloutonne en partant d'un sommet de degré maximum et en ajoutant, tant qu'il y en a, un sommet de degré maximum adjacent à tous les

précédents. La borne supérieure q est le nombre maximum d'éléments de degré supérieur ou égal à q-1, c'est-à-dire qui ont q-1 liens de longueur inférieure ou égale au seuil de connexité.

- Le taux de triplets bien représentés. On ne considère que les triplets composés de deux éléments de la classe, et d'un élément en dehors. Ce triplet est dit *bien représenté* si et seulement si la distance entre les deux éléments de la classe est la plus petite des trois valeurs.

Une classe est d'autant meilleure qu'elle a :

- un seuil de connexité petit,
- un diamètre faible par rapport à Δ et pas trop grand comparé à s_k,
- un taux de liens élevé, voisin de 50 %,
- une clique qui contient une forte proportion des éléments de la classe et
- un taux de triplets bien représentés supérieur à 50 %.

Quand la classe résulte d'un assemblage de proche en proche, qu'il y a un *effet de chaîne*, le diamètre est fort, le pourcentage de liens faible, il y a beaucoup de cliques ayant peu d'éléments et de nombreux triplets mal représentés ; des éléments proches de ceux qui sont réunis dans cette classe n'y figurent pas.

On peut ainsi, dans une partition, juger des classes et éventuellement éliminer l'une d'entre elles, ou certains éléments inclassables que l'on retrouve souvent comme des singletons.

4.2. Indices de qualité des partitions

Pour pouvoir comparer des partitions construites d'après un même tableau de distance, il faut définir des critères différents de ceux qui sont optimisés. De plus, il faut qu'ils ne dépendent pas des cardinaux des classes, qu'ils ne privilégient ni les partitions en classes équilibrées ni celles qui contiennent beaucoup de singletons. Nous avons défini quatre indices qui répondent à ces conditions. Ils sont basés sur le principe qu'une partition P est d'autant meilleure que les grandes valeurs de distance correspondent aux liens inter-classes et les petites valeurs aux liens intra-classe.

Une première étape consiste donc à définir un seuil s, frontière entre les grandes et les petites valeurs de distance. Quel que soit son nombre de classes, une partition P sur X induit une bipartition des paires d'éléments de X : il y a les paires inter-classes, les arêtes externes, et les paires intra-classe qui correspondent aux arêtes internes. Soient (Le | Li) cette bipartition, Ne et Ni le nombre respectif d'éléments de chaque classe. Ne est le nombre d'arêtes du cocycle, égal à la somme des produits deux à deux des cardinaux des classes.

$$Ne = 1/2 \sum_{k=1,\,..\,p} | X_k | (N - | X_k |) \text{ et}$$

$$Ni = 1/2 \sum_{k=1,\,..\,p} | X_k | (| X_k | -1).$$

On a bien évidemment $Ne + Ni = N(N\text{-}1)/2$, quantité notée par la suite $N2$.

Ces cardinaux induisent une autre bipartition des $N2$ paires d'éléments de X, les Ne plus grandes valeurs de distance et les Ni plus petites, notée (Gd | Pd). Le seuil s est défini comme la valeur telle qu'il y a Ne paires dont la distance est supérieure ou égale à s. Une partition sur X, parfaite suivant le principe énoncé ci-dessus, donne deux bipartitions sur les paires qui sont identiques. Nos deux premiers indices mesurent la similitude de ces bipartitions.

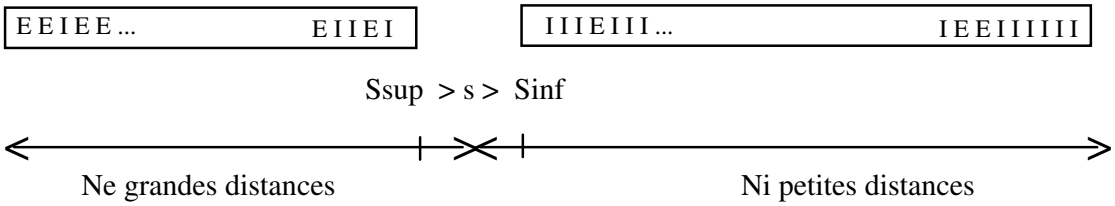

Figure 2 : Deux bipartitions des valeurs de distance rangées suivant l'ordre décroissant : les liens externes (E) et internes (I) et les grandes et les petites distances

4.2.1. Taux de concordance

C'est le pourcentage de paires concordantes, c'est-à-dire des grandes distances qui sont des liens externes ou des petites distances qui sont des liens internes, la distinction entre *Gd* et *Pd* étant faite au seuil de la *Ne*-ème valeur de distance. Comme il peut y avoir des ex-æquo, il faut définir des seuils précis. Rangeons tout d'abord les valeurs de *D* suivant l'ordre décroissant. Soit *s* la *Ne*-ème valeur de distance et *s'* la valeur suivante ($s \geq s'$). Si $s > s'$, on notera $Ssup = s$ et $Sinf = s'$, sinon, *Ssup* est la plus petite valeur strictement supérieure à *s* et *Sinf* la plus grande valeur strictement inférieure à *s*. Entre *Ssup* et *Sinf*, soit il n'y a pas de valeur de distance, soit il n'y a que des valeurs égales à *s*. Le taux de concordance $\tau(c)$ est calculé en comptant les paires qui sont dans l'intersection des deux bipartitions. Si *Clas*(i) désigne le numéro de classe de l'élément x_i,

Inter = 0
Pour toutes les paires (*x,y*)
  Si (*Clas*(x) ≠ *Clas*(y) et *D*(*x,y*) ≥ *Ssup*) *Inter* = *Inter* + 1
  Si (*Clas*(x) = *Clas*(y) et *D*(*x,y*) ≤ *Sinf*) *Inter* = *Inter* + 1
$\tau(c) = 1 - Inter / N2$

4.2.2. Le taux des poids

Le taux des poids $\tau(p)$ est calculé à partir des sommes des distances sur chacune de ces quatre familles, notées respectivement $\sum(Le)$, $\sum(Li)$, $\sum(Gd)$ et $\sum(Pd)$.

$$\tau(p) = \frac{\sum(\text{Le})}{\sum(\text{Gd})} \times \frac{\sum(\text{Pd})}{\sum(\text{Li})}$$

Ces deux quotients représentent respectivement le poids des arêtes externes rapporté au maximum que l'on peut réaliser avec *Ne* valeurs de distance, et le poids minimum pour *Ni* arêtes internes rapporté au poids trouvé. Ils sont, comme leur produit, inférieurs ou égaux à 1. Un taux proche de 1 signifie que les arêtes inter-classes sont parmi les plus grandes, et les arêtes intra-classe parmi les plus petites, donc que la partition est une des meilleures possibles.

Les partitions qui maximisent la séparation ont pour ce critère des valeurs fortes, puisqu'on essaye de mettre dans le cocycle le maximum de grandes distances. Mais il n'en va pas de même pour le taux de concordance, car il se peut que des paires éloignées lient des éléments d'une même classe.

4.2.3. Quotient des longueurs moyennes

Toujours indépendant des cardinaux des classes, il y a un indice très simple pour apprécier la qualité d'une partition ; c'est le quotient de la longueur moyenne des arêtes externes sur la longueur moyenne des arêtes internes.

$$\theta(l) = \frac{\sum(\text{Le}) / \text{Ne}}{\sum(\text{Li}) / \text{Ni}}$$

On peut espérer qu'il est supérieur à 1. Plus il est grand, plus les deux types d'arêtes, externes et internes, sont différents, et la partition fondée.

4.2.4. Taux de triplets bien représentés

Cet indice de qualité des classes peut se généraliser aux partitions. Soit T l'ensemble des triplets constitués de deux éléments dans une même classe, le troisième étant dans une classe différente.

$$T = \{ (i,j,k) \mid Clas(i) = Clas(j) \text{ et } Clas(i) \neq Clas(k) \}.$$

Bien qu'on ne puisse pas toujours le réaliser, on aimerait que la distance entre les deux éléments dans la même classe soit plus petite que leurs distances à l'élément extérieur. Compte tenu du fait que la séparation d'une classe est souvent inférieure à son diamètre, on ne peut s'attendre à ce que tous les triplets vérifient cette condition. Nous comptons le taux de triplets bien représentés en rapportant ce nombre au cardinal de T.

$$\tau(t) = \frac{| (i,j,k)T \mid D(i,j) \leq \text{Min}\{D(i,k),D(j,k)\} |}{| T |}$$

Sur ces quatre indices, le premier et le dernier sont ordinaux ; ils ne dépendent que de la préordonnance, c'est-à-dire du préordre sur les paires induit par les valeurs de distance. On trouvera des indices similaires dans Lerman [1981]. Le second et le troisième dépendent également des valeurs ; ce dernier est tellement naturel que je ne saurais en réclamer la paternité. Ces quatre indices, même s'ils ne sont pas tous concordants, permettent généralement de choisir, parmi plusieurs partitions ayant même nombre de classes, celle qui est la plus conforme à la distance initiale.

5. SIMULATIONS

Les indices de qualité des partitions permettent non seulement d'en choisir une parmi plusieurs concurrentes, mais aussi de déterminer quels sont, en moyenne, les critères d'optimisation les plus efficaces. Sur une distance particulière, ce n'est guère instructif, mais si l'on observe tous ces critères sur un type particulier de distances, et que ce sont plus souvent les uns que les autres qui donnent les meilleures partitions, on pourra choisir quel critère utiliser (sur ce type de distance), sans avoir à les essayer tous.

Il y a plusieurs types de distances que l'on rencontre fréquemment en classification : Nous allons tester les dix critères sur

- les distances euclidiennes, calculées à partir de données quantitatives,
- les distances booléennes, calculées d'après des attributs présents ou absents, et
- les distances d'arbre, qui apparaissent dans les problèmes d'évolution.

Pour décider qu'un critère est plus performant qu'un autre, encore faut-il qu'il y ait des classes dans les données. Sinon, les critères aboutissent à des partitions différentes, mais de qualité très voisines. Pour chacun des types ci-dessus, nous avons généré des distances pour lesquelles il y a une partition à p classes qui est "naturelle" et

que les méthodes devraient retrouver plus ou moins précisément. Pour que chaque problème de partitionnement ne soit pas trop simple, nous générons des classes dont l'intersection n'est pas vide, si bien que les différents critères règlent différemment certaines affectations.

Les simulations sont réalisées pour N=30 avec p classes initiales non équilibrées. On fait donc tourner les programmes jusqu'à avoir une partition à p classes. Les valeurs $p = 3$ et $p = 5$ sont traités ; chaque série de tests porte sur 300 tableaux de distance.

Nous comparons les dix partitions obtenues selon les critères optimisés, à l'aide des quatre indices de qualité des partitions. Ceux-ci sont considérés avec la même importance. Chaque indice permet de classer les critères d'optimisation suivant un ordre croissant, du moins bon au meilleur ; il les place sur une échelle ordinale. Nous attribuons aux critères un nombre de points égal au rang de la partition dans ce préordre, soit un point pour la plus mauvaise partition et 10 points pour la meilleure (en tenant compte des ex-æquo). Pour chaque distance, un critère obtient au moins 4 points si sa partition obtient quatre fois la plus mauvaise note, et au plus 40 points si elle obtient toujours la meilleure. Les chiffres indiqués sont les moyennes sur les 300 distances. Ces chiffres, divisés par 4, donnent le rang moyen de la partition construite par chaque critère.

Exemple : Voici une sortie du programme de test pour une distance sur laquelle on cherche une partition à 5 classes :

```
Cardinaux des classes  9  5  5  5  6

Critère : Somme des écarts inter-classes en norme L1
Partition à 5 classes   0.920  0.782  1.62  0.84
Critère : Moyenne des écarts inter-classes en norme L1
Partition à 5 classes   0.772  0.742  1.30  0.70
Critère : Moyenne des plus petits écarts aux autres classes en norme L1
Partition à 5 classes   0.687  0.685  1.14  0.58
Critère : Somme des écarts inter-classes en norme L2
Partition à 5 classes   0.915  0.785  1.63  0.85
Critère : Moyenne des écarts inter-classes en norme L2
Partition à 5 classes   0.754  0.734  1.18  0.61
Critère : Moyenne des plus petits écarts aux autres classes en norme L2
Partition à 5 classes   0.690  0.665  1.14  0.55

Partitions de diamètre minimum
Partition à 5 classes   0.837  0.634  1.31  0.69

Critère : Inertie par rapport à un centre réel
Partition à 5 classes   0.899  0.744  1.54  0.81
Critère : Inertie par rapport à un centre virtuel
Partition à 5 classes   0.899  0.749  1.53  0.80
Critère : Inertie par la moyenne des distances intra-classe
Partition à 5 classes   0.915  0.798  1.64  0.85
```

Pour le premier indice, le taux de concordance, la plus mauvaise partition est celle qui est donnée par le critère $\overline{\sum \text{ppd}}$ puis vient le même critère en norme L_2, $\overline{\sum \text{ppd}^2}$ qui reçoit 2 points et ainsi de suite jusqu'au critère Σe qui reçoit le maximum des 10 points.

5.1. Distances euclidiennes

Dans un premier temps, nous avons étudié le cas très classique des distances euclidiennes en appliquant le protocole de simulation suivant :

On génère des distances, ayant a priori p classes, dans un espace de dimension p. Chaque centre est placé sur un axe, à distance r de l'origine, et les éléments de cette classe sont tirés au hasard dans une boule de rayon r. Ces boules ont une intersection non vide, comme le montre la Figure 3 pour p=2 ; il se peut qu'un élément de la classe 1 soit plus proche du centre de la classe 2 que de la sienne. Les classes ne sont pas équilibrées ; il y a au moins un élément par classe, et la boule dans laquelle on tire un nouvel élément est sélectionnée au hasard. Le tirage des coordonnées des éléments étant fait, on calcule les distances deux à deux.

Moyenne des points obtenus sur les distances euclidiennes

	$\sum e$	$\overline{\sum e}$	$\overline{\sum ppd}$	$\sum e^2$	$\overline{\sum e}^2$	$\overline{\sum ppd}^2$	Δ	I_r	I_v	I_m
p = 3	24	17	11	26	17	10	20	25	29	32
p = 5	27	14	10	30	13	8	21	25	31	35

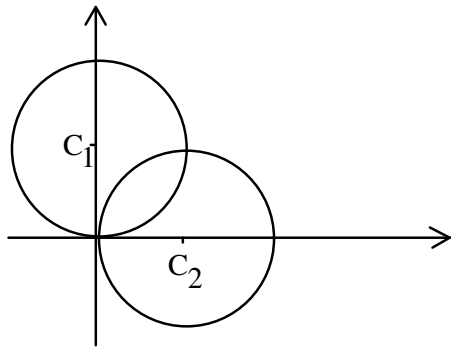

Figure 3 : Enveloppe des classes (p = 2)

Pour les critères de séparation, la supériorité de la somme des carrés des écarts est évidente ; la moyenne des plus petites distances aux autres classes, en norme L_1 comme L_2, donne des résultats médiocres. Ce dernier, ainsi que les critères de moyenne des écarts, ne donnent jamais les meilleurs résultats, quel que soit l'indice. Les critères de diamètre et d'inertie par rapport à un centre réel font jeu égal. Mais ce sont les critères d'inertie par rapport à un centre virtuel et d'inertie par la moyenne qui sont les plus efficaces ; dans les deux cas, il n'y a pas de centre explicite, seulement une estimation de la part d'inertie due à chaque élément. Que celle-ci soit ajustée par un modèle d'arbre en étoile ou par un modèle métrique ne change pas grand-chose.

5.2. Distances booléennes

Pour construire une distance booléenne, il suffit de tirer des vecteurs binaires aléatoires et de mesurer la distance de Hamming entre ces vecteurs. Si l'on veut qu'il y ait p classes, il suffit de consacrer une composante à chacune d'entre elles et donc d'engendrer p attributs qui prendront la valeur 1 si et seulement si l'élément appartient à cette classe. Pour compléter le tableau, on ajoute q attributs dont les valeurs 0 ou 1 sont tirées au hasard. Si bien que deux éléments d'une même classe ont une distance moyenne de q/2 et deux éléments de classes différentes ont une distance de 2 + q/2. Pour N = 30 nous avons pris q=6, valeur qui assure qu'il n'y aura pas trop de vecteurs identiques.

Moyenne des points obtenus sur les distances booléennes

	$\sum e$	$\overline{\sum e}$	$\overline{\sum ppd}$	$\sum e^2$	$\overline{\sum e^2}$	$\overline{\sum ppd^2}$	Δ	I_r	I_v	I_m
p = 3	28	19	10	29	17	9	15	19	30	32
p = 5	28	16	10	32	13	9	21	24	29	35

Le critère de l'inertie par la moyenne reste le meilleur, d'autant plus clairement que le nombre de classes augmente. En dehors de l'inertie de centre virtuel et de la somme des carrés des écarts, tous les autres critères donnent des résultats médiocres. Si l'on pondère les attributs booléens pour calculer les distances, on retrouve le même ordre sur les critères et seul le diamètre améliore ses résultats, sans pour autant que l'on puisse le considérer comme un critère recommandable.

5.3. Distances d'arbre

Pour tirer au hasard une distance d'arbre, on part d'un *X*-arbre aléatoire. Toutes les méthodes ne sont pas équivalentes et certaines donnent des topologies biaisées. Nous avons implémenté la procédure de Yule-Harding [1971] qui consiste à subdiviser aléatoirement toute classe d'au moins deux éléments en deux sous-classes. Elle engendre des arbres étiquetés équiprobables qui sont codés dans une table binaire ; les lignes correspondent aux éléments de *X* et les colonnes aux arêtes internes de l'arbre. Ensuite, on pondère les arêtes de façon aléatoire. Pour faire apparaître des classes, on fixe à 1 le rapport des longueurs moyennes des arêtes internes sur les arêtes externes. Les variations des longueurs suffisent à créer des classes, même si l'on n'en connaît pas vraiment le nombre (Guénoche & Garreta, 2001). Ensuite, on établit, par sommation des longueurs des arêtes le long de tous les chemins entre feuilles, la distance d'arbre correspondante. Les résultats restent identiques si ce rapport est fixé à .5 ; les problèmes sont plus difficiles, mais l'ordre sur les critères reste le même.

Moyenne des points obtenus sur les distances d'arbres

	$\sum e$	$\overline{\sum e}$	$\overline{\sum ppd}$	$\sum e^2$	$\overline{\sum e^2}$	$\overline{\sum ppd^2}$	Δ	I_r	I_v	I_m
p = 3	21	15	11	24	14	10	24	23	26	28
p = 5	23	10	13	27	8	12	27	26	29	29

Ici, les partitions naturelles sont moins évidentes que pour les distances précédentes parce que, dans les arbres construits, tous les nœuds sont de degré 3. Mais il suffit qu'il y ait, dans l'arbre, des arêtes plus longues que les autres pour induire des partitions "vraies". D'ailleurs les meilleures partitions construites, sur 3 ou 5 classes, ont un quotient des moyennes des liens inter-classes sur intra-classe de l'ordre de 1.8 et 80 % des triplets sont bien représentés, comme pour les distances euclidiennes ou booléennes ; ceci indique clairement qu'il y a des classes à trouver.

Il y a une sorte de nivellement des "bons" critères, mais aucun n'atteint le score de 30. Pour les partitions à 3 et 5 classes, nous retrouvons en tête ces mêmes critères d'inertie, de somme des carrés des écarts et de diamètre qui fait ici nettement mieux que pour les autres types de distances.

6. CONCLUSION

L'apport essentiel de cette étude est de proposer des indices de qualité des classes et des partitions. Quelle que soit la méthode de partitionnement, même s'il s'agit d'autres critères ou de méthodes hiérarchiques, voire d'un choix d'expert, on peut toujours comparer deux partitions et décider que l'une est plus en accord avec la distance donnée que l'autre. Pour un ensemble d'éléments dont la structure en classes est connue, c'est aussi une façon de choisir, parmi plusieurs distances, celle qui est le plus en accord avec la partition correspondante.

Maintenant, que ressort-il de cette étude comparative ?

- Le nombre maximum de points est toujours réalisé par le critère de l'inertie par la moyenne qui domine nettement l'inertie réelle et, moins nettement, l'inertie virtuelle par ajustement des longueurs d'un arbre en étoile. Ce résultat, prévisible sur les distances euclidiennes, est plus surprenant sur les distances booléennes. Il est plus mitigé sur les distances d'arbre.
- Les critères de moyenne des écarts et de moyenne des plus petites distances aux autres classes, quelle que soit la norme, sont très inefficaces. Le premier est néanmoins classique, mais son attirance pour les singletons l'entraîne dans de mauvaises partitions. Le second, imaginé pour pallier ce défaut, est très décevant. Nous avons aussi compté le nombre de problèmes pour lesquels ces critères donnent la meilleure partition au sens d'un des quatre indices ; il est quasi nul !
- Pour la somme des écarts, il y a peu de différences entre les normes L_1 et L_2 ; mais elle sont toujours dans le même sens, et la somme des carrés des écarts inter-classes donne nettement plus souvent la meilleure partition que la simple somme, même pour les distances booléennes.

Je serais donc tenté de recommander comme critère d'optimisation pour les méthodes de partitionnement, et ce quel que soit le type de distances :
- la somme des carrés des écarts et surtout
- l'inertie par la moyenne,

car l'approximation par une distance d'arbre est pénalisante en temps de calcul.

BIBLIOGRAPHIE

Arabie P., Hubert L., "An Overview of Combinatorial Data Analysis", in *Clustering and Classification*, P. Arabie, L. Hubert & G. de Soete (Eds.), River Edge N.J., World Scientific Publ., 1996, p. 5-63.

Brucker P., "On the complexity of clustering problems, in *Optimization and Operations Research,* Lecture Notes in Economics and Mathematical Systems n° 157, M. Beckmann, H.Künzi (Eds.), Heidelberg, Springer-Verlag, 1978, p. 45-54.

Fichet B., "Projection on an Acute Symmetrical Closed Convex Cone and its Application to Star Graphs", *Proceedings of Compstat'86*, Heidelberg, Physica-Verlag, 1986, p. 176-181.

Gascuel O., Levy D., "A reduction algorithm for approximating a (non metric) dissimilarity by a tree distance", *J. of Classification*, 13, 1996, p. 129-155.

Glover, F., Laguna, M.,*Tabu Search*, Dordretcht, Kluwer, 1997.

Guénoche A., "Partitions with minimum diameter", *Colloque de l'I.F.C.S,* 1989 Charlottesville.

Guénoche A., Hansen P., Jaumard B. Efficient algorithms for divisive hierarchical clustering with the diameter criterion, *J. of Classification*, 1991, 8, 1, p. 5-30.

Guénoche A., Garreta H., "Can we have confidence in a tree representation ?" Proceedings JOBIM'2000, in *Lecture Notes in Computer Sciences*, 2066, Berlin, Springer-Verlag, 2001, p. 43-53.

Hansen P., Delattre M., "Complete-link cluster Analysis by graph coloring", *Journal of the American Statistical Association*, 73, 1978, p. 397-403.

Hansen P., Jaumard B., "Minimum sum of diameters clustering", *Journal of Classification*, 4, 2, 1987, p. 215-226.

Hansen P., Jaumard B., "Cluster analysis and Mathematical programming", *Mathematical Programming*, 79, 1997, p. 191-215.

Harding E.F., "The probabilities of rooted-tree shapes generated by random bifurcations, *Advances in Applied Probabilities*, 3, 1971, p. 44-77.

Hubert L., "Spanning trees and aspects of clustering", *Br. J. of Math. Statist. Psychol.*, 27, 1974, p. 14-28.

Lerman I.C., *Classification et analyse ordinale des données*, Paris, Dunod, 1981.

Mirkin B., *Mathematical Classification and Clustering*, Dordretcht, Kluwer, 1996.

Monma C., Suri S., "Partitioning points and graphs to minimize the maximum or the sum of diameters", in *Proceedings of the sixth International Conference on the Theory and Applications of Graphs*, 1989, New York.