Mathématiques et sciences humaines
Mathematics and social sciences

197 | Printemps 2012
Catégories, classification, complexité, consensus…
Autour des travaux de Jean-Pierre Barthélemy

Sur le consensus en catégorisation libre

About consensus in free sorting task

Alain Guénoche

Édition électronique
URL : http://journals.openedition.org/msh/12188
DOI : 10.4000/msh.12188
ISSN : 1950-6821

Éditeur
Centre d'analyse et de mathématique sociales de l'EHESS

Édition imprimée
Date de publication : 22 avril 2012
Pagination : 65-82
ISSN : 0987-6936

Référence électronique
Alain Guénoche, « Sur le consensus en catégorisation libre », *Mathématiques et sciences humaines* [En ligne], 197 | Printemps 2012, mis en ligne le 02 mai 2012, consulté le 21 avril 2019. URL : http://journals.openedition.org/msh/12188 ; DOI : 10.4000/msh.12188

Math. & Sci. hum. / Mathematics and Social Sciences (50e année, n° 197, *2012*(1), p. 65–82)

SUR LE CONSENSUS EN CATÉGORISATION LIBRE

Alain GUÉNOCHE[1]

RÉSUMÉ – *À partir de jugements individuels sous forme de catégories (un profil de partitions sur un ensemble X), on cherche à établir des catégories collectives, ici appelées concepts. Nous comparons deux approches combinatoires. La première consiste à calculer une partition consensus, la médiane du profil, c'est-à-dire la partition de X dont la somme des distances aux jugements individuels est minimum ; les concepts sont alors les classes de cette partition consensus. La seconde commence par calculer une distance D sur X, basée sur le profil, et à construire un X-arbre associé à D ; les concepts sont alors certains sous-arbres de cet X-arbre. Nous cherchons à comparer ces deux approches, à mesurer leur congruence, en particulier, dans quelle mesure les classes de la partition consensus, sont des sous-arbres du X-arbre et réciproquement.*

MOTS CLÉS – Consensus, Données catégorielles, Partitions, Représentation arborée

SUMMARY – About consensus in free sorting task

Starting from individual judgments given as categories (i.e., a profile of partitions on an item set X), we attempt to establish a collective partitioning of the items. For that task, we compare two combinatorial approaches. The first one allows us to calculate a consensus partition, namely the median partition of the profile, which is the partition of X whose sum of distances to the individual partitions is minimal. Then, the collective classes are the classes of this partition. The second one consists in calculating, first, a distance D on X based on the profile and then in building an X-tree associated to D. The collective classes are then some of its subtrees. We compare these two approaches and more specifically study the extent to which they produce the same decision as a set of collective classes.

KEYWORDS – Categorization data, Consensus, Partitions, Tree representation

1. INTRODUCTION

Dans ce texte, nous cherchons à comparer deux approches combinatoires du traitement de données catégorielles. Celles-ci correspondent à des sujets, aussi appelés experts, qui répartissent des items – photos, sons, produits – selon des *catégories* individuelles. On admettra qu'un item n'est classé qu'une seule fois (par un expert) et donc que chaque sujet exprime son jugement sous la forme d'une partition dont le nombre de classes est libre, d'où l'appellation de *catégorisation libre*. Les données sont donc un profil Π de partitions sur un même ensemble X. C'est une situation que l'on retrouve aussi :

[1]Institut de Mathématiques de Luminy, 163 avenue de Luminy, 13009 Marseille, guenoche@iml.univ-mrs.fr

- quand les items sont décrits par des variables nominales, avec les données binaires comme cas particulier, chaque variable étant une partition, à deux classes dans ce cas ;
- quand on applique une méthode de partitionement sur un ensemble X décrit avec des données re-chantillonnes (*bootstrapped data*).

On cherche alors à classer les éléments de X, c'est-à-dire passer d'un *profil* de partitions, calculées ou correspondant aux variables ou aux catégories individuelles, à une partition unique, dont les classes collectives sont ici appelées *concepts*.

Une approche classique [Celeux *et al.*, 1989] consiste à considérer chaque partition comme une fonction caractéristique sur les paires et à additionner ces fonctions pour établir une similitude ou une distance sur X, puis à calculer une partition à partir de cette somme. Elle ne résiste pas à des expérimentations rigoureuses. Pour rester dans le cadre de l'Analyse Combinatoire de Données, c'est-à-dire en laissant les items dans leur espace de représentation, les partitions, nous proposons deux façons de traîter le problème.

- La première consiste à construire une partition médiane pour Π, dont la somme des distances aux partitions du profil est minimum. C'est elle qui représente au mieux l'ensemble des catégories individuelles, et qui peut être considérée comme le jugement collectif des experts.
- La seconde est la méthode développée par Barthélemy [1991] et Dubois [1991]. Il s'agit de calculer une distance D entre items, et de représenter cette distance sous forme d'un X-arbre A, c'est-à-dire un arbre dont l'ensemble des feuilles est X et les nœuds sont les racines des sous-arbres correspondant aux classes. La distance tenant compte de toutes les partitions permet le passage de l'individuel au collectif, et les sous-arbres de A sont considérés comme des concepts.

La question est de savoir si ces deux méthodes produisent, sur les mêmes données, des résultats similaires. Plutôt que de comparer les concepts construits sur des données classiques (*benchmark*), nous allons établir un protocole de simulation. Partant d'une partition quelconque, on génère un profil fait de partitions similaires en effectuant un nombre fixé de transferts d'un élément d'une classe dans une autre. Pour chaque profil, on construit alors la partition consensus et la série des scissions du X-arbre correspondant. On calcule alors les valeurs d'indices, dont les valeurs moyennes permettent de mesurer la congruence des deux méthodes.

Dans la Section 2, nous décrivons la façon de calculer des partitions médianes, optimales pour des profils de taille limitée, approchées pour des problèmes de taille quelconque. Dans la Section 3, nous reprenons précisemment la méthode de Barthélemy et Dubois et une façon de déterminer une partition optimale dans un X-arbre. Dans la Section 4, nous décrivons notre processus de simulation qui permet de mesurer la congruence de ces deux méthodes. Il permet d'affirmer la supériorité de la méthode du consensus médian pour construire des concepts à partir de catégories individuelles. Tout au long de ce texte, nous traitons un petit problème de catégorisa-

tion libre, proposé par P. Gaillard, et qui porte sur 16 (extraits de) morceaux de musique, classés par 17 experts musiciens [Stervinou, 2011].

EXEMPLE

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Amélie	2	3	4	7	2	4	1	1	5	3	6	6	2	1	5	8
Arthur	1	4	4	2	1	1	4	4	3	4	2	5	1	5	3	4
Aurore	1	1	2	3	3	2	1	3	4	1	3	4	2	4	3	2
Charlotte	3	6	5	1	3	5	6	2	3	6	1	3	5	4	2	5
Clément	7	3	5	8	4	5	1	1	4	3	9	6	7	6	2	2
Clémentine	1	1	2	3	1	2	1	5	4	1	3	2	2	5	4	2
Florian	2	5	6	8	1	6	7	7	5	3	4	3	2	4	7	7
Jean-Philippe	2	3	3	1	2	3	4	4	2	3	1	1	2	1	4	4
Jérémie	1	2	3	4	1	3	2	5	5	2	5	6	3	6	5	3
Julie	4	4	3	4	4	3	1	1	2	4	2	2	3	2	1	3
Katrin	1	2	2	2	1	3	3	3	1	2	3	4	2	3	2	3
Lauriane	2	1	1	3	2	1	4	4	3	2	1	3	2	4	4	1
Louis	3	1	3	3	3	1	3	2	2	1	2	3	1	2	2	1
Lucie	4	2	3	4	4	1	5	6	6	2	6	6	1	5	6	3
Madeleine	3	2	1	5	3	1	2	2	4	4	5	4	1	2	2	3
Paul	1	4	4	1	1	4	3	3	1	4	3	2	1	3	3	3
Vincent	5	2	2	1	1	2	3	3	4	2	3	4	5	3	4	3

TABLE 1 : Les catégorisations des 17 experts sous forme de partitions.
Chaque ligne correspond à un juge et indique le numéro de classe des 16 morceaux.
Pour Amélie, il y a 8 classes, {7, 8, 14} est la première, {1, 5, 13} la seconde, etc.

2. PARTITION CONSENSUS

C'est le problème de partitionnement d'items décrits par des variables nominales qui est à l'origine des travaux sur le consensus de partitions avec l'article de Régnier [1965] qui introduisit la notion de *partition centrale* ou médiane, celle dont la somme des distances aux partitions du profil est minimum.

2.1. CONSENSUS PAR LA MÉDIANE

Soit X un ensemble à n éléments, $\mathcal{P}$ l'ensemble de toutes les partitions de X et Π un *profil* de m partitions de X pas nécessairement différentes. Pour une partition donnée $P \in \mathcal{P}$ (classes disjointes, non vides dont l'union est égale à X), tout élément $x_i \in X$ appartient à la classe notée $P(i)$. Dans ce qui suit, δ désigne le symbole de Kronecker habituel ; on a donc $\delta_{P(i)P(j)} = 1$ si x_i et x_j sont réunis dans P, $\delta_{P(i)P(j)} = 0$ sinon.

Étant donné un ensemble X fini et un profil Π, le problème de la *partition consensus* consiste à déterminer une partition $\pi \in \mathcal{P}$ qui résume au mieux le profil au sens d'un certain critère. Étant données deux partitions P et Q de X, nous établissons une mesure de similitude S entre P et Q dont la valeur est d'autant plus élevée qu'un grand nombre de paires d'éléments réunis (resp. séparés) dans P le sont également dans Q. Les partitions sur X étant des relations d'équivalence sur les

paires (d'éléments de X), on mesure l'écart entre deux partitions par la distance de la différence symétrique entre ces relations [Barthélemy & Monjardet, 1981], notée Δ, et la similitude est définie par $S(P,Q) = \frac{n(n-1)}{2} - |\Delta(P,Q)|$ ou encore :

$$S(P,Q) = \sum_{i<j} \Big(\delta_{P(i)P(j)}\delta_{Q(i)Q(j)} + (1-\delta_{P(i)P(j)})(1-\delta_{Q(i)Q(j)}) \Big). \tag{1}$$

Cette similitude entre partitions est équivalente à l'indice de Rand [1971] non normalisé, puisque celui-ci est égal au pourcentage de paires conjointement réunies ou séparées.

Le score d'une partition P relativement à un profil $\Pi = (P_1, \ldots, P_m)$ est défini par la somme des similitudes de P relativement à chacune des partitions du profil :

$$S_\Pi(P) = \sum_{k=1}^{m} S(P, P_k). \tag{2}$$

La partition qui maximise S_Π est médiane pour le profil Π, puisque la similitude entre partitions est le complémentaire de la somme des cardinaux des différences symétriques.

Étant donné un profil $\Pi = (P_1, \ldots, P_m)$, on note T_{ij} le nombre de partitions dans lesquelles deux éléments donnés x_i et x_j sont réunis. Dans ces conditions, le score d'une partition P relativement au profil Π peut s'écrire :

$$\begin{aligned} S_\Pi(P) &= \sum_{i<j} \Big(\delta_{P(i)P(j)} T_{ij} + (1-\delta_{P(i)P(j)})(m - T_{ij}) \Big) \\ &= 2\sum_{i<j} \delta_{P(i)P(j)} T_{ij} - \sum_{i<j} \delta_{P(i)P(j)} m + \sum_{i<j} m - \sum_{i<j} T_{ij} \end{aligned}$$

Les quantités $\sum_{i<j} m$ et $\sum_{i<j} T_{ij}$ ne dépendent pas de P. Ainsi, en éliminant ces deux termes et en multipliant par un facteur $1/2$, maximiser $S_\Pi(P)$ est équivalent à maximiser la quantité :

$$\sum_{i<j} \delta_{P(i)P(j)} T_{ij} - \frac{1}{2} \sum_{i<j} \delta_{P(i)P(j)} m.$$

Si l'on note $R(P)$ l'ensemble des paires réunies dans P, c'est-à-dire telles que $\delta_{P(i)P(j)} = 1$, on obtient un critère équivalent à $S_\Pi(P)$:

$$W_\Pi(P) = \sum_{(i<j)\in R(P)} \Big(T_{ij} - \frac{m}{2} \Big). \tag{3}$$

Le critère W_Π s'interprète de façon très intuitive. Il signifie que, dans une partition P, une paire d'éléments de X a une contribution positive au critère (resp. négative) quand ces deux éléments sont réunis dans plus (resp. moins) de la moitié des partitions de Π.

Soit $\mathbf{K}_n$ le graphe complet sur X dont les arêtes sont pondérées par W : $w(i,j) = T_{ij} - m/2$. Soit P une partition en p classes $P = (X_1, \ldots, X_p)$. La quantité $W(X_k) = \sum_{(x_i \neq x_j) \in (X_k)^2} w(i,j)$ est le poids de la clique correspondant à X_k. On a alors

$$W_\Pi(P) = \sum_{k=1}^{p} W(X_k) = \sum_{k=1}^{p} \sum_{(x_i \neq x_j) \in (X_k)^2} \Big(T_{ij} - \frac{m}{2} \Big). \tag{4}$$

Nous nous attachons à construire une partition qui maximise W_Π en présentant une méthode optimale et un algorithme efficace pour établir une partition sous-optimale. Il s'agit de l'heuristique de Fusion-Transfert, FT, qui combine classification ascendante hiérarchique et procédure de transferts, dont le principe était déjà proposé par Régnier [1965].

EXEMPLE

La Table 2 indique le score de chaque paire : $2w(i,j) = 2T_{ij} - m$. Les morceaux 1 et 2 étant classés ensemble dans 3 partitions (Aurore, Clémentine et Julie) leur score est bien de $6 - 17 = -11$. On constate qu'il y a peu de valeurs positives marquées en gras.

2	-11														
3	-15	-5													
4	-9	-13	-13												
5	**9**	-13	-15	-5											
6	-15	-7	**9**	-17	-15										
7	-11	-5	-13	-15	-13	-15									
8	-17	-13	-15	-15	-15	-15	**5**								
9	-9	-15	-17	-13	-7	-17	-17	-11							
10	-9	**11**	-7	-13	-11	-9	-7	-15	-15						
11	-17	-15	-15	-5	-15	-13	-11	-3	-9	-17					
12	-13	-17	-13	-11	-13	-15	-15	-15	-3	-13	-9				
13	-1	-13	-3	-13	-7	**1**	-17	-17	-13	-11	-17	-15			
14	-17	-15	-17	-15	-17	-15	-3	-1	-11	-17	-3	-5	-17		
15	-17	-13	-15	-13	-15	-17	-5	**5**	-3	-15	-7	-13	-15	-9	
16	-15	-11	-1	-17	-15	-1	-5	-5	-17	-13	-9	-15	-5	-11	-9
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

TABLE 2 : Scores des paires de morceaux de musique d'après le profil de la Table 1

2.2. MÉTHODES D'OPTIMISATION

Maximiser W_Π revient à construire une partition de poids maximum ou encore un ensemble de cliques disjointes dans $(\mathbf{K}_n, W)$, qui soit de poids maximum. Suite aux contributions de plusieurs auteurs, Křivánek et Morávek [1986], Barthélemy et Leclerc [1995] et Hudry [2012], le problème est classé NP-difficile, et donc on ne connaît pas d'algorithmes polynomiaux qui donnent une solution optimale.

2.2.1. Méthode exacte

Comme il est déjà signalé dans Régnier [1965], le problème de la partition consensus (ou centrale) est un problème d'optimisation discrète que l'on peut résoudre par

programmation linéaire en nombres entiers. Étant donnée une partition P, en posant $\alpha_{ij} = \delta_{P(i)P(j)}$, le critère W_Π s'écrit

$$W_\Pi(P) = \sum_{i<j} \alpha_{ij} w(i,j) \tag{5}$$

avec la contrainte que α est une relation d'équivalence sur X. Le problème d'optimisation revient donc à trouver la matrice α maximisant W_Π sous les contraintes :

$$\begin{cases} \forall (i<j), \alpha_{ij} \in \{0,1\} \\ \forall (i<j<k), \alpha_{ij} + \alpha_{jk} - \alpha_{ik} \leq 1 \\ \forall (i<j<k), \alpha_{ij} - \alpha_{jk} + \alpha_{ik} \leq 1 \\ \forall (i<j<k), -\alpha_{ij} + \alpha_{jk} + \alpha_{ik} \leq 1 \end{cases}$$

Il s'agit d'un problème de programmation linéaire en 0-1 à $\binom{n}{2}$ inconnues et $3\binom{n}{3}$ contraintes qui peut être résolu de façon exacte pour établir une partition π, réalisant l'optimum de la fonction W_Π sur $\mathcal{P}$. Pour $n = 100$, cela fait 4950 variables et 485100 contraintes. Ce sont les tailles limites des problèmes traitables par le logiciel libre GLPK (GNU Linear Programming Kit). Pour les profils de partitions à deux classes, le temps de calcul peut devenir prohibitif dès $n = 30$.

2.2.2. Méthode approchée

De nombreuses méthodes approchées ont été envisagées, à commencer par la *Méthode des transferts* proposée par Régnier [1965]. Elle consiste, en partant d'une partition quelconque, à affecter un élément à une autre classe tant que le critère à optimiser croît. C'est une simple méthode de descente qui établit un maximum local de la fonction de score. Dans ce qui suit, nous exposons une heuristique d'optimisation de W_Π qui donne d'excellents résultats. Elle est dérivée de la méthode du lien moyen et de la méthode des transferts, suivie d'une procédure d'optimisation stochastique ; d'autres procédures de ce type ont été testées par De Amorim *et al.* [1992].

La première partie, Fusion, correspond à une méthode ascendante hiérarchique. On part de la partition atomique P_0 et, à chaque étape, on réunit les deux classes qui maximisent la valeur de la partition résultante ; ce sont les deux classes pour lesquelles la somme des poids des arêtes interclasses est maximum. Le processus s'arrête quand aucune fusion ne permet plus d'accroître le critère. On aboutit à la partition $\pi = (X_1, \ldots, X_p)$ telle que toute partition π_{ij} obtenue par réunion des classes X_i et X_j est de score plus faible : $W_\Pi(\pi_{ij}) < W_\Pi(\pi)$.

Dans la seconde partie, on commence par calculer le poids de l'affectation de chaque élément x_i à chaque classe X_k. Soit $K(i,k) = \sum_{x_j \in X_k} w(i,j)$. Si $x_i \in X_k$, $K(i,k)$ est la contribution de x_i à sa classe, et au score de la partition courante. Sinon, cette valeur correspond à une affectation éventuelle à autre classe $X_{k'}$. La différence $K(i,k') - K(i,k)$ est la variation du critère consécutive au transfert de x_i de la classe X_k à la classe $X_{k'}$. Notre procédure consiste à déplacer, à chaque étape, l'élément qui maximise un gain non négatif de score. On l'affecte ainsi, soit à une autre classe à laquelle il contribue positivement (s'il en est), soit à une classe supplémentaire. Dans ce cas, cet élément devenu singleton a une contribution nulle au score, ce qui fait que le critère augmente.

Nous avons implémenté cette stratégie en gérant une table, indexée sur X et sur les classes de la partition courante, qui contient les valeurs de K ; la procédure s'arrête lorsque chaque item a une contribution positive ou nulle à sa classe qui est supérieure ou égale à toute autre.

À cet algorithme déterministe, qui renvoie une partition π, nous avons ajouté une procédure d'optimisation stochastique. Ayant remarqué que la partie Transfert était très efficace, nous avons choisi de l'appliquer à des partitions aléatoires obtenues à partir de la meilleure partition courante, par échanges aléatoires d'éléments entre classes (*swapping*). Il y a deux paramètres à définir, le nombre maximum d'échanges ($SwapMax$), et le nombre maximum d'essais consécutifs sans amélioration de W_Π ($NbEs$).

Algorithme de Fusion-Transfert (FT)

1. Procédure de Fusion
   - Partir de la partition atomique P_0
   - Calculer le gain de la fusion de toute paire de singletons ($w(i, j)$)
   - Tant que le score augmente
     - réunir les deux classes donnant un gain maximum
     - mettre à jour les coûts de la fusion de la nouvelle classe avec les classes restantes
2. Procédure de Transfert
   - Calculer le poids $K(i, k)$ de chaque élément x_i dans chaque classe X_k
   - Tant qu'il existe un élément dont le poids dans sa classe est négatif ou n'est pas maximum
     - Le placer dans la classe où sa contribution est positive et maximum, si elle existe, sinon en faire un singleton
     - mettre à jour les poids des éléments dans les deux classes modifiées
3. Procédure stochastique
   - $ess \leftarrow 0$
   - Tant que ($ess < NbEs$)
     - Soit *swap* un entier aléatoire entre 1 et $SwapMax$;
     - A partir de la partition courante π, Faire *swap* échanges d'éléments tirés au hasard;
     - Appliquer la procédure de Transfert qui renvoie la partition π' ;
     - Si ($W_\Pi(\pi') > W_\Pi(\pi)$), { $\pi \leftarrow \pi'; ess \leftarrow 0;$ } sinon $ess \leftarrow ess + 1$.

Complexité : Hormis la procédure stochastique, FT est une méthode de complexité polynomiale : $O(mn^2 + n^3)$.

Le calcul de W est en $O(mn^2)$. Dans la partie Fusion, il y a au plus n itérations pour lesquelles il faut trouver les 2 classes à fusionner ($O(n^2)$) et mettre à jour les

gains ($O(n^2)$). Pour la partie Transfert, il faut d'abord calculer les poids des éléments ($O(mn^2)$) puis, à chaque transfert, déplacer un élément ($O(n)$), dans une autre classe ($O(n)$) et mettre à jour les poids de tous les éléments dans les deux classes concernées ($O(n)$). La procédure de transfert, et la procédure stochastique sont nécessairement finies, puisque les transferts ne sont réalisés qu'en cas d'accroissement de W.

Grâce à un protocole de simulations qui permet de générer des profils pour lesquels on peut calculer une partition consensus (optimale), nous avons montré que la méthode FT donne, même pour des problèmes difficiles, des résultats optimaux dans 80 % des cas et très proches de l'optimal dans tous les autres [Guénoche, 2011]. L'utilité de la procédure stochastique dépend de la difficulté du problème ; elle est essentielle dans le cas d'un profil de bipartitions. Nous avons également comparé FT à d'autres heuristiques, comme celle qui consiste à améliorer la partition du profil de plus haut score par une suite de transferts, ou la méthode de Louvain [Blondel *et al.*, 2008] qui peut s'appliquer à toute matrice de pondérations positives et négatives sur les paires ; la méthode FT reste la plus performante en moyenne.

EXEMPLE

Dans la partition consensus du profil des partitions de la Table 1, il n'y a que de petites classes, dont 7 réduites à un seul élément. Le score de chaque classe est indiqué, ainsi qu'un taux de robustesse, ρ, égal au pourcentage moyen de juges qui réunissent les paires de cette classe. La partition obtenue par l'algorithme de Fusion-Transfert est bien de score optimal 34.

- Classe 1 : (1, 5) ($Score = 9, \rho = 0,765$)
- Classe 2 : (2, 10) ($Score = 11, \rho = 0,824$)
- Classe 3 : (3, 6) ($Score = 9, \rho = 0,765$)
- Classe 4 : (7, 8, 15) ($Score = 5, \rho = 0,549$)
- Singletons : (4|9|11|12|13|14|16)

3. REPRÉSENTATION ARBORÉE D'UN PROFIL DE PARTITIONS

Au début des années 90, J.-P. Barthélemy [1991] et D. Dubois [1991] ont eu l'idée, pour déterminer les catégories collectives correspondant à un profil de partitions, de mesurer une distance entre items et de représenter cette distance sous forme d'un arbre à distances additives ou X-arbre. Rappelons qu'un X-arbre est un arbre dont les feuilles (sommets externes) sont étiquetés par les éléments de X, dont les nœuds (sommets internes) sont non étiquetés et de degré au moins 3, et dont les arêtes ont une longueur positive ou nulle [Barthélemy et Guénoche, 1988]. À chaque X-arbre A est associée une distance d'arbre D_A, telle que $D_A(x, y)$ soit la longueur de la chaîne dans l'arbre entre les feuilles x et y ; c'est la somme des longueurs des arêtes sur cette unique chaîne. Les distances d'arbre sont les distances exactement représentables par un X-arbre, qui est unique. Donc, étant donné un tableau de distance D entre items, on cherche à construire un X-arbre A, dont la distance d'arbre D_A soit la plus proche possible de D ; c'est un problème d'approximation.

Le choix d'une métrique sur X permettait donc de passer des jugements individuels aux catégories collectives, par l'intermédiaire des sous-arbres. Un item est connecté à un ensemble d'éléments formant un sous-arbre, non pas parce qu'il est plus proche, comme dans un arbre hiérarchique, mais parce qu'il s'associe aux autres éléments de ce sous-arbre par opposition aux paires situées hors de ce sous-arbre. C'est la notion de *score* développée par Sattah et Tversky [1977] qui fait qu'une paire $\{x, y\}$ s'oppose (dans l'arbre) à une autre paire $\{z, t\}$ parce que

$$D(x, y) + D(z, t) \leq \min\{D(x, z) + D(y, t), D(x, t) + D(y, z)\}. \tag{6}$$

C'est bien sûr une notion différente de celle des scores des paires du consensus, c'est pourquoi nous utiliserons le terme de *poids*. Le poids d'une paire $\{x, y\}$ est donc le nombre de paires $\{z, t\}$ qui vérifient l'équation 6. L'algorithme ADDTREE de Sattah et Tversky réunit à chaque itération les paires de poids maximum, et construit ainsi un X-arbre associé à une distance D.

3.1. DISTANCES ENTRE ITEMS D'APRÈS UN PROFIL

La question de la métrique sur X d'après un profil de partitions fut vite résolue. Les partitions étant, fondamentalement, des relations sur les paires d'éléments de X qui sont soit réunis soit séparés, la distance *naturelle* entre x et y est la proportion de partitions dans lesquelles x et y sont séparés, soit la *distance de la séparation*[2].

$$D_s(x_i, x_j) = \mid \{P \in \Pi \text{ avec } P(i) \neq P(j)\} \mid = m - T_{ij}.$$

EXEMPLE

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	14	16	13	4	16	14	17	13	13	17	15	9	17	17	16
2	14	0	11	15	15	12	11	15	16	3	16	17	15	16	15	14
3	16	11	0	15	16	4	15	16	17	12	16	15	10	17	16	9
4	13	15	15	0	11	17	16	16	15	15	11	14	15	16	15	17
5	4	15	16	11	0	16	15	16	12	14	16	15	12	17	16	16
6	16	12	4	17	16	0	16	16	17	13	15	16	8	16	17	9
7	14	11	15	16	15	16	0	6	17	12	14	16	17	10	11	11
8	17	15	16	16	16	16	6	0	14	16	10	16	17	9	6	11
9	13	16	17	15	12	17	17	14	0	16	13	10	15	14	10	17
10	13	3	12	15	14	13	12	16	16	0	17	15	14	17	16	15
11	17	16	16	11	16	15	14	10	13	17	0	13	17	10	12	13
12	15	17	15	14	15	16	16	16	10	15	13	0	16	11	15	16
13	9	15	10	15	12	8	17	17	15	14	17	16	0	17	16	11
14	17	16	17	16	17	16	10	9	14	17	10	11	17	0	13	14
15	17	15	16	15	16	17	11	6	10	16	12	15	16	13	0	13
16	16	14	9	17	16	9	11	11	17	15	13	16	11	14	13	0

TABLE 3 : Distance de la séparation entre les morceaux de musique

[2]On peut pondérer les jugements selon un degré d'expertise des juges, ou du nombre de paires séparées dans chaque partition. Dans ce dernier cas, chacun attribue le même nombre de points, quelle que soit sa partition P, en créditant chaque paire séparée d'une fraction $\frac{1}{\frac{n(n-1)}{2} - |R(P)|}$ de points.

3.2. X-arbres et sous-arbres

Initialement, c'est la méthode ADDTREE qui a été utilisée pour construire l'arbre (cf. [Barthélemy et Guénoche, 1988]). Rappelons qu'il s'agit d'une méthode de groupement (ascendante) dans laquelle à chaque itération :

- on calcule le poids de chaque paire (en énumérant tous les quadruplets);
- on réunit la paire de poids maximum (on peut en réunir plusieurs, si leur poids atteint un maximum théorique de $\frac{(n-2)(n-3)}{2}$) que l'on connecte à un nouveau nœud dans l'arbre;
- on calcule les longueurs de trois arêtes (par des formules détaillées dans l'ouvrage pré-cité) ; deux mènent aux sommets réunis et la troisième est l'arête interne qui connecte le nouveau nœud au reste de l'arbre;
- on réduit la dimension du tableau de distance en remplaçant la paire d'éléments réunis par leur sommet adjacent dans l'arbre.

Par la suite, pour des raisons de complexité (ADDTREE est en $O(n^4)$ à chaque itération), mais aussi pour des raisons de qualité de l'arbre reconstruit, en terme d'approximation de la distance initiale mais aussi d'aptitude à retrouver un arbre connu, la méthode NJ [Saitou et Ney, 1987] a été utilisée. Il s'agit également d'une méthode de groupement, mais la paire réunie est choisie différemment : on retient celle qui introduit l'arête interne de longueur minimum, appliquant un principe de parcimonie cher aux évolutionnistes. Elle donne, comme nous l'avons vérifié sur les simulations du paragraphe 4, des résultats bien meilleurs que ADDTREE (en terme de qualité des sous-arbres).

Contrairement aux arbres hiérarchiques, les X-arbres ne sont pas enracinés et la notion de sous-arbre doit être précisée. De fait, un X-arbre est un ensemble de bipartitions (scissions) ; chacune correspond à une arête de l'arbre. Cette arête oppose les deux classes de part et d'autre. Dans un X-arbre il y a n arêtes, dites externes, adjacentes aux feuilles et $n-3$, arêtes dites internes. Il y a donc $2(n-3)$ classes ou sous-arbres possibles, qui n'ont pas 1 ou $n-1$ éléments.

C'est là que s'arrête la partie formelle de la méthode de Barthélemy-Dubois. Le découpage de l'arbre en catégories collectives revenait à l'utilisateur. La méthode se limitait à la représentation d'un profil de partitions sur X sous la forme d'un X-arbre. Comment s'y prenait l'utilisateur ? Généralement, la lecture des X-arbres est guidée par la longueur des arêtes internes qui mènent aux sous-arbres ; plus une arête est longue, plus le sous-arbre correspondant est considéré comme valide et donc interprété comme une catégorie collective mise en évidence par la distance. Ces longues arêtes désignent des classes *bien séparées* choisies par l'utilisateur au vu de l'arbre. Pour achever la méthode d'un point de vue algorithmique, je m'en suis tenu à cette pratique courante. Mais le nombre de catégories retenues reste à définir. Je me suis fixé sur le nombre de classes de la partition consensus qui ne sont pas des singletons.

EXEMPLE
L'arbre obtenu par la méthode NJ appliquée à la distance de la séparation est représenté dans la Figure 1.

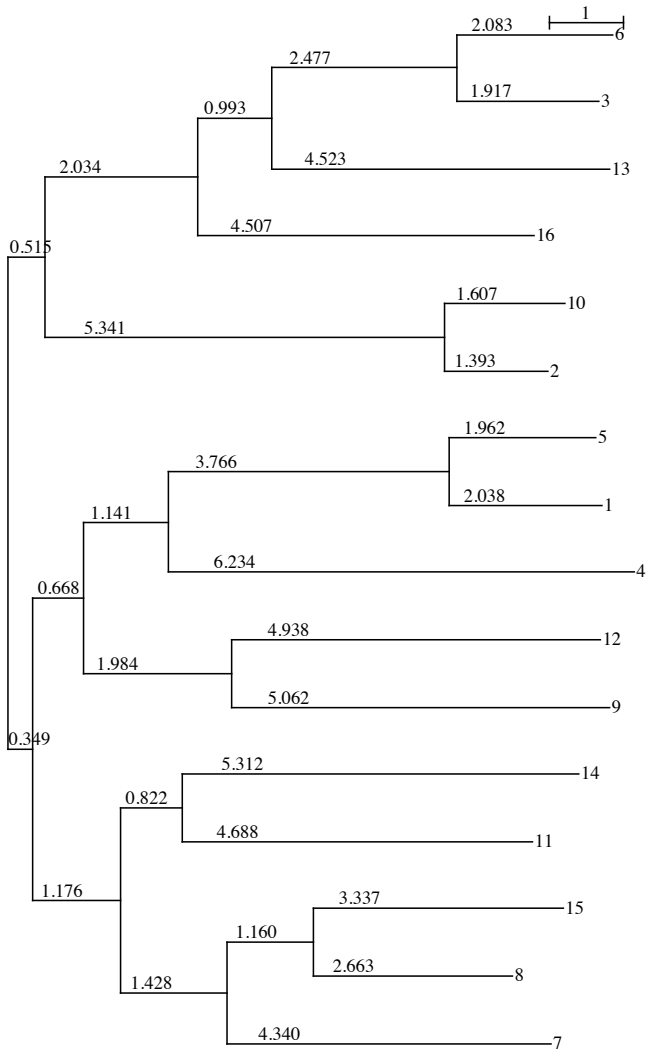

FIGURE 1. Le X-arbre des 16 morceaux de musique avec les longueurs d'arêtes

Notons tout d'abord que les classes $1, 5|2, 10|3, 6|7, 8, 15$ sont bien des sous-arbres. Comme il y a 4 classes d'au moins 2 éléments dans la partition consensus, nous recherchons les 4 sous-arbres les mieux séparés :

- Long = 5,341 → Classe 1 : {2, 10}
- Long = 3,766 → Classe 2 : {1, 5}
- Long = 2,477 → Classe 3 : {3, 6}

- Long = 2,034 → Classe {3, 6, 13, 16} → éliminée puisqu'elle contient 3 et 6 déjà sélectionnés,
- Long = 1,984 → Classe 4 : {9, 12} {de score −3 !}
  vient après
- Long = 1,428 → Classe 5 : {7, 8, 15} ,,, qui n'est pas parmi les 4 sous-arbres les mieux séparés !

4. CONGRUENCE DES MÉTHODES

Ces deux méthodes sont-elles congruentes, c'est-à-dire, donnent-elles des résultats compatibles ? C'est pour répondre à cette question que nous avons établi un protocole de simulation et défini quelques critères pour mesurer cette cohérence.

4.1. DES PROFILS ALÉATOIRES AU CONSENSUS PLUS OU MOINS MARQUÉ

On part d'une partition de X à n éléments en p classes équilibrées, c'est la partition initiale du profil. Ensuite, on génère $m-1$ partitions en appliquant à la partition initiale t transferts aléatoires, dans lesquels un élément pris au hasard est affecté à une classe de la partition en cours, ou à une nouvelle classe. Pour le premier transfert, on tire une classe au hasard entre 1 et $p+1$ et, si une nouvelle classe a été ajoutée, entre 1 et $p+2$ pour le second transfert, etc. Ainsi, les partitions obtenues n'ont généralement pas le même nombre de classes.

À valeurs fixées pour n, p et m, suivant la valeur de t on peut obtenir soit des profils homogènes dans lesquels la partition initiale est la partition consensus, soit des profils très dispersés pour lesquels la partition atomique est le plus souvent la partition consensus [Guénoche, 2011]. En faisant varier la taille moyenne des classes et le nombre de transferts, on obtient des problèmes avec une catégorisation forte, autour des classes de la partition initiale, ou faible, avec peu de paires réunies dans une majorité de partitions, pour lesquelles le consensus sera une partition avec beaucoup de classes et donc de score faible.

4.2. DES CRITÈRES

À partir de chaque profil, nous calculons donc la partition consensus (π) et l'arbre A par approximation de la distance de la séparation. Tous les sous-arbres contenus dans A sont obtenus comme partie connexe suite à la suppression d'une arête interne. À partir de ceux-ci, nous construisons deux partitions :

- Pour chaque sous-arbre, on calcule son score, c'est-à-dire la somme des scores des paires de feuilles réunies. Ceci permet de déterminer la partition P_A, dont les classes sont des sous-arbres et qui maximise la somme des scores ; c'est la meilleure partition contenue dans l'arbre, de score $W_\Pi(P_A)$;
- Soit N_c le nombre de classes de π dont l'effectif est supérieur à 1. La partition P_S est constituée des N_c classes les mieux séparées dans A. On retient donc les N_c scissions dont les longueurs d'arêtes sont les plus grandes, et pour chacune sa classe de score maximum ; les autres éléments sont des singletons dans P_S.

C'est ce que ferait un utilisateur à qui on indiquerait le nombre de classes à extraire, et ce que nous avons fait dans l'Exemple. Ces N_c classes permettent de mesurer le score $W_\Pi(P_S)$ des classes bien séparées dans l'arbre.

Dans la Table 4, nous indiquons tout d'abord les trois valeurs $W_\Pi(\pi)$, $W_\Pi(P_A)$ et $W_\Pi(P_S)$ puis trois critères:

- Pour déterminer dans quelle mesure les classes de la partition consensus sont des sous-arbres du X-arbre, nous avons calculé pour chaque classe de π d'au moins 2 éléments, la taille du plus petit sous-arbre qui contient cette classe. Leurs tailles sont souvent très proches, voire égales, donc nous indiquons ci-dessous le pourcentage de classes de la partition consensus qui est identique à un sous-arbre (τ_c).
- À partir des $2(n-3)$ sous-arbres, nous construisons la partition P_A, dont le score n'est jamais supérieur à $W_\Pi(\pi)$ mais souvent égal, et nous indiquons le pourcentage de problèmes pour lesquels ces scores sont égaux, et donc les partitions identiques.
- Enfin, nous comptons le pourcentage de problèmes pour lesquels le score des classes les mieux séparées est identique au score de la partition consensus.

Toutes les valeurs indiquées sont des moyennes sur 100 profils aléatoires de même type.

4.3. RÉSULTATS

Nous nous sommes restreints à des profils qui correspondent à des protocoles de catégorisation libre. C'est-à-dire que le nombre d'experts et d'items à classer n'est pas supérieur à 50, et les partitions initiales, qui engendrent ces profils, sont équilibrées.

- n : nombre d'items classés
- m : nombre d'experts (de partitions)
- p : nombre de classes de la partition initiale du profil
- t : nombre de transferts de la partition initiale pour générer les partitions

$n=m$	p	t	$W_\Pi(\pi)$	$W_\Pi(P_A)$	$W_\Pi(S)$	τ_c	$\pi=P_A$	$\pi=S$
10	3	3	40,2	40,2	39,6	0,98	0,98	0,79
10	2	5	33,9	33,2	28,8	0,83	0,80	0,25
20	3	5	463,4	454,1	462,9	0,99	0,94	0,92
20	5	10	33,0	32,8	−3,4	0,92	0,92	0,01
20	3	15	11,8	11,2	−114,2	0,83	0,79	0,04
50	5	10	4954,7	4954,7	4954,7	1,0	1,0	1,0
50	10	20	233,5	231,7	−10,9	0,92	0,66	0,00
50	5	30	29,8	29,4	−1876,9	0,86	0,84	0,00

TABLE 4. Scores de la partition consensus, de la partition en sous-arbre de score maximum et de la partition en sous-arbres les mieux séparées.

5. CONCLUSIONS

Ma première conclusion est que la recherche d'un consensus par arbre n'était pas une mauvaise idée ; elle était due à Jean-Pierre Barthélemy. Pour les problèmes faciles ou difficiles, les X-arbres construits contiennent les classes de la partition consensus. Plus de 80 % des classes consensus sont des sous-arbres et sinon, ils diffèrent d'un élément ou deux. De plus les meilleures partitions de l'arbre en sous-arbres donnent des scores très proches de l'optimum. Donc la représentation d'un profil par un X-arbre peut permettre de dégager un consensus cohérent.

Ma deuxième conclusion est qu'il n'est pas toujours facile de lire ces arbres. Cette meilleure partition ne correspond pas nécessairement aux arêtes les plus longues, et le score des classes les mieux séparées est sensiblement plus faible que celui de la partition consensus, dès lors que le problème devient difficile.

Cette difficulté vient du choix des classes dans l'arbre. Nous avions tenté d'apprécier la robustesse des arêtes internes d'un X-arbre [Guénoche et Garretta, 2001] en comptant le nombre de quadruplets dont les valeurs de distance supportaient l'arête. C'est une mesure générale pour toute reconstruction d'arbre à partir d'une distance, qui pourrait être utilisée pour sélectionner les sous-arbres. Après ces essais, on peut affirmer, dans le cas de la distance de la séparation, que le score des sous-arbres est un bien meilleur indicateur, puisqu'il permet d'établir P_A.

Ma dernière conclusion est que s'il faut calculer les scores des sous-arbres pour déterminer la meilleure partition extraite de l'arbre, autant le faire suivant l'ordre inverse ; calculer le score des paires et construire la partition consensus par un processus ascendant (Fusion) suivi des procédures de transfert, déterministe et stochastique. La méthode de consensus médian, pour les profils de partitions, me paraît donc plus appropriée que le consensus par arbre, qui donne toutefois des résultats corrects, si l'on sélectionne les sous-arbres autrement que par la longueur des arêtes.

Maintenant, pour le praticien qui a patiemment recueilli les données, il est très frustrant d'arriver à un consensus qui ne contient que quelques maigres classes et beaucoup de singletons. Deux interprétations sont possibles :

- il n'y a pas de catégories majoritaires, mais il y a quand même des classes soutenues (minoritairement) par un certain nombre de juges ;
- il n'y a pas de catégories majoritaires, parce qu'il y a plusieurs opinions divergentes et que, si l'on pouvait sélectionner des sous-groupes d'experts, on verrait apparaître des catégories différentes correspondant à ces groupes.

C'est pour traiter ces deux cas que nous développons les paragraphes ci-dessous.

5.1. UN CONSENSUS MOU

On remarquera que, s'il n'y a aucune paire (strictement) majoritaire, la partition atomique est la partition consensus. C'est peu informatif et ceci suggère qu'il n'y a aucune classe collective au vu du profil. D'où l'idée d'abaisser le seuil de $m/2$ de façon à avoir plus de paires de score positif.

Au lieu de poser $w(i,j) = T_{ij} - m/2$, on peut choisir un seuil σ et poser

$$w(i,j) = T_{ij} - \sigma.$$

Si $\sigma < m/2$, les pondérations seront augmentées et les classes peuvent s'étendre ou de nouvelles classes de poids positif apparaître. On notera que cette opération n'est qu'une translation des pondérations puisque l'ordre des valeurs n'est nullement affecté.

EXEMPLE
À partir du profil des 17 juges de la Table 1, le seuil majoritaire est de 8,5. Si l'on fixe $\sigma = 6$, on obtient une partition de score optimal 84 et les classes suivantes :

- Classe 1 : 1, 5 ($Score = 14, \rho = 0,765$)
- Classe 2 : 2, 10 ($Score = 16, \rho = 0,824$)
- Classe 3 : 3, 6, 13, 16 ($Score = 30, \rho = 0,500$)
- Classe 4 : 7, 8, 14, 15 ($Score = 22, \rho = 0,461$)
- Classe 5 : 9, 12 ($Score = 2, \rho = 0,412$)
- Singletons : 4 | 11

5.2. DES SOUS-GROUPES D'EXPERTS

Pour classer les juges, ce sont les partitions qu'il faut comparer et mettre ensem-ble les partitions voisines, de façon à faire apparaître des sous-groupes plus homogènes. Pour déterminer s'il y a lieu de subdiviser les experts, et quelle subdivision il convient d'adopter, nous définissons la notion de *score généralisé*. Soit Π un profil décomposé en q sous-profils disjoints ($\Pi = \bigcup_{k=1,\ldots,q} \Pi_k$) et C_k la partition consensus associée au sous-profil Π_k. Le score généralisé de Π est la somme pondérée (par la taille des sous-profils) des scores des partitions consensus.

$$W^q = \sum_{k=1}^{q} |\Pi_k| \times W_{\Pi_k}(C_k).$$

Ainsi la partition consensus de Π, notée C_1, définit un score généralisé

$$W^1 = |\Pi| \times W_{\Pi}(C_1)$$

et s'il existe une q décomposition de Π avec $W^q > W^1$, on peut adopter le point de vue que Π contient q groupes d'opinions ayant chacun leur consensus.

Un tel problème d'optimisation n'est pas simple, puisqu'il faut à la fois déterminer q et la q-décomposition optimale pour connaitre le score généralisé. Celui-ci ne peut être évalué qu'après calcul des q partitions consensus, qui est NP-difficile. On essayera donc des partitions de Π selon des métriques classiques dans l'espace des partitions. Généralement, on parle d'*indice de distance*, c'est-à-dire de fonction de

similitude, à valeurs d'autant plus fortes que les partitions sont proches. Deux partitions P et Q sont en accord sur $\{x, y\}$ s'ils sont simultanément réunis ou séparés. Sinon, elles sont en désaccord si x et y sont réunis dans l'une et séparés dans l'autre.

On utilise souvent l'indice de Rand [1971], qui est simplement le pourcentage de paires pour lequel il y a accord. Il est donc compris entre 0 et 1 et $1-Rand(P,Q)$ est la distance de la différence symétrique sur les ensembles $R(P)$ et $R(Q)$ des paires réunies respectivement dans P et dans Q. On utilisera de préférence l'indice de Jaccard ou l'indice de Rand corrigé par Hubert et Arabie [1985] qui ont l'avantage de décroître de façon monotone quand les partitions diffèrent de plus en plus. Conformément aux voisinages entre partitions établis sur la base de transferts, on pourra aussi utiliser la *Distance des transferts* [Denœud et Guénoche, 2006] qui pour deux partitions P et Q est égale au plus petit nombre de transferts pour passer de P à Q.

EXEMPLE

La partition consensus optimale de Π est de score 34, ce qui donne un score généralisé $W^1 = 578$. Peut-on mieux faire ? Certainement pas en considérant que chaque musicien a une opinion personnelle (dont le consensus est elle-même), peu compatible avec celles des autres, car le score généralisé à 17 classes (singletons) donne $W^{17} = 365$ donc nettement moins que le consensus collectif. Mais assurément, en décomposant Π en deux sous-profils. Plusieurs bipartitions des experts permettent d'améliorer ce score généralisé. La meilleure décomposition est de score généralisé $W^2 = 9\times37+8\times50 = 733$. Elle désigne clairement deux groupes équilibrés qui donnent des partitions consensus (au seuil majoritaire) nettement différentes, puisqu'il faut 9 transferts pour passer de l'une à l'autre :

- Groupe 1 (Amélie, Arthur, Clément, Florian, Jean-Philippe, Katrin, Lauriane, Paul, Vincent)
  - Classe 1 : 1, 5, 13 ($Score = 11, \rho = 0,704$)
  - Classe 2 : 2, 3, 6, 10 ($Score = 10, \rho = 0,593$)
  - Classe 3 : 7, 8, 16 ($Score = 15, \rho = 0,778$)
  - Classe 4 : 11, 14 ($Score = 1, \rho = 0,556$)
  - Singletons : 4|9|12|15
- Groupe 2 (Aurore, Charlotte, Clémentine, Jérémie, Julie, Louis, Lucie, Madeleine)
  - Classe 1 : 1, 4, 5 ($Score = 5, \rho = 0,619$)
  - Classe 2 : 2, 7, 10 ($Score = 9, \rho = 0,714$)
  - Classe 3 : 3, 6, 13, 16 ($Score = 28, \rho = 0,833$)
  - Classe 4 : 8, 9, 11, 15 ($Score = 8, \rho = 0,595$)
  - Singletons : 12|14

Ces deux extensions du calcul du consensus d'un ensemble de partitions permettent donc d'approfondir l'étude d'un profil peu homogène qui ne donnerait qu'une partition proche de la partition atomique.

Un logiciel de calcul de cette partition consensus à partir soit d'un ensemble de partitions, soit d'un tableau de variables nominales, peut être téléchargé depuis le site http://bioinformatics.lif.univ-mrs.fr/.

Remerciements. À l'occasion de ce texte, dédié à la mémoire de Jean-Pierre Barthélemy, je voudrais remercier également tous ceux qui m'ont incité à clarifier (dans mon esprit) ce sujet : à commencer par Danièle Dubois avec qui j'ai collaboré depuis la première heure, à Bruno Leclerc qui est un chercheur fondamental sur ce problème de consensus de partitions et à Pascal Gaillard qui, dans la continuité des travaux de Psychologie Cognitive de Danièle Dubois, s'intéresse de façon très concrète aux protocoles de catégorisation libre et à leur analyse.

BIBLIOGRAPHIE

DE AMORIM S.G., BARTHÉLEMY J.-P., RIBEIRO C.C. (1992), "Clustering and clique partitioning: simulated annealing and tabu search approaches", *Journal of Classification* 9, p. 17-42.

BARTHÉLEMY J.-P., MONJARDET B. (1981), "The median procedure in cluster analysis and social choice theory", *Mathematical Social Sciences* 1, p. 235-267.

BARTHÉLEMY J.-P., GUÉNOCHE A. (1988), *Les arbres et les représentations des proximités*, Paris, Masson, et *Trees and Proximity Representations*, London, J. Wiley, 1991.

BARTHÉLEMY J.-P. (1991), « Similitude, arbres et typicalités », D. Dubois (ed.), *Sémantique et cognition – Catégories, prototypes et typicalité*, Paris, Édition du CNRS.

BARTHÉLEMY J.-P., LECLERC B. (1995), "The median procedure for partitions", I.J. Cox, P. Hansen and B. Julesz (eds), *Partitioning data sets, DIMACS Series in Discrete Mathematics and Theoretical Computer Science* 19, Providence (RI), Amer. Math. Soc., p.3-34.

BLONDEL V., GUILAUME J.-L., LAMBIOTTE R., LEFEBVRE E. (2008), "Fast unfolding of communities in large networks", *Journal of Statistical Mechanics : Theory and Experiment* P10008.

BRUCKER F., BARTHÉLEMY J.-P. (2007), *Éléments de classification*, Paris, Hermès.

CELEUX G., DIDAY E., GOVAERT G., LECHEVALIER Y., RALAMBONDRAINY H. (1989), *Classification Automatique des Données*, Paris, Dunod.

DENŒUD L., GUÉNOCHE A. (2006), "Comparison of distance indices between partitions", V. Batagelj *et al.* (eds.), *Proceedings of IFCS'2006, Data Science and Classification*, Springer, p. 21-28.

DUBOIS D. (1991), *Sémantique et cognition – Catégories, prototypes et typicalité*, Paris, Édition du CNRS.

GUÉNOCHE A., GARRETA H. (2001), "Can we have confidence in a tree representation?", O. Gascuel, M.-F. Sagot (eds.), *JOBIM 2000, Lecture Notes on Computer Science 2066*, p. 45-56.

GUÉNOCHE A. (2011), "Consensus of partitions, a constructive approach", *Advances in Data Analysis and Classification* 5(3), p. 215-229.

HUBERT L., ARABIE P. (1985), "Comparing partitions", *Journal of Classification* 2, p.193-218.

HUDRY O. (2012), "NP-hardness of the computation of a median equivalence relation in Classification (Régnier's problem)", *Mathématiques et Sciences humaines* 197, p. 83-97.

KŘIVÁNEK M., MORÀVEK J. (1986), "NP-hard problems in hierarchical-tree clustering", *Acta Informatica* 23, p. 311-323.

RAND W.M. (1971), "Objective criteria for the evaluation of clustering methods", *J. Amer. Statist. Assoc.* 66, p. 846-850.

RÉGNIER S. (1965), « Sur quelques aspects mathématiques des problèmes de classification automatique », *I.C.C. bulletin* 4, p. 175-19. Reprint in *Mathématiques et Sciences humaines* 82, 1983, p. 13-29.

STERVINOU A. (2011), *La perception structurelle et temporelle d'extraits de musiques contemporaines par les adolescents musiciens et non-musiciens*, Toulouse, thèse de l'Université de Toulouse II Le Mirail.

SAITOU N., NEI M. (1987), "The neighbor-joining method: a new method for reconstructing phylogenetic trees", *Mol. Biol. Evol.* 4, p. 406-425.

SATTAH S., TVERSKY A. (1977), "Additive Similarity Trees", *Psychometrica* 42, p. 319-345.