Consensus of partitions : a constructive approach

Alain Guénoche
CNRS, Institut de Mathématiques de Luminy
guenoche@iml.univ-mrs.fr

Abstract

Given a profile $\Pi \subset \mathcal{P}$ of partitions of X, we try to establish a consensus partition containing a maximum number of joined or separated pairs in X that are also joined or separated in the profile. To do so, we define a score function, $S_\Pi : \mathcal{P} \rightarrow \mathbb{N}$ associated to any partition on X and consensus partitions for Π are those maximizing this function. Equivalently, these consensus partitions have the median property for the profile and the symmetrical difference distance. This optimization problem can be solved, in certain limits, by integer linear programming. We define a polynomial heuristic which can be applied to partitions on a large set of elements. When an optimal solution can be computed, we show that the partitions built by this algorithm are very close to the optimum which is reached in practically any case, except for binary partitions.

1 Introduction

There are many situations where data consist in several partitions on the same set of items X. For instance :

- when the items are described by nominal variables, binary tables being a particular case, since each variable is a partition;
- when experts or judges distribute items - pictures, sounds, products - according to individual *categories* (Dubois, 1991), frequently for sensory estimations;
- after applying several concurrent methods for partitioning the same data.

Then, we aim at clustering elements of X, that is to establish, from a *profile* of partitions, computed or corresponding to variables or individual categories, a single partition summarizing the whole profile.

This consensus problem appears also with the *bootstrap clustering* (Kerr et al. 2001) which consists in generating several sets of data from the initial one ; for instance one can randomly weight variables or put some randomness in the edges of a graph. After clustering, to each data set corresponds a partition, whatever the method used to get it, and their consensus is supposed to be more robust than the partition of a single set.

The first problem (nominal variables) is at the origin of partition consensus works wih the article of Régnier (1965) who introduces the notion de *partition centrale*, the median one, minimizing the sum of the distances to partitions in the profile. He is also at the origin of the transfer distance between partitions (see Charon et al., 2006). If we claim a constructive approach, it is because we do not tackle the question following an axiomatic point of view, or looking for a specific element in the lattice of partitions. For such an approach, we refer

to Mirkin (1975), Bartelemy & Monjardet (1981), Leclerc (1984), Monjardet (1990) and Barthelemy & Leclerc (1995).

We follow the same steps as in phylogenetic tree consensus for which we try to build a X-tree having a maximum number of traits observed in the profile ; it is the extended majority rule (Felsenstein, 2003). In the X-tree case, these traits are the bipartitions of X corresponding to internal edges of the profile trees. In the partition case, one try to build a partition having the maximum number of pairs, joined or separated within the profile. We first show, in section 2, that this problem is the same as the central partition one.

We try to get a solution introducing, in section 3, a very efficient algorithm to establish a sub-optimal partition. It is the *Fusion-Transfert* method denoted FT, which combines ascending hierarchical scheme and a transfer procedure, based on an idea proposed by Regnier. In section 4, we develop a simulation process permitting to generate profiles and problem instances more or less difficult to solve. When an optimal partition can be determined, we show that the FT method gives, even for very difficult problems, optimal results in more than 80% of the instances and score values that are very close to the optimum in any case. In section 5, we detail several connected problems which can be treated by the same algorithm with a few modifications.

2 Consensus formalization

Let X be a set of n elements, $\mathcal{P}$ be the set of all the partitions of X and $\Pi \subset \mathcal{P}$ a *profile* of m partitions. For a given partition $P \in \mathcal{P}$ (disconnected non empty classes covering X) any element $x_i \in X$ belongs to a class denoted $P(i)$. In the following, δ is the usual Kronecker symbol ; consequently, $\delta_{P(i)P(j)} = 1$ iff x_i and x_j are joined in P, $\delta_{P(i)P(j)} = 0$ otherwise. Given a set X and a profile Π, the *consensus partition problem* is to determine a partition $\pi \subset \mathcal{P}$ which summarizes at the best the profile according to some criterion.

We first recall that partitions are equivalence relations on elements of X. Consequently, two partitions P and Q are close when the number of common joined or separated pairs of elements is large. So we can evaluate a similarity mesure S between P and Q using the symmetrical difference distance between these relations, denoted Δ, and the similarity is $S(P,Q) = \frac{n(n-1)}{2} - |\Delta(P,Q)|$. It is the same as the Rand index before normalization, since this latter is the percentage of pairs commonly joined or separated.

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

The score of a partition P relatively to a profile $\Pi = (P_1, \ldots, P_m)$ is defined as the sum of the similarity values between P and any partition in the profile :

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

Proposition 1 *Any partition maximizing S_Π is median for profile Π.*

Proof
This similarity measure between partitions is the complementary part of the sum of symmetrical difference distances values. Consequently maximizing one is the same as minimizing the other.

Given a profile $\Pi = (P_1, \ldots, P_m)$, let T_{ij} be the number of partitions in which two elements x_i and x_j are joined. In these conditions, the score of partition P relatively to profile Π can be written :

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

Quantities $\sum_{i<j} m$ and $\sum_{i<j} T_{ij}$ only depend on the profile Π and not on P. Thus, maximizing $S_\Pi(P)$ is equivalent to maximize :

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

Let $R(P)$ be the set of joined pairs in P. We get a criterion equivalent to $S_\Pi(P)$:

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

Criterion S'_Π can be interpretated intuitively : for a partition P, a pair of $R(P)$ has a positive contribution (resp. negative) when both elements are joined in more (resp. less) than half the partitions in Π.

Let $\mathbf{K}_n$ be the complete graph on X, the edges being weighted by W such that $w(i,j) = T_{ij} - m/2$. Let P be a partition into p classes $P = (X_1, \ldots, X_p)$. Quantity $W(X_k) = \sum_{(x_i,x_j) \in X_k} w(i,j)$ is the weight of the clique corresponding to X_k. Then we have

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

3 Optimization problem

To maximize S'_Π is a *clique partitionning problem*, since we seek for a set of disconnected cliques in $(\mathbf{K}_n, W)$, having a maximal weight. It is an extension to weighted graphs of the Zahn problem (1964), which has been proved NP-hard by Krivanek & Moravek (1986). Hence, no polynomial algorithm, giving an optimal solution, is known. But we have some evident properties :

- $\max_{P \in \mathcal{P}} S'_\Pi(P) \geq 0$.
  Let P_0 be the atomic partition of X in which all the classes are singletons (and any pair is separated). Then, $R(P_0) = \emptyset$ and $S'_\Pi(P_0) = 0$. Consequently, whatever is the profile Π, there always exists a partition with nul score. P_0 realizing the maximum of function S'_Π over $\mathcal{P}$ can be interpreted as : partitions in the profile too much desagree to admit a non trivial consensus; in that case, the consensus partition is the atomic partition.
- Let $E = \{(i,j)$ such that $w(i,j) \geq 0\}$; the weighted graph $G_R = (X, E, W)$ will be denoted *Graphe de Régnier* in homage to the author, even if he does not refer to graphs in his article. If $E = \emptyset$, the atomic partition is a consensus partition ; it is not necessarily unique, in case there are edges in E with weight 0. On the other hand, if $E \neq \emptyset$ there exists a non trivial consensus partition, having at least one majority pair, both elements being in at least half the partitions in the profile.

- There exists another situation for which the consensus partition is known ; it is when each connected component of G_R is a clique, for instance a set of disconnected edges. This partition is necessarily optimal since any inter-classes pair has a negative value.

Proposition 2 *Let CC be the connected component partition of G_R. Any consensus partition is finer than CC.*

Proof
All the edges between classes of CC have a negative weight. So, any class in a consensus partition which is not included in a class of CC could be subdivided, increasing the score value.

Example 1 *Let Π be a set of 3 partitions $P_1 = \{123|45|6\}$, $P_2 = \{135|246\}$ and $P_3 = \{15|24|36\}$. One get the following T and W tables :*

T	1	2	3	4	5	6	$2 \times W$	1	2	3	4	5	6
1	-						1	0					
2	1	-					2	-1	0				
3	2	1	-				3	1	-1	0			
4	0	2	0	-			4	-3	1	-3	0		
5	2	0	1	1	-		5	1	-3	-1	-1	0	
6	0	1	1	1	0	-	6	-3	-1	-1	-1	-3	0

The edges of the Graphe de Regnier are $(1,3)$, $(1,5)$, and $(2,4)$. One can easily find an optimal partition, $\pi_1 = \{135|24|6\}$ with score 2, the same as partitions $\pi_2 = \{13|24|5|6\}$ and $\pi_3 = \{15|24|3|6\}$.

3.1 Optimal methods

In this section we study in which limits and how this optimization problem can be solved in an optimal way ?

3.1.1 By integer linear programming

As it is mentionned in Régnier (1965), the consensus partition problem is a discrete one which can be solved by linear programming. Given a partition P, with notations $\alpha_{ij} = \delta_{P(i)P(j)}$, the S'_Π criterion can be formulated :

$$S'_\Pi(\alpha) = \sum_{i<j} \alpha_{ij} w(i,j). \tag{5}$$

The optimization problem is to determine a symmetric matrix α maximizing S'_Π under constraints indicating that P is a equivalence relation on X.

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

It is a NP-hard problem of integer linear programming with $n(n-1)/2$ variables and $3\binom{n}{3}$ constraints. There exist optimal resolution methods to find α, and a partition π, realizing the global maximum of function S'_Π over $\mathcal{P}$.

A detailled Matlab implementation is decribed in the T. Brenac monography (2002) dealing with the dual problem. We used software GLPK (GNU Linear Programming Kit) to

calculate maximal scores when it possible. In fact, the set of constraints $\alpha_{ij} + \alpha_{jk} - \alpha_{ik} \leq 1$ makes a table indexed by constraints and by pairs $(i < j)$ of elements of X. For each triple $(i < j < k)$ there are 3 constraints, one with coefficients $a_{ij} = 1, a_{jk} = 1, a_{ik} = -1$, the second one with $a_{ij} = 1, a_{jk} = -1, a_{ik} = 1$ and the third one with $a_{ij} = -1, a_{jk} = 1, a_{ik} = 1$, the other coefficients being equal to 0. Consequently, there are $\frac{n(n-1)(n-2)}{2}$ linear constraints. For $n = 100$ that makes 4950 binary variables and 485.100 constraints. These are the limits of our simulations, making instances that are not always computable, particularly for binary tables.

3.1.2 By enumeration

As a consequence of Proposition 2, the connected components of the *Graphe de Régnier* can be treated separately. For each class C_j, we look for an optimal decomposition (not necessarily in two subclasses) enumerating partitions of C_j. For that, we apply the NEXEQU algorithm of Nijenhuis & Wilf (1978). It generates all the equivalence relations over C_j starting from the partition in one single class and passing just one time through all the others. This sequential algorithm goes from one partition to the next in $O(n)$. It can be used for any class with no more than 12 elements ; over, the computing time is too high. We remark that it is not n which bounds the computation, but the maximal size of a connected component of G_R. The optimality of partitions of more than 1000 elements have been proved !

This enumeration is not always necessary. If in class C_i there is only one pair (x, y) having negative weight $(w(x, y) < 0)$, to increase the score, x and y must be separated. If the two sums of weights, $\sum_{z \in C_i} w(x, z)$ and $\sum_{z \in C_i} w(y, z)$ are larger than or equal to $-w(x, y)$, it is useless to separate x and y. Obviously, the assignment of any z, on the x side or on the y side would make a lower weighted subdivision.

More, when there is only one pair of negative weight, it is not necessary to enumerate the whole set of partitions over C_i, but only bipartitions of $C_i \setminus \{x, y\}$ which is much more efficient and can be performed until 20 elements using another algorithm in the Nijenhuis & Wilf book.

3.2 The Fusion-Transfert (FT) method

Numerous heuristics have been proposed, beginning with the *transfer method* proposed by Régnier. Starting from any partition, it consists in assigning one element to any other class as long as criterion increases. It is a simple "hill climbing" method which stops on a local maximum of the score function. Its value depends on the initial partition which is not indicated in the original article. One can suppose that P_0 or the profile partitions have been used in the first implementations (on IBM 704 and UNIVAC 1107) which allow, according to the author, to treat instances until $n \times m = 10000$).

Function w can be interpreted as *similarity* on X, so one can apply any hierarchical clustering or partitioning method of G_R (Guénoche, 2008). In the following, we detail a heuristic, the Fusion-Transfer method (FT), to optimize S'_Π which is very efficient. It is derived from the Newman method (2004), which is itself inspired from Ward (1956), and also from the transfer method.

The first part, Fusion, corresponds to a hierarchical ascending method. Starting from the atomic partition P_0, at each step the two classes maximizing the score value of the resulting partition are joined ; that are the two classes for which the sum of weights of the inter-classes edges is maximum. The process stops when there is no fusion increasing the score. It leads

to partition $\pi = (X_1, \ldots, X_p)$ such that any partition π_{ij} corresponding to fusion of classes X_i and X_j has a lower score : $S'_\Pi(\pi_{ij}) < S'_\Pi(\pi)$.

In the second part, the weight of the assignment of any element to any class is computed. Let $K(i,k) = \sum_{x_j \in X_k} w(i,j)$. If $x_i \in X_k$, $K(i,k)$ is the contribution of x_i to its own class, and also to $S'_\Pi(\pi)$. Otherwise, this value corresponds to a possible assignment to another class $X_{k'}$. The difference $K(i,k') - K(i,k)$ is the score variation consecutive to the transfer of x_i from class X_k to class $X_{k'}$. Our procedure consists in moving at each step the element maximizing this gap. Element x_i is assigned either to class $C_{k'}$ if $K(i,k') \geq 0$ otherwise to a new class making a singleton. In this latter case, this element has a nul contribution to the score, increasing the criterion value. We have implemented this algorithm, making a table, indexed on X and on the classes of the running partition containing the values of K ; It stops when each item has a non negative contribution to its class which is larger than for any other one.

Algorithme de Fusion-Transfert
Hierarchical procedure

- Start from P_0
- Compute the variation of fusion of any pair ($w(i,j)$)
- While score S' increases
  - Join the two classes giving the maximum variation
  - Update the fusion gains of the new class with the remaining ones

Transfer procedure

- Compute the weight of any element in any class
- Memorize the maximum value for any element
- While there exists an element of which the weight in its class is not maximum,
  - put it into the class where its contribution is maximum, if ≥ 0, otherwise make it a singleton;
  - update the weights of the elements in both modified classes

Proposition 3 *FT is a polynomial method with complexity* $O(mn^2) + O(n^3)$

Proof
Computation of table W is in $O(mn^2)$. In the fusion part, there are at most n iterations in which we first search the 2 classes to join ($O(n^2)$) before to update the gains ($O(n^2)$). The hierarchical procedure is in $O(n^3)$ as most hierarchical clustering algorithms. In the transfer part, we compute first the weights of each element in each class and its maximum value in $O(n^2)$. To test if an element is in the best class is in $O(1)$. For each transfert ($O(n)$), to put an element in another class and to update the weights are in $O(n)$. Hence, the transfer procedure is in $O(n^2)$.

4 A simulation protocol

To evaluate this heuristic, simulations have been made. A balanced partition P_1 of X in p classes is the initial one in the profile. Then, $m-1$ partitions are generated, applying P_1 t random transfers, that is one random element either in another class of the running partition

or in a new class. For the first transfer, the new class is selected between 1 and $p+1$ and, if a new class has been added, between 1 and $p+2$ for the second transfer and so on. Doing so, partitions in a profile have not the same number of classes.

To fixed values for n and m, according to t, one can get either homogeneous profiles for which the initial partition, or one in the profile, is the consensus partition, or very inhomogeneous profiles for which the atomic partition makes the consensus. We first try to characterize difficult problems, those for which the optimal solution is hard to find.

For comparison, we have also tested the transfer procedure applied to the CC partition (connected components of the *Graphe de Régnier*). This method is referenced as CT in the following. We fix $n = 50$, $p = 5$ and make the number of transfers used to generate profiles vary. We observe three types of problems, those for which there are few partitions ($m = 10$), a medium number ($m = n = 50$) or a high one ($m = 100$). The pourcentages de problems (over 100 trials) for which FT or CT have found the optimum ($\%_{FT}$ and $\%_{CT}$) are given in Table 1.

n=50	$m = 10$					$m = 50$					$m = 100$				
t	10	20	25	30	40	10	20	25	30	40	10	20	25	30	40
$\%_{FT}$	100	98	90	88	89	100	94	86	99	100	100	95	83	100	100
$\%_{CT}$	100	89	72	63	83	100	88	63	93	100	100	92	73	99	100

Table 1 : Percentages of problems for which heuristics FT and CT have found the optimal partition. Average values over 100 profiles of m partitions on 50 elements generated after t transfers from a initial partition in 5 balanced classes.

Obviously, on one hand FT does better than CT and, on the other hand, for $t = n/2$, (25 transfers), the heuristics are less efficient. We have also verified (results not printed here) that profiles with a constant number of classes provide easier problems, as those for which the initial partition is selected at random in $\mathcal{P}$.

Consequently, we restrict to hard cases, with $t = n/2$, making n and m vary. Over 100 random profiles we evaluate the average values of :

- the score of the best partition in the profile ($S_{Best\Pi}$)
- the score of the optimal partition (S_{Opt}),
- the score of the partition computed by FT (S_{FT}), and
- the score of the partition computed by CT (S_{CT}),

and also

- the percentage of problems for which FT finds the optimum ($\%_{FT}$).
- the percentage of problems for which CT finds the optimum ($\%_{CT}$).

4.1 Difficult problems

4.1.1 Few partitions: $p = n/10$, $t = n/2$ and $m = 10$

n	$S_{Best\Pi}$	S_{Opt}	S_{FT}	$\%_{FT}$	S_{CT}	$\%_{CT}$
20	98.6	137.9	137.8	98	137.4	92
30	68.2	166.1	166.0	94	165.5	82
50	-9.4	227.0	226.8	90	225.9	72
100	-251.1	350.2	349.8	81	348.8	58

4.1.2 A medium number of partitions: $p = n/10$, $t = n/2$ and $m = n$

$n = m$	$S_{Best\Pi}$	S_{Opt}	S_{FT}	$\%_{FT}$	S_{CT}	$\%_{CT}$
20	102.4	171.8	171.7	96	171.4	91
30	-88.2	193.9	193.8	93	192.8	78
50	-1059.3	166.6	166.0	86	164.9	63
100	-7348.6	68.0	67.9	96	67.6	89

4.1.3 Many partitions: $p = n/10$, $t = n/2$ and $m = 100$

n	$S_{Best\Pi}$	S_{Opt}	S_{FT}	$\%_{FT}$	S_{CT}	$\%_{CT}$
20	130.4	296.7	296.5	93	295.9	87
30	-572.7	211.4	210.9	85	209.7	70
50	-2332.6	116.6	116.0	83	115.2	73
100	-7348.6	68.0	67.9	96	67.6	89

Table 2, 3 et 4 : Average results for 100 profiles of m partitions of n elements generated by $t = n/2$ transfers from the balanced partition in $n/10$ classes.

First conclusions :

- in each table, FT gives better results than CT in the average, but it is not true for all instances;
- the score values given by FT are very close to the optimal ones ;
- the optimum is reached by FT in more than 80% of the difficult cases ; for the easy ones ($t < n/2$ ou $t > 2n/3$) it is more than 90% ;
- the best partition in the profile is far from the consensus.

4.2 A stochastic variant

Binary tables ($p = 2$ for all the partitions) generate consensus problems much more difficult. The discrete linear programming method can hardly be used for simulations when n is larger than 30. Some instances are quickly solved, but others can necessitate several hours, because the solution relaxing integrity constraints is far from the integer one. Already, for $n = 20$ the *Graphe de Régnier* is connected, and partitions cannot be enumerated.

Nevertheless, we obtain results indicating that, for $n = 20$ (resp. $n = 30$), FT find the optimum in only 55% (resp. 28%) of the cases, keeping $t = n/2$. It is much less efficient than before, even if the score values remain into a range lower than 5 % of the optimum. Consequently we have developped a *stochastic variant* of the transfer procedure inspired from the one proposed by Lin & Kernighan (1973) for the traveling salesman problem :

Stochastic procedure for transfers

- Apply the transfer procedure to the partition π_0 resulting of the fusion part;
- Do n trials
  - From π_0, do k swaps between 2 random elements taken in 2 distinct classes (k is a random number between 1 and $n/2$) ;
  - Apply the transfer procedure to get partition π
  - If $S'(\pi) > S'(\pi)$, $\pi_0 := \pi$
- Return partition π_0 which has maximum score

The stochastic procedure for transfer is equivalent to apply n times the simple procedure. Since this latter is in $O(n^2)$, the stochastic procedure is in $O(n^3)$, which does not extend the global complexity of the FT method. Now, we get on binary tables the results in Table 5 ($t = n/2$ and $n = m$):

n	$S_{Best\Pi}$	S_{Opt}	S_{FT}	$\%_{FT}$	S_{CT}	$\%_{CT}$
20	151.8	195.9	195.7	92	195.2	85
25	246.5	321.1	320.7	92	320.2	87
30	347.2	478.6	476.8	82	476.7	77

Table 5 : Results for binary tables binaires with stochastic procedure for transfer.

On binary tables the FT method gives slightly better results than CT and remains very close to the optimal values. We also want to precise that if $t < n/3$ or $t > 2n/3$, both methods finds the optimal in 100% of problems. It suggests a test we have realized : Let π be the FT partition and $\theta_\Pi(\pi)$ the maximum transfer distance between π and any $P_k \in \Pi$. When $\theta_\Pi(\pi) \leq n/3$ or $\theta_\Pi(\pi) \geq 2n/3$, the problems are easy to solve, and π is the optimal solution. Then, we come back to the ordinary difficult problems, applying the stochastic procedure for transfer. For all the instance (100 % of the trials), FT as CT have found the optimum !

5 Extensions

In this section, we tackle some linked problems having solutions derived from consensus partitions methods.

5.1 A weak consensus

We have underlined that if there is no majority pairs, the atomic partition is the consensus partition. It is disappointing, poorly informative, suggesting that there is no valid class from this profile. But the majority threshold ($m/2$) can be decreased, making higher values in the complete weighted graph. Doing so, there will be more edges in the *Graphe de Régnier*. Instead of $w(i,j) = T(i,j) - m/2$ a threshold σ can be chosen and we pose

$$w(i,j) = T(i,j) - \sigma.$$

If $\sigma < m/2$, the weights will be increased and classes with positive weight will appear. We remark that this operation is just a translation of the weights since the order value is not modified.

Considering the *Split distance* on X defined by : $D_s : X \times X \longrightarrow \mathbb{N}_+$ in which $D_s(x_i, x_j)$ is the number of partitions separating x_i from x_j. It is a distance, since :

- $D_s(x_i, x_j) = 0$ means that they are joined in all the partitions in the profile, and so they can be identified ;
- $D_s(x_i, x_j) = D_s(x_j, x_i)$;
- $D_s(x_i, x_j) + D_s(x_j, x_k) \geq D_s(x_i, x_k)$, since it is true for all the partitions.

The order on pairs according to D_s is the same as for T and W (the preordonnances are identical). So, making the σ threshold vary, is to add or remove edges in the *Graphe de Régnier* following this order. Connected components at a given threshold are the classes of a hierarchy computed from D_s. It does not mean that the consensus partition is reductible to the choice of a cut level in this hierarchy, since the transfer part comes to mix the classes, optimizing the score criterion.

5.2 A robustness measure for classes and partitions

We have defined the score of a partition $S'_{\Pi}(P)$ as the sum of the weights of the classes. A class score is as large as the pairs are often joined in the profile partitions. One can measure the robustness of a class counting the average percentage of partitions putting its pairs together. This measure does not depend on threshold σ ; majority consensus will have better values than weak consensus :

$$Rob(X_k) = \frac{2.T(X_k)}{m.|X_k|.(|X_k|-1)}.$$

This quantity (between 0 and 1) is the average percentage of partitions joining any pair in the class. The higher it is, the better is the class, since it contains pairs frequently joined in the profile. The robustness of a singleton is not defined by this formula ; there is no problem fixing it to 0.

The same computation can also be done for any partition, indicating its "strength". For a partition $P = \{X_1, \ldots, X_k, \ldots, X_p\}$,

$$St(P) = \frac{\sum_k T(X_k)}{m \sum_k \text{ such that } {}_{|X_k|>1} |X_k|.(|X_k|-1)/2}.$$

It looks like what we did when selecting the best partition in the profile, but we count edges and not percentage of partitions. This other criterion can be used to compare partitions, for instance to select among consensus ones. For Example 1, $\pi_1 = \{135|24|6\}$ have a strength equal to 1/2, but the two others $\pi_2 = \{13|24|5|6\}$ and $\pi_3 = \{15|24|3|6\}$ get 2/3.

5.3 A consensus with a fixed number of classes

Another natural extent of the FT algorithm is to impose the number of classes for the searched partition. This can be justified when all the profile partitions have the same number of classes or when the solution imply this constraint for instance for storage or task assignment. One can remark that the above linear programming formulation does not permit to fix the number of classes. Another formalization must be adopted, with binary variables denoting that x_i belongs or not to class C_k.

For FT, it suffices to join the classes until the required number is reached, even if fusions make the score decreasing and then, to make transfers between existing classes, forbidding to add singletons.

5.4 A consensus for which all the joined pairs are majority

Within the classes of a consensus partition, all the joined pairs do not have a positive weight. One can seek for a consensus having this property, that is with only pairs joined in at least half the partitions. It is easy to built from the *Graphe de Régnier*, since the searched classes are the cliques of this graphe. This set of classes does not make a partition but a covering of X. If a strict partition with disconnected classes is searched, the FT method can be modified : in the fusion parts two classes can be joined if and only if they make a clique and this condition can be tested before transfers.

5.5 An fair influence of partitions

An elementary remark, when comparing partitions established by judges (categorization data proposed by experts), is the following : according to the number and to the cardinality of

the classes, each expert contributes disparately to the weights. A judge making two balanced classes, if n is even ($n = 2q$), gives $|R(P)| = q(q-1)$ score points, but the one who choose classes with only 2 elements gives only q points.

One can demand to the experts to give exactly the same quantity of score whatever are their judgements. This can be done giving to any joined pair a fraction $\frac{1}{|R(P)|}$ of score point, admitting that the atomic partition does not give any point. For this weighting of pairs, it is not any more half the number of partitions $m/2$ that must be subtracted, but half the sum $\mu = \sum_k |R(P_k)|$ and formula (4) becomes :

$$S'_{\Pi}(P) = \sum_{k=1,..p} \sum_{x_i,x_j \in C_k} \Big(\frac{1}{|R(P_k)|} - \frac{\mu}{2} \Big). \tag{6}$$

The optimization problem remains the same, except $w(i,j)$ is not an integer, but the FT algorithm can always be used.

6 Conclusions

In this article, we have introduced, with the *Graphe de Régnier*, another formulation of the consensus partition problem. It appears to be a special case of the Clique Partitioning problem, where edges are weighted positively and negatively. We have defined a score function which permits to decide if there is a non trivial solution, and sometimes to test if a partition is optimal. More, this formulation makes it possible to extend the majority consensus notion, in a stronger sense, imposing to all the intraclass pairs to be joined in at least half the partitions or more, or in a weaker sense, fixing a threshold lower than the majority.

The optimized score function is the sum of class weights, which permits to evaluate the robustness of the classes and consequently their respective quality. It corresponds to the average percentage of partitions in the profile joining the intraclass pairs, quantifying the ability of any partition to summarize the profile.

We have also defined the FT method, which has been tested on medium size problems ($n \leq 100$). When the optimal partition can be computed, we have shown that the FT consensus is identical in practically all the cases, except for binary partitions (in which it is more than 80%). And when they are not identical, the FT score is very close to the optimum. Method FT can also be applied to large instances ($n > 1000$), computation time being around 15 seconds for $n = 1000$. And when the transfer distance between the FT consensus and any partition in the profile is lower than $n/3$ or larger than $2n/3$, one can bet it is an optimal partition.

Acknowledgments

This work is supported by ANR PiriBio. I would like to thank B. Estrellon and K. Nouioua (LIF, Marseille) for their help with GLPK.

References

Barthélemy J.P., Monjardet B. (1981) The Median Procedure in Cluster Analysis and Social Choice Theory, *Math. Soc. Sci.*, 1, 235-268.

Barthélemy J.P., Leclerc, B. (1995) The median procedure for partitions, *DIMACS Series in Discrete Mathematics and Thoretical Computer Science*, 19, 3-34.

Brenac T. (2002) *Contribution des méthodes de partition centrale à la mise en evidence expérimentale de catégories cognitives*, INRETS, Arcueil.

Charon I., Denoeud L., Guénoche A., Hudry O. (2006) Maximum transfer distance between partitions, *Journal of Classification*, 23, 1,103-121.

Dubois, D. (1991) *Sémantique et cognition - Catégories, prototypes et typicalité*, Edition du CNRS.

Felsenstein J. (2003) *Inferring Phylogenies*, Sunderland (MA), Sinauer Associates Inc.

Guénoche A. (2008) Comparison of algorithms in graph partitioning, *RAIRO*, 42, 469-484.

Kerr M., Churchill G.A. (2001) Bootstraping clustering analysis : Assessing the reliability of conclusions from microarray experiments, *PNAS*, 98 (16), 8961-8965.

Krivanek M., Moravek J. (1986) NP-hard problems in hierarchical-tree clustering, *Acta Informatica*, 23, 311-323.

Leclerc B. (1984) Efficient and binary consensus functions on transitively valued relations, *Math. Soc. Sci.*, 8, 45-61.

Lin S., Kernighan B.W. (1973) An effective heuristic algorithm for the travelling salesman problem, *Operations Research*, 21, 498-516.

Mirkin B.G. (1975) On the problem of reconciling partitions, in *Quantitative Sociology*, Academic Press, 441-449.

Monjardet B. (1990) Arrowian characterization of latticial federation consensus functions, *Math. Soc. Sci.*, 20, 51-71.

Newman ME. (2004) A Fast algorithm for detecting community structure in networks. *Phys Rev E Stat Nonlin Soft Matter Phys*, 69:066133.

Nijenhuis A, Wilf H. (1978) *Combinatorial algorithms*, Academic Press.

Régnier S. (1965) Sur quelques aspects mathématiques des problèmes de classification automatique, *Mathématiques et Sciences humaines*, 82, 1983, 13-29, reprint of *I.C.C. bulletin*, 4, 1965, 175-191.

Ward J.H. (1963) Hierarchical grouping to optimize an objective function, *J. of American Statistical Association*, 58, 301, 236-244.

Zahn C.T. (1964) Approximating symmetric relations by equivalence relations, *SIAM J. on Appl. Math.*, 12, 840-847.