Representation and evaluation of partitions

Alain Guénoche[1] and Henri Garreta[2]

[1] Institut de Mathématiques de Luminy,
[2] Laboratoire d'Informatique Fondamentale de Marseille,
163, avenue de Luminy, 13288 Marseille Cedex 9, France.

Abstract. Many methods lead to build a partition of a finite set, given a metric. In this text we propose some criteria to evaluate the quality of each class and other parameters to compare several partitions on the same data set. Then, we indicate how to represent graphically a partition as a tree of boxes, each one containing information about the quality of a class.

1 Introduction

Let X be a finite set of N elements and D a distance array containing the proximity values between any two elements of X. Let us recall that a p-partition P on X is a set of p separate classes, such that:

$$\forall i, j \quad X_i \cap X_j = \emptyset \quad \text{and} \quad \bigcup_{i=1}^{p} X_i = X$$

Many methods permit to build classes and/or partitions on X. The most classical ones optimize some criterion over the set of partitions with a given number of classes (Hansen and Jaumard (1997)). Criteria commonly used are the split of classes, the diameter of the partition, the sum of (squared) distances between elements in different classes or a function of inertia, computed as the sum of squared distances to a center (real or virtual) (Guénoche (2002)). Unfortunately, most of these criteria lead to NP-hard optimization problems (Brucker (1978)) and sub-optimal solutions are calculated. A projection into an euclidian space, using a multidimensional scaling or a factorial method is also commonly realized; then, a center method (k-means) is applied to the set of the projected points. More rarely, a sequential clustering method is applied (Hansen et al. (1995)); it consists in building a class as homogeneous as possible, then to iterate the procedure on X minus this class, until there is no class sufficiently homogeneous. In that case, we do not get a partition on X, but only on the clustered elements. Other very efficient techniques (Self Organisation Map (Kohonen (1982)), or Adaptative Quality-based Clustering (Thijs et al. (2001)) are often used, particularly for clustering large sets of proteins from their expression levels.

Whatever is the method, we are confronted with two problems, (i) the representation of the partitions, and (ii) the measurement of the quality of

both classes and partition. They are often represented by lists of elements or by hypergraphs, that is a very poor and difficult way to understand partitions. This is a reason why many users prefer to apply a hierarchical clustering method and to have a look to the classes in the tree, because there are several very good softwares to display them.

In paragraph 1, we introduce several criteria to evaluate the quality of a class. In paragraph 2, we indicate other criteria for comparing partitions; they are not linked to those that are commonly used to build partitions (split, diameter or inertia) and they are also independent of the cardinality of the classes. In paragraph 3, we explain the representation of a partition by a "tree of boxes" , each box being characterized by the quality values of a class. An application, `QualiPart`, will be briefly presented.

2 Evaluating classes

The set X endowed with a distance D is a metric discrete space. In the Combinatorial Data Analysis approach (Arabie and Hubert (1996)) it is considered as a complete graph G, having X as its set of vertices and the edges being valued by D. We denote by *threshold graph at level s* the graph $G_{\leq s}$ with X as vertices and as edges all pairs (x, y) such that $D(x, y) \leq s$. Let Δ be the diameter of G that is the largest distance value.

For each class $X_k \subseteq X$ with $n_k = |X_k|$ elements, one can evaluate its homogeneity by calculating the following parameters:

- The *connectivity threshold* s_k. Each element x_i has a nearest neighbor at distance d_i. The threshold s_k of X_k is the largest value of d_i for x_i belonging to X_k.

$$s_k = \underset{x_i \in X_k}{Max} \quad \underset{x_j \in X_k}{Min} \quad D(x_i, x_j)$$

  It is also the length of the longest edge of a minimal spanning tree of the distance reduced to this class. If, in G, all the edges longer than or equal to s_k are removed, X_k is no more connected and this class is not consistent with the threshold.
- The *rate of links*. They correspond to pairs in X_k, having a length less than or equal to the connectivity threshold. The smallest value is $2/n_k$ (in the case where there is just a tree) and the largest one is 1 (when there is no value greater than s_k). The less it is, the weaker is the class, as a result of a chaining effect.
- The *diameter* δ_k. We indicate the ratio δ_k/Δ that must be as small as possible. All the links of X_k are less than or equal to this diameter and, at this threshold, X_k is a clique of $G_{\leq \delta_k}$. We shall remark that if the distance reduced to X_k fulfills the ultrametric condition, the connectivity threshold is equal to the diameter.

- The *size of the largest clique* of $G_{\leq s_k}$ included in X_k. The larger this cardinality, the stronger is the class. Unfortunately, the MaxClique-problem is NP-hard. Consequently, we just evaluate an interval. The lower bound is provided building a greedy clique: we start from a vertex having maximum degree and we add, as long as it exists, a vertex with maximum degree adjacent to all the previous elements in this clique. The upper bound q is the greatest number of elements of X_k having, in $G_{\leq s_k}$, a degree greater than or equal to $q-1$; it means that they have $q-1$ links not greater than s_k which might realize a clique (which is not tested).
- The *rate of well designed triples*. We only consider triples made of two elements x and y belonging to X_k and one element z out of it. Such a triple is well designed iff

$$D(x,y) \leq Min\{D(x,z), D(y,z)\}$$

  This condition is not satisfied when an element out of the class is closer to an element in the class than to another element in the class.

In our approach, we consider a class to be homogeneous if it has a diameter δ_k clearly less than Δ and not too large compared to s_k; the ratio of links not greater than s_k would be high, close to 50%; the maximal clique would contain from third to half the number of elements; the rate of well designed triples will be greater than 50%. If the class concatenates elements just closed to a single one, that is when there is a *chaining effect*, the diameter is high, the ratio of links weak and there are many cliques with very few elements.

3 Comparing partitions

To compare the quality of several partitions build from the same data set, we must define criteria not identical to those that can be eventually optimized when constructing one of these partitions. Moreover, these quality criteria must be independent of the number of elements in the classes. They must neither favor balanced partitions i.e., having classes with approximatively the same number of elements, nor those that present many singletons. We have selected four parameters that fulfill these conditions. They are based on the principle that a partition P is good if large distance values are between-class links and if small values are within-class links.

Whatever the number of classes, a partition P on X induces a bipartition of the pairs of X elements: on the one hand the between-class pairs correspond the external edges, and on the other hand the within-class pairs correspond to internal edges. Let $(L_e|L_i)$ be this bipartition, and N_e and N_i be the number of elements of each part.

$$N_e = \tfrac{1}{2} \sum_{k=1}^{p} |X_k| \, (N - |X_k|) \qquad N_i = \tfrac{1}{2} \sum_{k=1}^{p} |X_k| \, (|X_k| - 1).$$

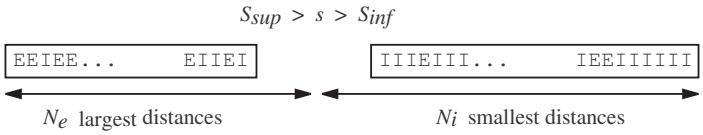

Fig. 1. Two bipartitions of the distance values ranked in decreasing order: external links (E) and internal links (I).

Evidently, we get $N_e + N_i = \frac{N(N-1)}{2}$, this quantity being denoted N_2 in the following.

These numbers induce another bipartition of the N_2 pairs in X (see Fig. 1). After ranking distance values in decreasing order, on the left side we have the N_e largest values and on the right side the N_i smallest values. This bipartition is denoted $(G_d|S_d)$. A perfect partition on X, according to the previous principle yields two identical bipartitions on pairs. Our two first parameters measure the similarity of these two bipartitions.

3.1 The rate of agreements on pairs

This is the percentage of pairs belonging to L_e and G_d or to L_i and S_d. They designate the largest distances used as external links and the smallest distances used as internal links, the difference between large (G_d) and small (S_d) being made at the threshold of the N_e-th distance value. As there can be some ties, we must define precisely what are the thresholds. First, we rank the distance values in decreasing order. Let s be the N_e-th largest distance value, and s' the next one $(s \geq s')$. If $s > s'$, we define $S_{sup} = s$ and $S_{inf} = s'$. Else, S_{sup} is the smallest value strictly greater than s and S_{inf} is the greatest value strictly lower than s. Between S_{sup} and S_{inf} either there is no further distance value, or there are values all equal to s. The *rate of agreements on pairs* is calculated by counting the pairs belonging to the intersection of the two bipartitions:

$Inter = 0$
For all pairs (x, y)
    If $Clas(x) \neq Clas(y)$ and $D(x, y) \geq S_{sup}$, $Inter = Inter + 1$
    If $Clas(x) = Clas(y)$ and $D(x, y) \leq S_{inf}$, $Inter = Inter + 1$
$\tau(a) = 1 - \frac{Inter}{N_2}$

3.2 The rate of weight

It is computed from the sums of distances on each of the four classes of pairs, respectively denoted by $\sum(L_e)$, $\sum(L_i)$, $\sum(G_d)$ and $\sum(S_d)$, as follows :

$$\tau(w) = \frac{\sum(L_e)}{\sum(G_d)} \times \frac{\sum(S_d)}{\sum(L_i)}.$$

These two ratios correspond to the weight of external links divided by the maximum that could be realized with N_e distance values and to the minimal weight of N_i edges divided by the weight of the internal links. Both are lower than or equal to 1 and so is the *rate of weight*. A rate close to 1 means that the between-class links belong to the largest edges and the intra-class links have been selected from the smallest ones. Consequently, this partition is one of the best possible ones.

Partitions maximizing the split have large values for this criterion, since they aim to put as external edges as many large distance values as possible. But it is not so for the rate of agreements on pairs, because large distance may link elements belonging to the same class.

3.3 The ratio of average lengths

Without depending on the cardinality of the classes, there is a very simple criterion to evaluate the quality of a partition; it is the ratio of the average lengths of the external edges and the average lengths of the internal edges:

$$\theta(l) = \frac{\frac{\sum(L_e)}{N_e}}{\frac{\sum(L_i)}{N_i}}$$

For a good partition, we can expect that it is greater than 1. The greatest feasible value is obtained replacing L_e and L_i by G_d and S_d in this formula. When $\theta(l)$ is large, the two types of edges are different and the partition is consistent. We have observed that if we select, for random euclidian or boolean distances, N points without cluster structure, this ratio is rarely larger than 1.3. But if the points are selected in balls centered on an axis and passing through the origin or in orthogonal hyperplans, this ratio is generally around 2 (Guénoche (2002)).

3.4 The ratio of well designed triples

This corresponding criterion for classes can be easily extended to partitions. Let T be the set of triples having just two elements in the same class.

$$|T| = \frac{1}{2}\sum_{k=1}^{p} |X_k| \left(|X_k| - 1\right) \left(N - |X_k|\right)$$

When the split of a class is lower than its diameter the rate of triples is lower than 1. Let x and y be in the same class and z in another one. We evaluate

$$\tau(t) = \frac{|\{x, y, z\} \in T \text{ such that } D(x, y) \leq Min\left\{D(x, z), D(y, z)\right\}|}{|T|}$$

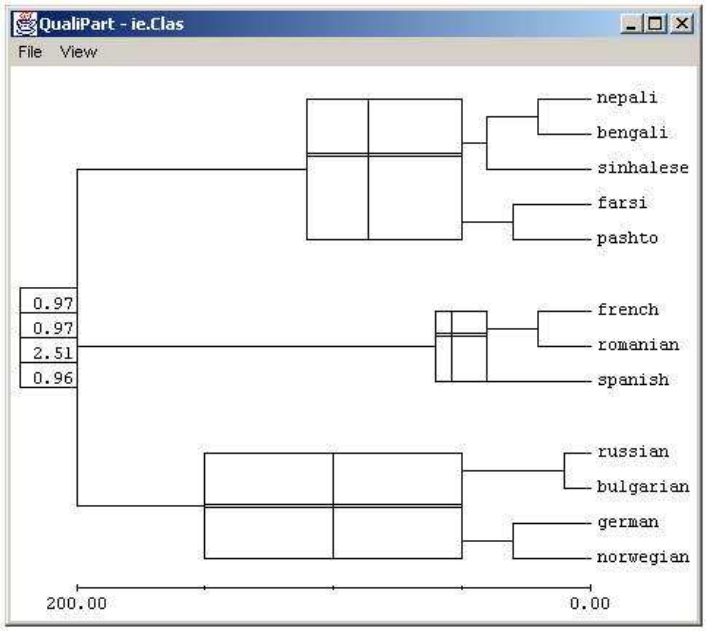

Fig. 2. Tree of boxes representing a partition in three classes of indo-european languages.

4 Representation of partitions

Classification trees or dendrograms are the most readable representation for hierarchies. They have a root, each leaf corresponding to an element has a label that can be written horizontally, if the tree expands vertically and if the horizontal axis is turned from large to small values (Cf. `NJPlot` [Perrière et al. 96]).

Partitions are also hierarchies with just one intermediate level between the elements and the whole set. So, in the tree style, but with much more information, we represent partitions as a tree of boxes. The root corresponds, as usual, to X to which as many rectangles (or boxes) as classes are linked. The horizontal axis is also scaled in the decreasing order. The root is placed at the position of Δ, the diameter of X, and all the leaves at position 0. Each class is included between two distance values, its threshold of connectivity s_k and its diameter δ_k. The larger the difference, the larger is the box and the class is less homogeneous. The height is just proportional to the number of elements.

We have seen that for a distance threshold lower than s_k, the class is disconnected. The subclasses can be represented by the single linkage hierarchy. It seems to be the best way to indicate their organization, and the level of the last union in this hierarchy is exactly s_k. So there remains, between levels

```
3
5 nepali bengali sinhalese farsi pashto
3 french romanian spanish
4 russian bulgarian german norwegian
```

Fig. 3. A partition of indo-european languages in three classes.

s_k and 0, the suitable place to draw it. The elements are ranked in an order compatible with this hierarchy.

Within the box, we draw two lines. Horizontally, it corresponds to the number of elements of the largest clique. When the lower and upper bounds are different, we take the average. A double line is placed near the top when it is high and near the bottom when it is small. Vertically, it corresponds to the rate of pairs having a distance not greater than s_k. This percentage varies along the width of the box. The line is drawn near the diameter when this rate is high and near the connectivity threshold in the opposite case.

Finally we display the four criteria values which qualify the partition in a box linked to the root.

5 Technical overview

An application, `QualiPart`, program has been written in Java. It uses two files ; the first one is for the classes and the second one for the distance array. The classes must be disjoint (it is tested) and their union form the complete set X. The first record in the file indicates the number of classes. Each class is described by its number of elements followed by the list of the labels of elements, separated by at least one space. The file given in Fig. 3 is the one represented by the tree of boxes in Fig. 2.

The distance file is in the PHILIP format. It must contain, in any order, all the labels referenced in the classes and labels must respect the same upper/lower case. In this file, there can be other elements absent in the classes. The useful distance array is extracted.

The program calculates first the criteria values for classes then those which evaluate the partition. All this values are shown in a text window, that can be saved as a text file. Simultaneously, it displays the tree of boxes. When clicking in a box, the corresponding class only remains on the screen and its criteria values are displayed in a message box. When clicking outside of the box, the full tree appears again. This drawings can be saved in PostScript format to be printed.

This application can be obtained from the authors, sending a e-mail to `garreta@luminy.univ-mrs.fr`.

Acknowledgements

This research was supported by the "Origines de l'homme, des langages et des langues" (OHLL) program. We also would like to thank an anonymous referee.

References

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

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

GUENOCHE, A. (2002): Partitions optimisées selon différents critères. *Mathématiques, Informatique et Sciences humaines*, to appear.

HANSEN, P. and JAUMARD, B. (1997): Cluster analysis and mathematical programming. *Mathematical Programming, 79*, 191-215.

HANSEN, P., JAUMARD, B., and MLADENOVIC, M. (1995): How to choose K entities among N. In: DIMAC Series in Discrete Mathematics and Theoretical Computer Science, 19, 105-115.

HUBERT, L.J. (1974): Spanning trees and aspects of clustering. *British J. of Math. Statist. Psychol.*, 27, 14-28.

KOHONEN, T. (1982): Self-organized formation of topologically correct feature maps. *Biol. Cybern.*, 43, 59-69.

PERRIERE, G. and GOUY, M. (1996): WWW-Query: An on-line retrieval system for biological sequence banks. *Biochimie*, 78, 364-369.

THIJS, G., MOREAU, Y., DE SMET, F., MATHYS, J., LESCOT, M., ROMBAUTS, S., ROUZE, P., DE MOOR, B., and MARCHAL, K. (2001): INCLUSive: INtegrated Clustering, Upstream sequence retrieval and motif Sampling. *Bioinformatics*, 2001, to appear.