Descriptive concept extraction with exceptions by hybrid clustering

Marie-Jeanne Lesot
Laboratoire d'Informatique de Paris 6
Université Pierre et Marie Curie
8 rue du capitaine Scott
75 015 Paris, France
E-mail: Marie-Jeanne.Lesot@lip6.fr

Bernadette Bouchon-Meunier
Laboratoire d'Informatique de Paris 6
Université Pierre et Marie Curie
8 rue du capitaine Scott
75 015 Paris, France
E-mail: Bernadette.Bouchon-Meunier@lip6.fr

***Abstract*— Natural concept modelling aims at representing numerically semantic knowledge ; generally, experts are asked to provide examples of linguistic terms associated with numerical data descriptions. We propose to exploit directly non labelled databases to extract the concepts that enable a semantic description of the data. Our method consists in identifying the subgroups corresponding to the concepts and then representing them as fuzzy subsets. For the identification step, we propose an algorithm based on a conjugate iterative use of the single linkage hierarchical clustering algorithm and the fuzzy c-means, that explicitely takes into account both a separability objective and a compactness aim; the description step builds membership functions as generalized gaussians. The adequacy of the results with spontaneous descriptions is illustrated on artificial and real databases.**

I. Introduction

Natural concept modelling aims at establishing a link between numerical data (temperature values for instance) and semantic knowledge expressed through linguistic terms (as "cold", "warm" and "hot" e.g.). Usually, these relationships are modelled as fuzzy subsets defined on the numerical domain. They are determined either point by point using labelled examples given by pools of experts [1] or globally using fuzzy partition inference techniques [6], [7], [16] which look for fuzzy subdivisions of the numerical domain correlated with the known data labels.

We consider a similar task in an unsupervised framework where no labels are provided: the data distribution itself contains information about the concepts which allows a semantic description of the dataset. For instance, it can be learnt from a dataset whether it can be described solely with two concepts, e.g. labelled as "low" and "high", or if it also requires a third concept, e.g. "intermediary"; then models of these concepts can be provided as fuzzy subsets leading to a fuzzy characterization of the data.

The concept extraction task can be divided into two steps: the first one consists in identifying the datapoints associated with each concept, and the second one in describing the extracted concepts with membership functions. A step of linguistic labelling can then take place to associate a semantic term to each extracted concept; this phase involves a human expert and is not considered here. The first stage is close to the clustering task [11], [10]: it must also decompose the dataset into homogeneous and distinct subgroups, corresponding to the concepts. Yet, there exists a difference regarding outliers or atypical data: in a concept modelling framework, they must be considered as minor but significant concepts, associated with linguistic expressions such as "abnormally low" and they should be present in the description as any other cluster.

Therefore, we propose the Outlier Preserving Clustering Algorithm (OPCA), that makes no difference between classic clusters (as detected by clustering algorithms), one-point clusters, corresponding to outliers (as provided by outlier detection techniques, e.g. [2], [12]) and lastly intermediate groups, corresponding to small isolated sets of similar outliers (which may be overlooked by both clustering and outlier detection methods). With this aim, OPCA explicitely takes into account simultaneously the clustering double objective, i.e. both compactness and separability of the obtained clusters: separability makes it possible to identify small sets of atypical data, and compactness to find homogeneous clusters. Therefore, OPCA is based on the combination of the single linkage hierarchical clustering algorithm and the fuzzy c-means.

After the clustering step, the second phase of descriptive concept extraction consists in describing the identified concepts with membership functions. We define them as instances of a parametric family, the generalized Gaussian functions.

The paper is organized as follows: after considering classic clustering algorithms in section 2, section 3 describes the method we propose to identify the concepts and section 4 the construction of the associated representative fuzzy subsets. Section 5 presents the results obtained on artificial and real datasets.

II. Outlier Handling

In this section, we consider how classic clustering algorithms handle outliers and introduce the justification for an algorithm taking into account simultaneously the compactness and separability objectives of clustering.

A. Fuzzy c-Means

The fuzzy c-means algorithm [3], or *fcm*, is an extension of the classic k-means algorithm which provides membership

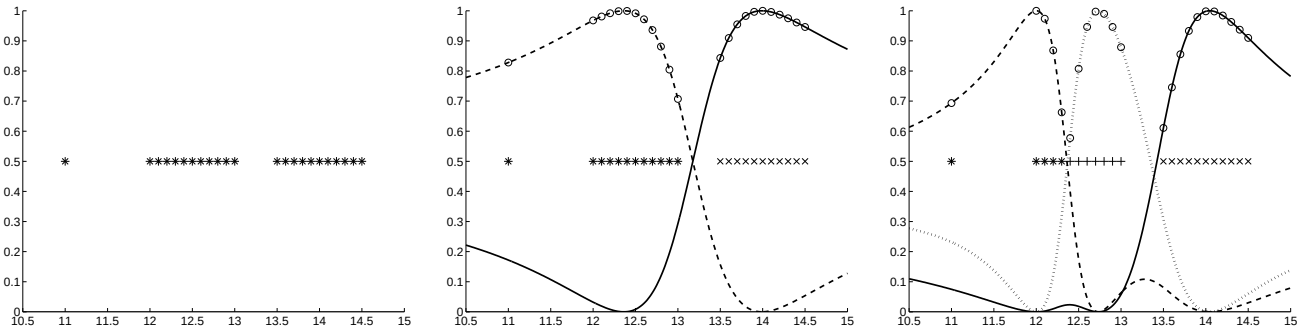


Fig. 1. Fuzzy c-means: (a) Considered one-dimension dataset; (b) Clustering results and membership functions for $c = 2$; (c) For $c = 3$.

degrees for each datapoint and each cluster, rather than crisp assignments. As compared to k-means, it can lead to more relevant results in case of poorly separated clusters and allows a lower sensitivity to initialization.

The way it handles outliers is illustrated on figure 1 for a one-dimension artificial datataset which can be decomposed into three clusters, interpreted as, from right to left, "high", "low" and "abnormal", the last term accounting for the specificity of the leftmost datapoint. When one applies *fcm* with $c = 2$ (fig. 1b), the outlier is assigned to the left group with a high membership degree and the clustering result ignores its specificity. If $c = 3$ (fig. 1c), the obtained clusters split the central group into two, instead of isolating the outlier (this result is independent of initialization).

This problem cannot be presented in the classic formulation where the outlier perturbs the clustering result: on the contrary, its handling is perturbed by other data and it cannot be isolated. One can explain these results by two pecularities of *fcm* [10]: the first reason is the fact that they do not take into account the separability of the groups. Indeed, the cost function they optimize penalizes a lack of compactness, i.e. a high variance of the clusters, but not a similarity between data assigned to different clusters. Therefore, with $c = 3$, they converge to the configuration of fig. 1 which indeed defines homogeneous clusters although they lack separability. A second reason comes from the definition of the cluster centers which involve all datapoints, with an influence weighted by their membership degrees, and lead to groups having the same size: small group centers cannot resist the influence of the numerous points which do not belong to their cluster and they are attracted towards the other datapoints.

B. Weighting Modification and Robust Algorithms

Some variants of *fcm* have been proposed to reduce the influence of datapoints which do not belong to a cluster, so that they do not intervene in its center definition [19], [9], [21]. They do not yet enable the algorithm to identify outliers, i.e. to define clusters centered on isolated datapoints: they mainly act on larger groups whose centers are not influenced by outliers, and thus better represent these large groups.

This addresses a robustness objective [5] which aims at reducing the influence of outliers, considered as perturbating datapoints that should be excluded. Some other variants of *fcm* are explicitely based on robust M-estimators incorporated in the cost function [8], the possibilistic c-means [13] can be seen in this framework. Another approach consists in replacing the traditional normal distributions by multivariate t-distributions [17], which are longer tailed distributions and thus give less weight to outliers in the parameter determination.

On the contrary, in a concept modelling framework, outliers are to be seen as the perturbed data, which other points prevent from modelling.

C. Noise Models

Some algorithms aim at identifying outliers in a specific subgroup which is then excluded from modelling: one can apply outlier detection methods e.g. [2], [12] as a preliminary step to clustering; Davé defines the noise clustering algorithm [4] based on the addition of a "noise cluster" which groups the data that are badly represented by "normal" clusters; likewise, Saint-Jean and Frélicot [20] decompose each cluster into 2 parts corresponding to its main component, learnt in a robust way, and its noisy points, then grouped in a noise cluster.

Using such an approach, it would be possible to perform a second clustering step on the noise cluster to identify relevant subgroups among outliers. We consider the problem from a different point of view which is based on the other interpretation of the *fcm* failure and consists in incorporating a separability objective in the clustering step. This enables to handle directly outlier subgroups as any other group without distinguishing between them.

D. Hierarchical Clustering Algorithms

Unlike the previous partitional algorithms which provide an optimal decomposition in a predefined number of clusters, hierarchical clustering [11] provides a whole nested sequence of decompositions: agglomerative hierarchical algorithms (AHC) start with each point in a distinct singleton cluster and progressively merge groups until obtaining a single cluster containing all data; each intermediary state can then be used as a clustering result. Divisive algorithms process the other way round, progressively splitting clusters; they are

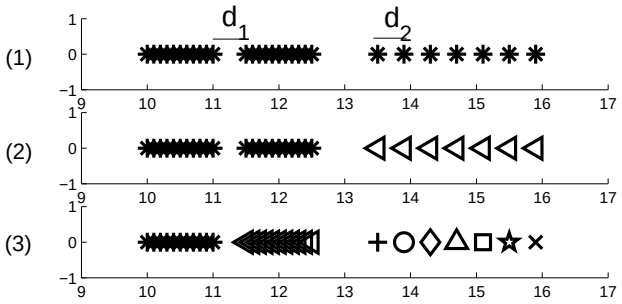


Fig. 2. Single linkage hierarchical algorithm AHC_{min}, unsensitivity to local contexts : (1) Dataset, $d_1 = d_2 = 0.4$; (2) Threshold $s = 0.9$; (3) $s = 0.3$. Each symbol corresponds to a different cluster.

usually computationnally more demandful, we do not consider them in the following. The various agglomerative algorithms differ by the merging criterion: some merge the least different clusters, where the distance between two candidates A and B is defined as $d(A, B) = op(\{d(a, b),\ a \in A, b \in B\})$, op is an aggregator which can be the *minimum*, in the single linkage strategy, the *maximum* in the complete linkage case, or the average. The Ward criterion carries out the merge which minimizes the increase of variance.

The merging criterion determines the properties of the obtained clusters: complete linkage minimizes the cluster diameter and thus leads to compact groups, the Ward criterion builds homogeneous groups with low variance[1]. The single linkage strategy is sensitive to the cluster separability as it is defined in terms of the minimal distance between data assigned to different groups. This property enables the algorithm to detect isolated small groups, which makes it a candidate algorithm for the descriptive concept extraction task.

However, single linkage AHC, which we will denote AHC_{min}, cannot be applied directly for two main reasons. First, it tends to build elongated clusters, in particular if natural groups are linked through a chain of neighbor data (*chaining effect*, cf. [11]). Indeed, the algorithm does not take into account a compactness objective and it only optimizes a separability criterion. Besides, it identifies outliers at a global level, without taking into account local information, and it is not adapted to data with variable densities: as any hierarchical algorithm, its stopping criterion is a threshold on the cost associated with each merge, i.e. a candidate merge between two groups A and B is carried out if its cost $d(A, B)$ is lower than a threshold s. For AHC_{min}, the merge cost only involves pairs of datapoints, but not their context: if the distance between two points is below s, they will be assigned to the same group, whatever their context is. This is illustrated on figure 2 where it seems natural to distinguish three groups; yet the distances d_1 and d_2 being equal, they are handled the same way. If $s > d_1$ (graph 2), the merge leads to 2 groups, otherwise, one gets 9 clusters. The global definition of s prevents the algorithm from comparing the values of d_1 to the local distance scale

Note that other criteria, as for instance complete linkage or Ward criteria, handle this case correctly, but they do not behave as desired for the outlier identification as they do not detect isolated small groups.

III. Descriptive Concepts Identification

The descriptive concept extraction can be seen as a clustering task submitted to constraints concerning the local outlier handling. Our approach consists in defining an algorithm which explicitely takes into account separability and compactness, whereas usually separability only appears in the validity criteria used to select the optimal number of clusters. It makes it possible to handle outliers as any other data subgroup. As each clustering algorithm is sensitive to a specific type of clusters, we propose to combine them to take advantage of their respective properties to define OPCA, an Outlier Preserving Clustering Algorithm. More precisely, we propose to combine in an iterative process the single linkage algorithm with the fuzzy c-means algorithm so as to exploit the ability of *fcm* to build compact clusters and that of AHC_{min} to extract well separated groups; the iterative process enables the algorithm to define outliers on a local scale.

This section presents OPCA, detailing the criteria used to determine which algorithm is to be applied and their appropriate parameters (respectively merging threshold for AHC_{min} and c-value for *fcm*).

A. Algorithm Outline

OPCA, which is detailed in table I, considers a group G obtained from previous steps and divides it by the most appropriate clustering algorithm. If G is separable, i.e. if G contains well separated subgroups, it is decomposed by AHC_{min}. Indeed, this algorithm is sensitive to data distribution gaps and it isolates separated groups. If G has a low separability, but still has a low compactness, it may correspond to a chaining case, which cannot be handled correctly by AHC_{min}; as it should nevertheless be decomposed, G is submitted to *fcm*. This process is then applied iteratively to each obtained cluster.

This process corresponds to a divisive approach as it progressively splits clusters, applying the most appropriate algorithm at each level.

B. Algorithm Selection Criteria

AHC_{min} is applied if G appears as separable, which we measure by the criterion $C_{AHC} = D_G/\delta_G$ where D_G is the maximal merging distance observed in G and δ_G the minimal distance[2] between distinct datapoints in G: D_G indicates the presence of gaps in the data distribution and δ_G indicates whether D_G is locally significant. If G corresponds to a low density region, δ_G is high and a large gap D_G is required to entail the subdivision. C_{AHC} compares G to a uniformly distributed dataset, taken as the reference of a group which should not be divided, and for which $C_{AHC} = 1$.

[1] AHC with the Ward criterion is a hierarchical variant to the k-means algorithm.

[2] possibly the maximum between this value and a minimal significant distance d_m defined by the user to incorporate *a priori* knowledge.

TABLE I

DESCRIPTIVE CONCEPT EXTRACTION ALGORITHM.

Initialization
$G =$ dataset
  select the minimal significant distance d_m
         the threshold diameter d_M
         the merge proportion α
Algorithm
if G is separable according to C_{AHC}
   compute the threshold s^* defined by eq. (2)
   decompose G with AHC with threshold s^*
otherwise, if G has a low C_{diam} compactness
   compute the optimal cluster number c^* with C_{stab}
   if the subdivision is justified by $C_{fcm}(c^*)$
     decompose G with *fcm* and c^*
iterate on the obtained subgroups

Fcm are applied if they provide a gain, defined as the quality of the optimal *fcm* decomposition (for the choice of the optimal c value, denoted c^*, see next section). The gain is measured in terms of compactness, as the average standard deviation decrease [18]

$$C_{fcm} = \frac{\frac{1}{c^*}\sum_{r=1}^{c^*}\sigma(C_r)}{\frac{1}{c^*}\sigma(G)} \quad (1)$$

where $\sigma(C)$ denotes the standard deviation of a group C and C_r, $r = 1..c^*$ the clusters built by *fcm*. The condition $C_{fcm} < 1$ also justifies a decomposition by comparing G to a uniformly distributed dataset.

We apply an additional criterion C_{diam} to decide whether *fcm* are justified, which enables the user to incorporate *a priori* knowledge: *fcm* are only applied if G diameter is higher than a threshold d_M (given by the user or computed as a function of G characteristics). d_M indicates a size below which a cluster may not be interpreted if it is not justified by a separability criterion; it defines a compactness criterion.

C. Algorithms Hyperparameter Selection

The hierarchical method produces a nested sequence of data decompositions, one of which must be selected as the result clustering; the stopping criterion is defined as a threshold s on the cost associated with each possible merge, so that only some of them are carried out. We define s as a function of the group characteristics, to allow a local adaptation to the data:

$$s = \bar{d} + \alpha\sigma(d) \quad (2)$$

where d is the vector of merge costs, $\bar{d}$ and $\sigma(d)$ are its mean and standard deviation, α is a hyperparameter which expresses the desired merging rate: assuming that d follows a Gaussian distribution, $\alpha = 2$ leads to perform the 95.5% cheapest possible merges, it corresponds to the expected outlier proportion, as outliers are associated with the most expensive merges. On account of s definition which depends on the iteration step, the outlier definition depends on their local context. This makes a difference with the noise clustering algorithm [4] where the distance determining the membership degree to the noise cluster depends on a single parameter, which does not allow adaptation to variable densities, as illustrated in section V.

To apply *fcm*, one must determine the desired number of clusters, c. Their exist many criteria, we apply a stability-based method: *fcm* produce highly stable results with respect to the random initialization, provided c is lower than the number of natural clusters. On the contrary, if c is higher, *fcm* exploit the unconstrained degrees of freedom and an instability of the cluster positions and the cost can be observed. Thus, denoting by J the *fcm* cost, for each c value, we compute $C_{stab} = \sigma(J)/\bar{J}$ quotient between the standard deviation and the average cost when initialization varies. The chosen value is the first one before destabilization.

D. Hyperparameters Role

Three hyperparameters are to be chosen, two can be set as functions of the dataset characteristics.

The minimal significant distance d_m amounts to data preprocessing as it defines datapoints which are to be grouped *a priori* and considered as a single point. By default, it can be defined as the minimal distance between distinct data but it enables the user to integrate *a priori* knowledge.

The threshold diameter d_M corresponds to the *a priori* maximal tolerated cluster size and indicates whether a group should be divided by *fcm* after a AHC_{min} use: it guarantees that the obtained subgroups can be semantically interpretated. The lower it is, the higher the final number of clusters. Its default value can be defined as a function of the whole dataset diameter d_X, we use $d_M = 1/4 d_X$.

Lastly, the hyperparameter α defines the proportion of possible merges that are carried out at each step of the iterative process: the higher it is, the lower the final number of clusters. Depending on the expected proportion of outliers, we usually choose $\alpha \in [3, 5]$.

IV. DESCRIPTIVE CONCEPT REPRESENTATION

OPCA identifies the underlying concepts in the dataset by determining the points associated with each of them. The next step consists in building fuzzy subsets that can model these concepts. The clustering phase made no hypothesis about the data dimensionality; for the representation step, we consider a classic attribute by attribute approach.

The fuzzy subsets which are to be built aim at representing the identified groups; thus, they differ from the fuzzy subsets associated with partition inference: in the second case, the membership functions must cover the whole numerical domain, as a consequence, the functions built by *fcm* are quite wide and not concentrated around the observed data (see fig. 1). On the contrary, in a description framework, the membership functions must focus on the data assigned to the clusters and not extrapolate in other regions.

We build the membership functions as instances of a parametric family of functions, which we choose to be the generalized Gaussian family, as proposed in [10]; other families

might have been selected as well. It is defined as

$$\mu(x) = \exp\left[-\left(\frac{\|x - \bar{x}(G)\|}{a}\right)^b\right] \quad (3)$$

where $(a, b) \in \mathcal{R}^2$; it corresponds to a Gaussian if $b = 2$. G denotes the group under study and $\bar{x}(G)$ its mean: as G is the result of a clustering step, it can be considered as homogeneous enough to be represented by its mean. The parameter a mainly influences the width of the plateau of high values, and b the decrease speed of the function: together, they control the core and support of the membership function.

It appears that a probabilistic representation ($b = 2$ and $a = \sigma(G)$) is not satisfying, as it assigns very low degrees to the furthest datapoints. Tests show that $b = 4$ leads to relevant functions; a is set as a function of the characteristics of the group G, so that the furthest data in the group have a degree equal 0.5. In that way, all members of the group have a membership superior or equal to 0.5.

The previous definition imposes a symmetric membership function, which may be a too constrained description. To have more flexibility, we define the membership function by two such functions with same value and derivative at $x = \bar{x}(G)$, representing respectively the data higher and lower than $\bar{x}(G)$, through two values a^+ and a^-.

In the case of clusters reduced to a single point, corresponding to an outlier, the previous choices cannot be applied. The point is therefore described by a triangular membership function, whose support is defined as a function of the minimal significant distance.

V. Results

We applied the proposed method to one- and two-dimension data to extract the descriptive concepts. In all cases, we used the default values for d_m and d_M.

Figure 3 presents the results obtained with $\alpha = 3$ on the artificial datasets of figures 1, 2 and a third more complex dataset: one can notice that in all three cases the built groups correspond to intuitive divisions and can be semantically interpreted ; on fig. 3(a), for instance, it highlights three concepts which could be linguistically described as "high", "low" and "abnormally low". Fig. 3(b) shows that OPCA is able to take into account local contexts to define separability between groups. Lastly fig. 3(c) illustrates a more complex case with varying densities and both global and local outliers, for which the result provided by OPCA corresponds to an intuitive decomposition. Fig. 3(d) show the clustering results obtained by the noise clustering algorithm [4] on this base for 3 different parameter values: being based on a single distance hyperparameter, it cannot take into account the local density and identify both outliers as belonging to the noise cluster.

Figure 4 shows results obtained with an artificial two-dimension dataset, generated by two Gaussian distributions to which two outliers of coordinates (-1, 0.8) and (1, -0.2) have been added. OPCA with $\alpha = 4.5$ detects the two major trends, isolate three minor singleton groups, corresponding to the two outliers and a locally isolated point in the left lower corner, and lastly identifies a small group of isolated points that can be interpreted as the extreme case of the major cluster.

Figure 5 represents the profiles built from real evaluations of two student classes to a same test. One can interpret these profiles and notice for instance that they have the same global structure, although the righthand group is less homogeneous: it contains two students having difficulties but it has a larger best student group. Thus the descriptive concept extraction enables the user to obtain semantic information about the class levels.

We applied the method to a dataset from the website[3] of the *Institut National de la Statistique et des Études Économiques* (INSEE) regarding the population of French regions in 2001; the obtained profile (fig. 6) is more complex than the previous ones. It highlights the existence of 4 "overpopulated" regions (corresponding to Rhône, Bouches du Rhône, Paris and Nord) a large group "densely populated", 5 intermediary categories, an important group "few inhabited" et lastly an outlier "sparsely populated" (corresponding to Lozère). As previously, it enables the user to extract a semantic information and to interpret the observed numerical values.

VI. Conclusion

Automatic descriptive concept extraction defines profiles describing numerical dataset. If an expert associates a linguistic label to the identified concepts, the method enables the user to get a semantic characterization of data and a representation of natural categories, without requiring the expert to label all datapoints.

The algorithm we propose to address this task is based on the combination in an iterative process of the fuzzy c-means with the single linkage hierarchical clustering algorithm: OPCA enables the user to handle similarly classic concepts associated with large groups, smaller clusters and outliers. Moreover, it uses a local definition of outliers which takes into account the context of each point. Each cluster is then described by a generalized Gaussian membership function which characterizes each extracted concept.

Beside the group constitution, OPCA provides a richer information through the iterative process and the chronological account of the groups constitution: ongoing work aims at exploiting this information to define exceptionality degrees [15] which can help determining the linguistic labels and in particular to choose fuzzy modifiers [14].

Acknowledgments

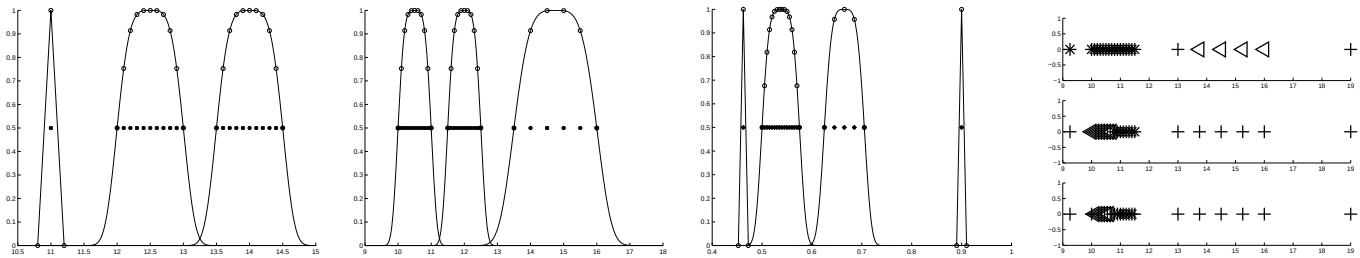
This research was partially funded by the RNTL research project ACEDU.

References

[1] N. Aladenise and B. Bouchon-Meunier. Acquisition de connaissances imparfaites : mise en évidence d'une fonction d'appartenance. *Revue Internationale de systémique*, 11(1):109–127, 1997.

[2] V. Barnett and T. Lewis. *Outliers in statistical data*. Wiley and Sons, 1994.

[3] http://www.insee.fr



(a) (b) (c) (d)

Fig. 3. Profiles obtained with the hybrid method for artificial databases, represented on fig. 1, 2, and a more complex case illustrating that the method takes into account local information. Fig. (d) illustrates the clustering results obtained on the database of fig. 3(c) with the noise clustering algorithm [4], for three hyperparametrer values, the cross corresponds to the noise cluster.

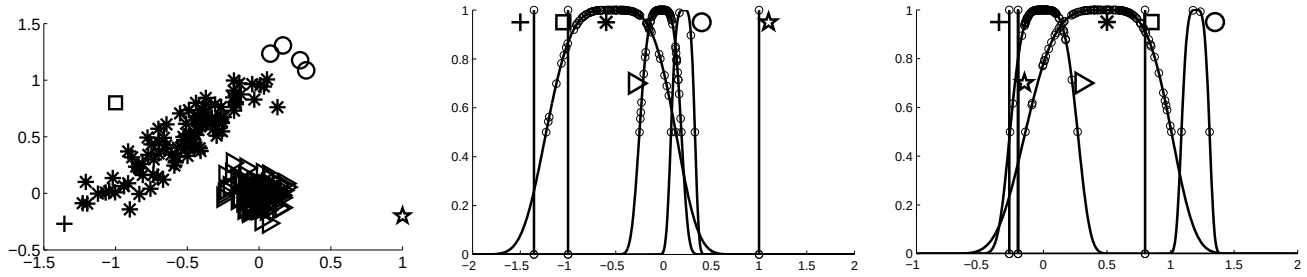


Fig. 4. Artificial two-dimension dataset. Left, built clusters, center and right, profiles for attribute 1 and 2 resp.; the symbols near the curves indicate the cluster each one represents.

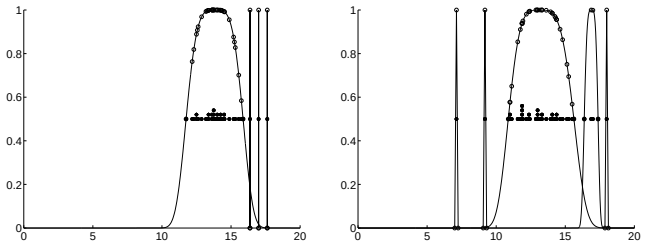


Fig. 5. Profiles of two students classes described by their mark to a same test.

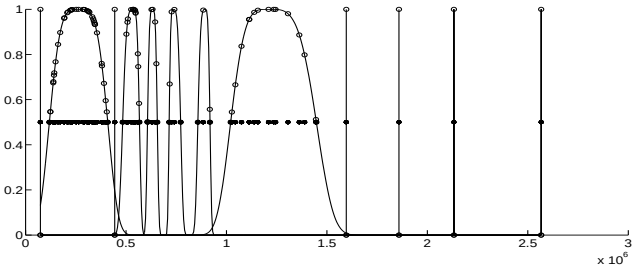


Fig. 6. Profile of the French metropolitan regions, described by their population.

[3] J. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithm*. Plenum, New York, 1981.

[4] R. Davé. Characterization and detection of noise in clustering. *Pattern Recognition Letters*, 12:657–664, 1991.

[5] R. Davé and R. Krishnapuram. Robust clustering methods: a unified view. *IEEE Transactions on fuzzy systems*, 5(2):270–293, 1997.

[6] T. Van de Merckt. Decision trees in numerical attribute spaces. In *Proc. of IJCAI'93*, pages 1016–1021, 1993.

[7] J. Dougherty, R. Kohavi, and M. Sahami. Supervised and unsupervised discretization of continuous features. In Morgan-Kaufman, editor, *Proc. of the 12th Int. Conf. on Machine Learning*, pages 194–202, 1995.

[8] H. Frigui and R. Krishnapuram. A robust competitive clustering algorithm with applications in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):450–465, 1999.

[9] F. Höppner and F. Klawonn. A new approach to fuzzy partitioning. In *Proc. of IFSA World Congress and 20th NAFIPS Int. Conf.*, pages 1419–1424, 2001.

[10] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis, Methods for classification, data analysis and image recognition*. Wiley, 2000.

[11] A. Jain, M. Murty, and P. Flynn. Data clustering: a review. *ACM Computing survey*, 31(3):264–323, 1999.

[12] E. Knorr, R. Ng, and V. Tucakov. Distance based outliers : algorithms and applications. *Very Large Data Bases Journal*, 8(3–4):237–253, 2000.

[13] R. Krishnapuram and J. Keller. A possibilistic approach to clustering. *IEEE Transactions on fuzzy systems*, 1:98–110, 1993.

[14] A. Laurent, C. Marsala, and B. Bouchon. Improvement of the interpretability of fuzzy rule based systems: Quantifiers, similarities and aggregators. *LNCS in Modelling with Words*, 2873:102–123, 2003.

[15] M.J. Lesot and B. Bouchon-Meunier. Cluster characterization through a representativity measure. In *Proc. FQAS'04*, Lyon, 2004.

[16] C. Marsala and B. Bouchon-Meunier. Fuzzy partitioning using mathematical morphology in a learning scheme. In *Proc. of FUZZ'IEEE'96*, 1996.

[17] G. McLachlan and D. Peel. Robust cluster analysis via mixtures of multivariate t-distributions. *LNCS*, 1451:658–666, 1998.

[18] M. Rezaee, B. Lelieveldt, and J. Reiber. A new cluster validity index for the fuzzy *c*-means. *Pattern Recognition Letters*, 19:237–246, 1998.

[19] P. Rousseeuw, E. Trauwaert, and L. Kaufman. Fuzzy clustering with high contrast. *Journal of Computational and Applied Mathematics*, 64:81–90, 1995.

[20] C. Saint-Jean and C. Frélicot. An hybrid parametric model for semi-supervised robust clustering. In *Int. Conf. on Recent Developments in Mixture Modelling (MIXTURES)*, Hambourg, Germany, 2001.

[21] K.L. Wu and M.S. Yang. Alternating c-means clustering algorithms. *Pattern recognition*, 35:2267–2278, 2002.