Fuzzy Prototypes Based on Typicality Degrees

Marie-Jeanne Lesot[1], Laure Mouillet[1,2], and Bernadette Bouchon-Meunier[1]

[1] LIP6, 8 rue du capitaine Scott, 75 015 Paris, France;
{Marie-Jeanne.Lesot, Bernadette.Bouchon-Meunier}@lip6.fr
[2] Thales Communications, 160 Bd de Valmy, BP 82, 92704 Colombes Cedex,
France; Laure.Mouillet@fr.thalesgroup.com

This paper considers the task of constructing fuzzy prototypes for numerical data in order to characterize the data subgroups obtained after a clustering step. The proposed solution is motivated by the will of describing prototypes with a richer representation than point-based methods, and also to provide a characterization of the groups that catches not only the common features of the data pertaining to a group, but also their specificity. It transposes a method that has been designed for fuzzy data to numerical data, based on a prior computation of typicality degrees that are defined according to concepts used in cognitive science and psychology. The paper discusses the construction of prototypes and how their desirable semantics and properties can guide the selection of the various operators involved in the construction process.

1 Introduction

Clustering [9, 8] provides a simplified representation of a dataset by decomposing it into homogeneous and distinct subgroups called clusters. Each cluster can then be described by a prototype: it is a unique individual which characterizes the data subgroup; the set of prototypes can be used to summarize the initial dataset. Our paper focuses on the prototype construction.

Most of the time, each cluster is mapped to a unique point, called its center. It can be defined in various ways, for instance as the group average – as in the k-means algorithm – or as a weighted average – as in the fuzzy c-means. This choice is based on the assumption that the average characterizes the group, which is not always the case; other points can be chosen, as for instance the median or the *Most Typical Value* [5].

Our hypothesis is that a prototype intrinsically is a *fuzzy* concept: human reasoning is not based on precise numerical values, but rather on imprecise notions. For instance, considering data describing the height of persons from various countries, it cannot be said that the typical French person is 1.6834m tall (fictitious value), which is the kind of result one would get using an average-based definition. It seems more appropriate to say that the typical French person is “rather tall”. This linguistic term conveys an imprecise notion which is best modelled using the fuzzy subset framework: the latter makes it possible to formalize categories with flexible and unclear boundaries, and thus prototypes as naturally handled by human beings.

Therefore, we consider the task of building fuzzy prototypes, defined as individuals described by fuzzy attribute values. The problem is then to construct, for each group, a fuzzy set describing the typical values of the numerical data pertaining to the group.

Rifqi [16, 15] proposes a method to build fuzzy prototypes for fuzzy data, i.e. data described by a set of attributes with values defined by fuzzy sets. This method rests on a definition of typicality inspired from cognitive science and psychology: Rosch [18] showed that all members of a category do not have the same representativeness and that the typicality of an instance depends both on its resemblance to the individuals pertaining to the same group and its dissimilarity with the individuals pertaining to the other groups. Our aim is to adapt this method to crisp data.

The paper is organized as follows: section 2 describes methods that provide enriched representatives for data subgroups, as compared to point-based descriptions. Section 3 presents the method we propose to handle crisp data and it discusses the notion of prototype and its associated properties. Lastly section 4 illustrates the results obtained on a real dataset.

2 State of the Art

2.1 Enriching Point Description

A first approach to obtain rich cluster descriptions is to apply fuzzy clustering algorithms [8]: they provide fuzzy subsets for each cluster. Yet these fuzzy subsets represent an imprecise decomposition of the database and model datapoints which have unclear membership and partially belong to several clusters simultaneously. Fuzzy prototypes are to be described by other fuzzy subsets: they do not aim at describing the cluster as a whole and representing all its points, but at extracting its most typical features and defining a relevant summary to provide a characterization of the subgroup. As a consequence, prototypes are expected to be more specific fuzzy subsets than the fuzzy clusters.

Thus fuzzy clustering methods provide a distribution in addition to the cluster centers, but this distribution cannot be applied directly to define fuzzy prototypes. Likewise, other point-based prototypes can be extended to distributions, but these distributions do not describe prototypes: one can associate a gaussian distribution to the arithmetic mean; yet, it is defined in such a way that each point in the cluster has a sufficiently high probability and it does not aim at defining a representative characterizing the group itself.

Table 1. Typicality-based algorithm to build a fuzzy prototype for a cluster g [15].

Given a resemblance measure r and a dissimilarity measure d [16],

1. Compute the typicality degree of each individual x in the group g
   a) Compute the internal resemblance $R_g(x)$ as the aggregation of the resemblance of x to the other individuals pertaining to the same group: $R_g(x) = ag_{y \in g}(r(x, y))$
   b) Compute the external dissimilarity $D_g(x)$ as the aggregation of the dissimilarity of x to the individuals pertaining to the other groups: $D_g(x) = ag_{y \notin g}(d(x, y))$
   c) The typicality $T_g(x)$ is the result of the aggregation of $R_g(x)$ and $D_g(x)$
2. The prototype is constructed by aggregating the individuals that are "typical enough", i.e. individuals whose typicality degree is higher than a pre-defined threshold.

More generally, fuzzy subset elicitation methods [1, 13] are techniques that are explicitely designed to provide fuzzy subsets describing the data. Some of them involve interaction with a human expert, others are based on partitioning methods [12, 6, 7]. Many belong to the parametric framework, i.e. consist in deciding on a desired form for the membership function, e.g. trapezoidal, gaussian or fuzzy c-means like, and optimizing its parameters so as to obtain a satisfying representative; the difficulty is then to define an appropriate cost function.

The prototype construction method we propose belongs to the elicitation technique framework, but it considers a specific characterization task: it aims at building a membership function which summarizes the data subgroup and does not describe it globally. It is based on the definition of typicality degrees: the membership functions are not directly constructed from the data but from an abstraction of the data in the form of typicality degrees.

2.2 Typicality-Based Approach

The method proposed in [16, 15] to construct fuzzy prototypes for fuzzy data uses the notion of typicality defined by Rosch [18]: the typicality of a datapoint is a function of both its within-class resemblance and its dissimilarity to other classes, as detailed in table 1. In the sequel, we refer to it as *Rosch typicality.* The prototype is then defined as an aggregation of the most typical datapoints.

It is to be noted that other typicality definitions exist, as the one used by Pal *et al.* [14] or the one underlying the Most Typical Value [5]. Yet these definitions are closer to "exceptionality" coefficients than typicality degrees, insofar as they are only based on the notion of a deviation with respect to a center. They can be interpreted as internal resemblance measures, which makes them a special case of the *Rosch typicality.*

3 Typicality-Based Prototype for Crisp Data

In this section, we discuss a method based on the *Rosch typicality* for crisp data. Considering crisp data makes a major difference with fuzzy data: in the latter case indeed, a fuzzy prototype is described with the same "language" as the considered data, i.e. fuzzy sets. In the case of crisp data, the prototype has a richer description than the datapoints.

After some general remarks on the typicality approach to fuzzy prototypes, we discuss the three steps of the algorithm described in table 1, with regard to the desired semantics and properties of the prototype. At each step, we discuss the parameters and the choices that have to be made and we illustrate their consequences on the classic iris database[1] [4]. In the first place, the resemblance and dissimilarity definitions must be defined for the crisp case. Then, their aggregation to compute the typicality degree has to be studied. The last step consists in determining how typical values are aggregated so as to obtain the prototype.

3.1 General Remarks

In the following, we consider that each group can be represented by a single prototype. This assumption is valid in our clustering framework: groups correspond to clusters and are thus homogeneous enough to be associated with a single representative.

It is to be noticed that the definition of a fuzzy prototype based on typicality degrees entails the definition of three different distributions:

- the membership function which indicates the extent to which a datapoint belongs to a cluster; this distribution is only available if clusters are obtained using a fuzzy clustering algorithm;
- the typicality distribution which indicates the extent to which a datapoint is typical of a group;
- the membership function describing the prototype itself.

Relationships between these distributions can be established: for instance, it seems natural to consider that for each group and each datapoint, the typicality is lower than the membership degree to the cluster. This expresses the fact that a datapoint which has a low membership cannot be typical; it also imposes that a point which is totally typical of a group totally belongs to it.

Lastly, prior to the prototype construction, one must decide between two approaches : the algorithm can be performed either on the individual as a whole, or it can be applied attribute by attribute. Taking into account all attributes makes it possible to represent correlations between attributes, which

[1] Note that for this database, labels are provided, so a preliminary clustering phase is not necessary, we construct one prototype for each of the three classes.

Table 2. Some examples of resemblance and dissimilarity measures for two vectors x and y; $\text{dist}(x, y)$ denotes a distance between x and y, Z is a normalizing factor, z a normalizing function, γ, σ, d_M and Γ are hyperparameters.

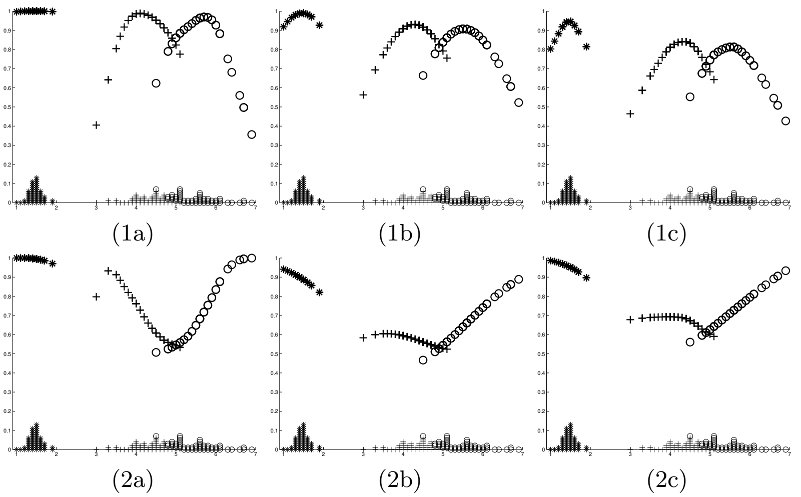
Resemblance measures, R	
scalar product	$(x \cdot y)$
distance-based	$1 - \frac{1}{Z}\text{dist}(x, y)$
polynomial	$\left(\frac{(x \cdot y)}{Z} + 1\right)^{\gamma}$
gaussian	$\exp\left(-\frac{1}{2\sigma^2}\text{dist}(x, y)^2\right)$
Fermi-Dirac [17]	$z\left(\frac{1}{1 + \exp\frac{\text{dist}(x,y) - d_M}{\Gamma}}\right)$
Dissimilarity measures, D	
normalized distance	$\frac{1}{Z}\text{dist}(x, y)$
measures depending on a resemblance measure	$1 - r(x, y)$ $\frac{1}{1 + r(x, y)}$

can provide useful information about the prototype. On the other hand, working attribute by attribute can emphasize attributes that have no typical values for a given group, and can therefore be removed from the prototype description, thereby simplifying the group description.

3.2 Similarity and dissimilarity measures

In the case of fuzzy data, the resemblance and dissimilarity measures involved in the first step of prototype construction apply to fuzzy sets. They are defined in the formal framework of comparison measures [16] as functions with values in the interval $[0, 1]$ depending on a triple characterization of the relative position of the two compared subsets A and B: they depend on their common elements (i.e. a fuzzy set measure of the intersection $A \cap B$) and on their distinctive elements (a fuzzy set measure of their set differences $A - B$ and $B - A$). For crisp data, the range of possibilities is reduced because the information on the relative positions of two points can only be measured with one quantity corresponding to a distance (or equivalently a dot product). Table 2 mentions some possibilities.

Figure 1 illustrates three types of measures, using the iris database. For visualization sake, we only consider a single attribute, the petal length (third one); the histograms represent the datapoint distribution. On the graphs on



(1a) (1b) (1c)

(2a) (2b) (2c)

Fig. 1. Examples of comparison measures for the third attribute of the iris database. (1) Internal resemblance (a) Fermi-Dirac [17], (b) gaussian, (c) distance-based; (2) External dissimilarity (a) Fermi-Dirac, (b) gaussian, (c) normalized distance.

the first (resp. second) row, the y-axis represents the internal resemblance (resp. external dissimilarity) measure for each datapoint. One can observe that the three measures lead to different values, but usually to the same datapoint ranking within subgroups. These graphs show that the resemblance is maximal for a central point, whereas the most dissimilar points correspond to extreme values. Whichever measure is used, the middle subgroup does not appear very dissimilar as it is too close to the other clusters.

As in any learning problem, the choice of a distance measure has a major influence on the results. It seems natural to select it according to the distance used in the clustering step, but the choice may also be guided by the problem at hand, which may impose the semantics of the resemblance or the dissimilarity. Each measure has its own semantics: the Fermi-Dirac measure [17] enables the user to monitor the decrease speed as a function of the distance increase using two hyperparameters, which makes it a flexible comparison measure; in the multi-dimensional case, the polynomial resemblance takes into account correlations between attributes.

Another interesting issue is the duality of the resemblance and the dissimilarity measures. In the comparison measure framework [16], dissimilarity measures are defined independently from the resemblance measures; in particular, they are not necessarily dual, i.e. defined as the complement to 1 of each other. In the crisp case, both are defined as a function of a single quantity (a distance), thus duality seems a natural property of the couple. Yet it is still possible to define non-dual measures, the difference can simply come from normalizing factors: two datapoints of the same class can be considered to be totally similar at a distance d_1 which differs from the distance d_2 at which points from two classes have a null dissimilarity. Such a choice implies defining a double semantics for the problem, and two different approaches to data comparison. On the contrary, duality can be justified considering the case of a datapoint x situated at an equal distance both to a point y pertaining to the same group and a point z belonging to another class: duality then expresses the fact that x is as similar to y as it is dissimilar to y.

3.3 Aggregation Operators

The second step of prototype construction consists in aggregating the internal resemblance and external dissimilarity to obtain a typicality degree. The aggregation has a strong impact on the semantics of the prototype. There exist many works on aggregation operators [3, 2], we consider here some properties which may be desired and their consequences on the aggregator choice.

Classic methods such as the mean, the Most Typical Value [5] or the probabilistic approach, only take into account data pertaining to the group to determine the group representative. They all choose the central element to represent the group, with varying definitions of “centrality”. In our framework, this behavior can be obtained in the degenerate case where the aggregator only takes into account the internal resemblance and not the external dissimilarity. In this case, the typicality degree corresponds to the resemblance measure as illustrated by the first row on Fig. 1. In the same way, although this is not as intuitive, it is conceivable to choose a degenerate aggregator that only takes the external dissimilarity into account: a datapoint is considered typical if it makes it possible to conclude about the class membership; the typicality notion can then be interpreted as some kind of discrimination power (cf. second row on Fig. 1).

In real data applications, the central and the discriminative elements often coincide insofar as the center of a class is a discriminative element [18]. Yet if the prototype is constructed attribute by attribute, a non discriminative center may be encountered. The relative weight given to the centrality and to the discrimination power in the typicality degree computation can be tuned thanks to the aggregation operator.

In the sequel, we study the impact of the aggregation operator on the prototype semantics taking up the aggregator categorization from [3] which distinguishes between constant and variable attitude aggregators. We illustrate their effects on Fig. 2, considering the iris third attribute and the Fermi-Dirac comparison measures (graphs (1a) and (2a) on Fig. 1).

Constant Behavior Operators

Constant attitude operators can be categorized into three groups, distinguishing conjunctive operators, such as the logical AND, disjunctive ones, such as the logical OR, and intermediate ones, also called tradeoff operators, such as weighted means.

Choosing a conjunctive operator (graph (a) on Fig. 2) implies that a datapoint can be considered typical if and only if it is both similar to the points of its cluster and distinct from other classes. This is a severe definition of typicality which leads to low values on average.

On the contrary, choosing a disjunctive operator implies that a datapoint can be considered typical either if it is discriminative or if it is central to the group. This leads to two kinds of typical points, as shown on graph (b) on Fig. 2: the extreme right points have a high typicality degree despite their low internal resemblance because of their high dissimilarity, the central point is typical for another reason, namely its high internal resemblance.

Lastly, tradeoff operators (graph (c) on Fig. 2) have an intermediate behavior, in particular, they possess the compensation property: a decrease in one criterion can be offset by an increase in another one for a constant result value. For instance, the extreme right point has a higher typicality degree than on graph (a) (min operator): its high dissimilarity compensates for its low resemblance.

Variable Behavior Aggregators

Variable behavior aggregators are conjunctive, disjunctive or tradeoff depending on the values that are aggregated as for instance the MICA operator [10]:

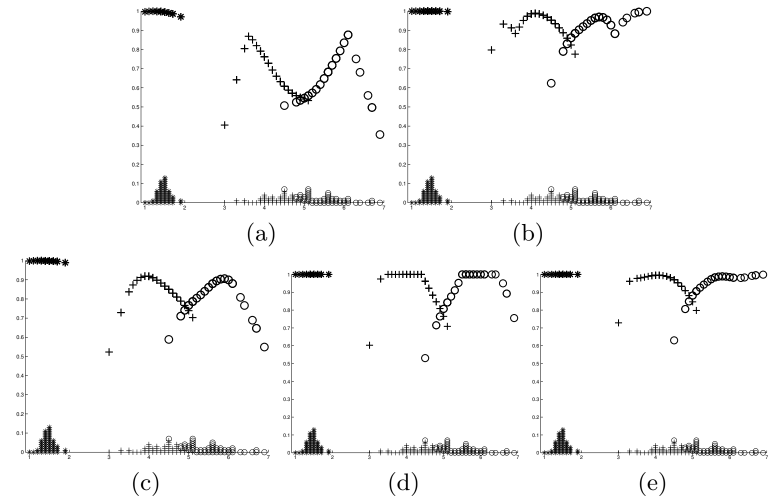
$$mica(x,y) = \max(0, \min(1, k + (x-k) + (y-k))) \tag{1}$$

where $k \in [0,1]$ is a hyperparameter which determines the position of the change in behavior: if both values are high, i.e. higher than the neutral value k, $mica(x,y) \geq \max(x,y)$, so the operator has a disjuntive behavior; if both values are lower than k, the operator has conjunctive behavior. As the MICA operator is disjunctive for high scores and conjunctive for low scores, it possesses the full reinforcement property [2]: two low values are aggregated into an even lower value and two high values lead to an even higher result. This is illustrated on graph (d) obtained with $k = 0.6$, where many points have a typicality of 1, due to their internal resemblance and external dissimilarity higher than the threshold value k. On the contrary, for the extreme left point of the right cluster, the obtained value is lower than the one obtained with the min operator.

The symmetric sum [19] is another example of variable attitude aggregator:

$$symSum(x,y) = \frac{g(x,y)}{g(x,y) + g(1-x, 1-y)} \tag{2}$$

where g is a continuous, increasing, positive function satisfying $g(0,0) = 0$. For this non-linear operator, the change in behavior occurs at threshold 0.5. For some choices of g, one also observes the full reinforcement property. Graph (e) on Fig. 2 illustrates this property with $g(x,y) = x \cdot y$.



(a) (b)

(c) (d) (e)

Fig. 2. Examples of aggregators applied with the Fermi-Dirac comparison measures on the 3rd attribute of the iris database. (a) $t = \min(R, D)$, (b) $t = \max(R, D)$, (c) $t = 0.7R + 0.3D$, (d) $t = mica(R, D)$ cf. (1) with $k = 0.6$, (e) $t = symSum(R, D)$ cf. (2) with $g(x, y) = x \cdot y$.

There is no way to define a priori which aggregation operator to choose: this choice depends on the considered application and the intended use of the constructed prototype.

3.4 Typical Data Aggregation

The last step consists in building the prototype itself; the objective is to define it as an aggregation of the most typical datapoints. Thus one must aggregate crisp values into a fuzzy subset, which is similar to the initial problem of building fuzzy prototypes characterizing crisp data. The difference is that the typicality degrees extracted from the initial data provide some kind of abstraction. Thus they may justify simple solutions, which are not applied to raw data, but to enriched data containing more information than the initial ones.

We propose to define two thresholds, τ_k and τ_s, that indicate the minimum typicality required to be included respectively in the prototype kernel and in the support. To perform the in-between interpolation, different solutions can be considered: one can simply use a parametric approach and build the trapezoidal membership function having the previous kernel and support. One can also perform a linear rescaling of the interval $[\tau_s, \tau_k]$ to $[0, 1]$ to get the membership degrees to the prototype. Note that in the attribute by attribute construction, both methods lead to the same prototypes.

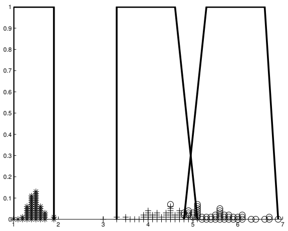


Fig. 3. Fuzzy prototype for the third attribute of the iris database, using Fermi-Dirac comparison measures and MICA aggretator.

Figure 3 illustrates the three prototypes obtained for the third attribute of the iris database. For this example, as the classes are quite homogeneous and well separated, almost all values have high membership degrees to the prototype of their class. The exceptions are located in the overlapping zone between two classes.

4 Experimental results

In this section, we consider a real two-dimensional dataset which describes the results of 150 students after two exams, shown on Fig. 4. We illustrate the construction of prototypes characterizing the various groups of students with the previously presented method. The prototypes have been constructed globally (not attribute by attribute) because there were obvious correlations between the two attributes.

Table 3 sums up the various notations and measures used in the sequel.

Clustering

The first step consists in clustering the dataset so as to decompose it into homogeneous subgroups. We applied the fuzzy c-means with an automatic choice of c based on the stability of the objective function [11], which led to $c = 5$. The results are shown on Fig. 4.

Resemblance and Dissimilarity Measures

The second step consists in computing the comparison measures. It was natural to base our resemblance and dissimilarity measures on the euclidian distance used in the clustering step, but they had to be normalized so as to take their values in $[0, 1]$. The normalization factor chosen for the dissimilarity measure indicates the distance from which two values are considered totally dissimilar; it is set to half the data diameter, the data diameter being the maximal distance between pairs of datapoints in the dataset.

Table 3. Notations and measures, where X denotes the set of examples, and $||x-y||$ the euclidian distance between x and y.

dissimilarity normalization factor	$Z_d = \frac{1}{2} \max_{x,y \in X} \|\|x-y\|\|$
dissimilarity measure	$d(x,y) = \min\left(1, \frac{\|\|x-y\|\|}{Z_d}\right)$
cluster diameter for cluster g	$diam(g) = \max_{x,y \in g} \|\|x-y\|\|$
resemblance normalization factor	$Z_r = \max_g diam(g)$
resemblance measure	$r(x,y) = \max\left(0, 1 - \frac{\|\|x-y\|\|}{Z_r}\right)$

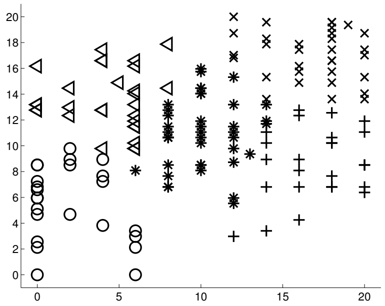


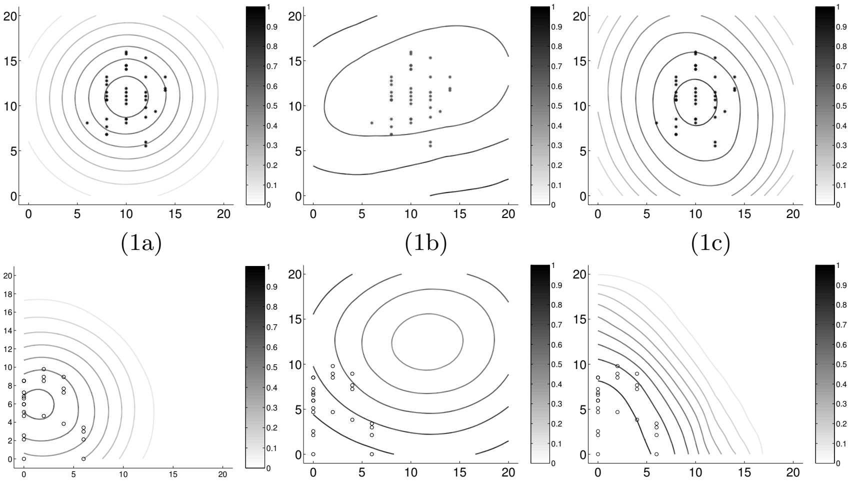
Fig. 4. Clustering results obtained on a 2D dataset. Each symbol depicts a different cluster.

The normalization factor chosen for the resemblance measure is the maximal cluster diameter, the diameter of a cluster g being defined as the maximal distance between pairs of datapoints pertaining to g.

The first two columns of Fig. 5 illustrate these choices for two clusters, the central and lower left ones (depicted by $*$ and $\circ$ on Fig. 4) and represent contour plots of the resemblance and dissimilarity measures respectively. The resemblance contours appear circular, which is due to the point distribution and the measure choice; the point maximizing the internal resemblance is the cluster average. The dissimilarity is maximal for extreme points, it is slightly higher for the left lower cluster than for the central one, which is too close to the four other clusters.

Typicality Degree Computation

We chose a variable behavior operator, the symmetric sum, introduced in (2), with $g(x,y) = x \cdot y$. This aggregator rewards datapoints that have a high internal resemblance (greater than the neutral value 0.5) and a high external dissimilarity; if both values are significantly high, they reinforce each other.



(1a) (1b) (1c)

(2a) (2b) (2c)

Fig. 5. Internal resemblance, external dissimilarity and typicality distribution for the central cluster (first row), depicted $*$ on Fig. 4, and the left lower cluster (second row), depicted by $\circ$.

Simultaneously, it sanctions datapoints that have a low internal resemblance (lower than 0.5) and a low external dissimilarity. In-between, a low internal resemblance can be offset by a high external dissimilarity and vice versa.

The results are illustrated on the last column of Fig. 5. It appears that the typicality distribution reflects the resemblance, modified to a certain extent by the dissimilarity influence. Thus typical values are not central values (maximizing the internal resemblance), but close values determined taking into account the external dissimilarity. These observations are consistent with the cognitive science results.

Typical Values Aggregation

The last step consists in aggregating the most typical values of each cluster into a fuzzy subset which characterizes the cluster. We consider that a datapoint whose typicality degree is higher than $\tau_k = 0.9$ totally belongs to the prototype, and we exclude from the prototype support the points having a typicality lower than $\tau_s = 0.7$. The interpolation between the so-defined kernel and support is performed through a linear rescaling of the interval $[\tau_s, \tau_k]$ on the interval $[0, 1]$.

The obtained results are illustrated on Fig. 6 which shows the contour plots of the five prototype membership functions. Their kernels contain values which simultaneously have a high internal resemblance and a low external dissimilarity. Our method enabled us to extract characterizations for the

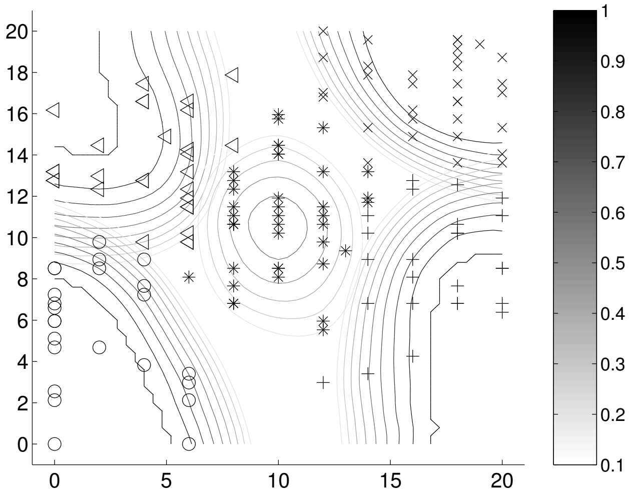


Fig. 6. Superimposition of the 5 obtained fuzzy prototypes.

five groups revealed by the clustering step. We think that these results can be improved if all the datapoints are not taken into account with the same importance in the typicality degrees computations: fuzzy clustering provides membership functions that indicate to what extent a datapoint belongs to several clusters. Intuitively, a datapoint that completely belongs to a group has to have more importance in the prototype computation than a datapoint that poorly belongs to it.

5 Conclusion

We propose a parametric fuzzy prototype construction method that consists in aggregating typical individuals to represent an homogeneous data subgroup. This method involves a certain number of parameters that must be chosen according to the desired semantics and properties of the prototype. Although these depend on the data and the application of the prototypes, it is possible to give insights on the methodology for choosing the appropriate parameters. For instance, if the prototype is used to classify new examples, its discrimination power should be emphasized, whereas if it is used to provide an explanation of the data, its centrality should be favored.

Our perspectives are to study in more detail the interrelations between the parameters and their possible redundancy. Another interesting issue is to study how the method can take into account fuzzy clusters: if they are provided by the clustering algorithm, the membership degrees to the various clusters could be used in the process, for instance to reduce the weight of datapoints that only partially belong to the cluster.

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. M. Detyniecki. *Mathematical aggregation operators and their application to video querying.* PhD thesis, Université de Paris VI, 2000.
3. V. Eude, B. Bouchon-Meunier, and E. Collain. Choix d'opérateurs d'agrégation pour l'évaluation de structures hiérarchisées. In *Proc. of IPMU'98*, pages 778–785, Paris, France, 1998.
4. R.A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:170–188, 1936.
5. M. Friedman, M. Ming, and A. Kandel. On the theory of typicality. *Int. Journal of Uncertainty, Fuzzyness and Knowledge-Based Systems*, 3(2):127–142, 1995.
6. A. Gyenesei. Fuzzy partitioning of quantitative attribute domains by a cluster goodness index. Technical Report 368, Turku Centre for Computer Science, 2000.
7. T.P. Hong and C.Y. Lee. Induction of fuzzy rules and membership functions from training examples. *Fuzzy Sets and Systems*, 84(1):33–47, 1996.
8. F. Höppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis, Methods for classification, data analysis and image recognition.* Wiley, 2000.
9. A. Jain, M. Murty, and P. Flynn. Data clustering: a review. *ACM Computing survey*, 31(3):264–323, 1999.
10. A. Kelman and R. Yager. On the application of a class of MICA operators. *Int. Journal of Uncertainty, Fuzzyness and Knowledge-Based Systems*, 7:113–126, 1995.
11. M.J. Lesot and B. Bouchon-Meunier. Descriptive concept extraction with exceptions by hybrid clustering. In *FuzzIEEE04*, Budapest, Hungary, 2004.
12. C. Marsala and B. Bouchon-Meunier. Fuzzy partitioning using mathematical morphology in a learning scheme. In *Proc. of the 5th IEEE Int. Conf. on Fuzzy Systems*, volume 2, New Orleans, USA, 1996.
13. S. Medasani, J. Kim, and R. Krishnapuram. An overview of membership function generation techniques for pattern recognition. *Int. Journal of Approximate Reasoning*, 19(3–4):391–417, 1998.
14. N. Pal, K. Pal, and J. Bezdek. A mixed c-means clustering model. In *FuzzIEEE97*, pages 11–21, 1997.
15. M. Rifqi. Constructing prototypes from large databases. In *Proc. of IPMU'96*, Granada, Spain, 1996.
16. M. Rifqi. *Mesures de comparaison, typicalité et classification d'objets flous : théorie et pratique.* PhD thesis, Université de Paris VI, 1996.
17. M. Rifqi, V. Berger, and B. Bouchon-Meunier. Discrimination power of measures of comparison. *Fuzzy sets and systems*, 110:189–196, 2000.
18. E. Rosch. Principles of categorization. In E. Rosch and B. Lloyd, editors, *Cognition and categorization*, pages 27–48. Lawrence Erlbaum associates, 1978.
19. W. Silvert. Symmetric summation: a class of operations on fuzzy sets. *IEEE Transactions on Systems, Man and Cybernetics*, 9:659–667, 1979.