A novel approach to fuzzy clustering based on a dissimilarity relation extracted from data using a TS system

Mario G.C.A. Cimino, Beatrice Lazzerini, Francesco Marcelloni*

Dipartimento di Ingegneria dell'Informazione: Elettronica, Informatica, Telecomunicazioni, University of Pisa, Via Diotisalvi 2, 56122 Pisa, Italy

Abstract

Clustering refers to the process of unsupervised partitioning of a data set based on a dissimilarity measure, which determines the cluster shape. Considering that cluster shapes may change from one cluster to another, it would be of the utmost importance to extract the dissimilarity measure directly from the data by means of a data model. On the other hand, a model construction requires some kind of supervision of the data structure, which is exactly what we look for during clustering. So, the lower the supervision degree used to build the data model, the more it makes sense to resort to a data model for clustering purposes. Conscious of this, we propose to exploit very few pairs of patterns with known dissimilarity to build a TS system which models the dissimilarity relation. Among other things, the rules of the TS system provide an intuitive description of the dissimilarity relation itself. Then we use the TS system to build a dissimilarity matrix which is fed as input to an unsupervised fuzzy relational clustering algorithm, denoted any relation clustering algorithm (ARCA), which partitions the data set based on the proximity of the vectors containing the dissimilarity values between each pattern and all the other patterns in the data set. We show that combining the TS system and the ARCA algorithm allows us to achieve high classification performance on a synthetic data set and on two real data sets. Further, we discuss how the rules of the TS system represent a sort of linguistic description of the dissimilarity relation.

Keywords: Fuzzy clustering; Fuzzy identification; Similarity relation

1. Introduction

The aim of cluster analysis is to organize a collection of patterns (usually represented as vectors of measurements, or points in a multidimensional space) into homogeneous groups (called *clusters*) based on pattern similarity [1,2]. Typically, similarity (more often dissimilarity) is expressed in terms of some distance function, such as the Euclidean distance or the Mahalanobis distance. The choice of the (dis)similarity measure induces the cluster shape and therefore determines the success of a clustering algorithm on the specific application domain. For instance, the Euclidean and Mahalanobis distances lead clustering algorithms to determine hyperspherical-shaped or hyperellipsoidal-shaped clusters, respectively. However, when applying clustering to data with irregular distribution, as it is often the case for image segmentation and pattern recognition [3], distance functions cannot adequately model dissimilarity [4–7]. Consider, for example, the dissimilarity between pixels of an image consisting of elements with irregular-shaped contours.

To solve this problem, some approaches can be found in the literature. For example, in Ref. [8], the dissimilarity between two points is defined as a function of their context, i.e., the set of points in the neighborhood of each such point. In Ref. [9], pre-defined concepts are used to define the "conceptual similarity" between points. In Ref. [10], Yang and Wu propose to adopt a total similarity related to the approximate density shape estimation as objective function of their clustering method.

* Corresponding author. Tel.: +39 050 2217678; fax: +39 050 2217600.
E-mail addresses: m.cimino@iet.unipi.it (M.G.C.A. Cimino), b.lazzerini@iet.unipi.it (B. Lazzerini), f.marcelloni@iet.unipi.it (F. Marcelloni).

In appearance-based vision, Jacobs et al. observe that classification systems, which can model human performance or use robust image matching methods, often exploit similarity judgement that is non-metric [11]. They show that existing classification methods can meet considerable difficulties when applied to non-metric similarity functions. They note, however, that, if an accurate choice of class representatives is performed, exemplar-based methods can be applied naturally and successfully. Thus, they propose a new approach which aims to retain both atypical (i.e., dissimilar from most other patterns belonging to the class) and boundary (i.e., belonging to the boundaries between classes) patterns. Further, they suggest to adopt the vector correlation between the distances from each image to other previously seen images as a good measure of how well an image can represent another in non-metric spaces.

In Ref. [12], Makrogiannis et al. introduce a region dissimilarity relation that combines feature-space and spatial information for color image segmentation. First, regions are produced from the original images by means of the Watershed transform. Each region is characterized by a vector of mean pixels intensities in the utilized color space. Regions are therefore clustered by using the mountain clustering method and the fuzzy C-means in sequence. Clusters can be regarded as robust and statistically reliable descriptors of local color properties. Then, each region is represented by the vector of membership values of the region to each cluster. This representation implicitly transforms the feature space to the space of membership values. In this space, the dissimilarity between two regions is computed, for instance, as the cosine between the two vectors which represent the regions. This space transformation allows introducing global information in the dissimilarity measure which is used for generating the minimal spanning tree and producing the final segmentation.

A different approach proposes to extract the dissimilarity relation directly from the data by guiding the extraction process itself with as little supervision as possible [13]. Following this approach, Hertz et al. suggest to learn distance functions by using a subset of labelled data [14]. In particular, they train binary classifiers with margins, defined over the product space of pairs of images, to discriminate between pairs belonging to the same class and pairs belonging to different classes. The signed margin is used as a distance function. Both support vector machines and boosting algorithms are used as product space classifiers. Using some benchmark databases from the UCI repository, the authors show that their approach significantly outperforms existing metric learning methods based on learning the Mahalanobis distance. Similarly, in previous papers [15,16], we extracted the dissimilarity relation directly from a few pairs of data with known dissimilarity rather than from pairs of data with known labels. Thus, our approach is more general than that adopted in Ref. [14]. More precisely, we adopted a multilayer perceptron (MLP). Once trained, the MLP can associate a dissimilarity value with each pair of patterns in the data set. Then, we used the dissimilarity measure generated by the MLP to guide an unsupervised fuzzy relational clustering algorithm. Though the results we obtained are better than those achieved by some widely used clustering algorithms based on spatial dissimilarity, two weak points can still be pointed out. First of all, the dissimilarity relation generated by the MLP is not interpretable in linguistic terms; then, owing to the generalization performed starting from a restricted number of known relationship values, the dissimilarity relation D produced by the MLP is, in general, neither irreflexive nor symmetric. Unfortunately, the most popular fuzzy relational clustering algorithms [17,18], any of which can be used to work on D, assume that D is at least a positive, irreflexive and symmetric square binary relation. This means that these algorithms can be applied to D, as we showed in Refs. [15,16], but their convergence to a reasonable partition is not guaranteed. In this paper we propose a solution to both the above problems. Indeed, we substitute the MLP network with a Takagi–Sugeno (TS) system [19], whose fuzzy rules are identified from a few pairs of patterns with known dissimilarity by using the method proposed in Ref. [20]. At the end of the identification phase, like the MLP, the TS system can associate a dissimilarity degree with each pair of patterns in the data, but, unlike the MLP, the TS system is capable of providing a sort of intuitive description of the dissimilarity relation.

Then, to make our approach independent of the characteristics of the relation generated by the TS model, we use the fuzzy relational clustering method, denoted any relation clustering algorithm (ARCA), recently proposed in Ref. [21], which can be applied to any type of relation matrix, with all guarantees of convergence. ARCA exploits the well-known fuzzy C-means (FCM) algorithm [22] to partition the data set based on the proximity of the vectors containing the dissimilarity values between each pattern and all the other patterns in the data set. We verified that ARCA produces partitions similar to the ones generated by the other fuzzy relational clustering algorithms, when these converge to a sound partition. On the other hand, as ARCA is based on the FCM algorithm, which has proved to be one of the most stable fuzzy clustering algorithms, ARCA is appreciably more stable than the other fuzzy relational clustering algorithms.

The effectiveness of the combination TS model–ARCA is shown using three examples of its application to a synthetic data set and to two public real data sets, respectively. We describe how our approach achieves very good clustering performance using a limited number of training samples. Further, we show how the TS model can provide an intuitive linguistic description of the dissimilarity relation.

We wish to point out that the method proposed in this paper is intended for use in all cases in which the dissimilarity relation can be learnt from a reasonably small portion of samples, which form the training set, using a fuzzy TS model. The capability of providing linguistic descriptions of the dissimilarity relation combined with the convergence guarantee of the relational clustering algorithm is the novelty of the method. The method works, in principle, with any kind of data set. In fact, as it unfolds the entire data onto as many dimensions as the number of data points in order to transform the relational clustering to object-based clustering, it is more appropriate for moderate-size data sets, typically containing up to a few hundreds of patterns. Actually, as it is well known, adopting distance functions more suitable for high-dimensional spaces in place of the Euclidean distance used by ARCA would allow us to alleviate the curse of dimensionality problem. We did not adopt this solution in the examples simply because it was not strictly necessary. As a final remark, we observe that, as the dimensionality of the data set increases, the interpretability of the set of fuzzy rules becomes less evident.

2. The TS approach to dissimilarity modelling

TS systems are a powerful fuzzy modelling technique. A TS system consists of a set of fuzzy rules [19]: the antecedent of each rule determines a region of the input space by means of a conjunction of fuzzy clauses that contain the input variables; the consequent is a mathematical function which approximates the behavior of the system to be identified in the region fixed by the antecedent. The generation of a TS model requires the following two steps: the *structure identification* and the *parameter identification*. The structure identification determines the number of rules and the input variables. The parameter identification estimates the parameters which define the membership functions of the fuzzy sets in the rule antecedents and the parameters which identify the consequent functions. The number of rules is generally determined based on a clustering algorithm, so that the number of rules is equal to the number of clusters which compose the partition assessed to be the best with respect to an appropriate validity index.

Let $Q = [\underline{x}_1, \ldots, \underline{x}_M]$ be the data set. The rules of the TS system used to model the dissimilarity relation have the following form:

$$\begin{aligned} r_i\colon &\text{If } X_{1,1} \text{ is } A_{i,1,1} \text{ and } \ldots X_{1,F} \text{ is } A_{i,1,F} \text{ and} \\ &X_{2,1} \text{ is } A_{i,2,1} \text{ and } \ldots X_{2,F} \text{ is } A_{i,2,F} \\ &\text{then } d_i = \underline{a}_{i,1}^T \underline{X}_1 + \underline{a}_{i,2}^T \underline{X}_2 + b_i,\ i = 1 \ldots C, \end{aligned}$$

where $\underline{X}_e = [X_{e,1}, \ldots, X_{e,F}]$, with $e = 1, 2$, are the two input variables of F components which represent the pair of patterns whose dissimilarity has to be evaluated, $A_{i,e,1}, \ldots, A_{i,e,F}$ are fuzzy sets defined on the domain of $X_{e,1}, \ldots, X_{e,F}$, respectively, $\underline{a}_{i,e}^T = [a_{i,e,1}, \ldots, a_{i,e,F}]$, with $a_{i,e,f} \in \Re$, and $b_i \in \Re$. The model output d, which represents the dissimilarity between the two input patterns, is computed by aggregating the conclusions inferred from the individual rules as follows:

$$d = \frac{\sum_{i=1}^{C} \beta_i d_i}{\sum_{i=1}^{C} \beta_i}, \tag{1}$$

where $\beta_i = \prod_{f=1}^{F} A_{i,1,f}(x_{j,f}) \prod_{f=1}^{F} A_{i,2,f}\,(x_{k,f})$ is the degree of activation of the ith rule, when the pair $(\underline{x}_j, \underline{x}_k)$ is fed as input to the rule.

The number C of rules, the fuzzy sets $A_{i,e,f}$ and the consequent functions of the rules are extracted from the data using a version of the method proposed in Ref. [20]. Let $T = \{\underline{z}_1, \ldots, \underline{z}_N\}$ be the set of known data, where $\underline{z}_h = [\underline{x}_i, \underline{x}_j, d_{i,j}] \in \Re^{2F+1}$, with $d_{i,j}$ the known dissimilarity between $\underline{x}_i$ and $\underline{x}_j$. First, the FCM algorithm is applied to T to determine a partition U of the input/output space [22]. The optimal number of clusters is computed by executing FCM with increasing values of the number C of clusters for values of the fuzzification constant m in {1.4, 1.6, 1.8, 2.0} and assessing the goodness of each resulting partition using the Xie–Beni index [23]. We plot the Xie–Beni index versus C and choose, as optimal number of clusters, the value of C corresponding to the first distinctive local minimum. Fuzzy sets $A_{i,e,f}$ are obtained by projecting the rows of the partition matrix U onto the fth component of the input variable $\underline{X}_e$ and approximating the projections by triangular membership functions defined as follows:

$$\begin{aligned} &A_{i,e,f}(X_{e,f}; l_{i,e,f}, m_{i,e,f}, r_{i,e,f}) \\ &\quad = \max\left(0, \min\left(\frac{X_{e,f}-l_{i,e,f}}{m_{i,e,f}-l_{i,e,f}}, \frac{r_{i,e,f}-X_{e,f}}{r_{i,e,f}-m_{i,e,f}}\right)\right) \end{aligned} \tag{2.a}$$

with $l_{i,e,f} < m_{i,e,f} < r_{i,e,f}$ real numbers on the domain of definition of $X_{e,f}$. In the cases of $l_{i,e,f} = m_{i,e,f} < r_{i,e,f}$, $l_{i,e,f} < m_{i,e,f} = r_{i,e,f}$ and $l_{i,e,f} = m_{i,e,f} = r_{i,e,f}$, formula (2.a) is not applicable and is replaced by the following three formulas (2.b)–(2.d), respectively:

$$\begin{aligned} &A_{i,e,f}(X_{e,f}; l_{i,e,f}, m_{i,e,f}, r_{i,e,f}) \\ &\quad = \begin{cases} \dfrac{r_{i,e,f} - X_{e,f}}{r_{i,e,f} - m_{i,e,f}} & \text{if } m_{i,e,f} \leqslant X_{e,f} < r_{i,e,f} \\ 0 & \text{otherwise,} \end{cases} \\ &\quad \text{if } l_{i,e,f} = m_{i,e,f} < r_{i,e,f}, \end{aligned} \tag{2.b}$$

$$\begin{aligned} &A_{i,e,f}(X_{e,f}; l_{i,e,f}, m_{i,e,f}, r_{i,e,f}) \\ &\quad = \begin{cases} \dfrac{X_{e,f} - l_{i,e,f}}{m_{i,e,f} - l_{i,e,f}} & \text{if } l_{i,e,f} < X_{e,f} \leqslant m_{i,e,f} \\ 0 & \text{otherwise,} \end{cases} \\ &\quad \text{if } l_{i,e,f} < m_{i,e,f} = r_{i,e,f}, \end{aligned} \tag{2.c}$$

$$\begin{aligned} &A_{i,e,f}(X_{e,f}; l_{i,e,f}, m_{i,e,f}, r_{i,e,f}) \\ &\quad = \begin{cases} 1 & \text{if } X_{e,f} = l_{i,e,f} = m_{i,e,f} = r_{i,e,f} \\ 0 & \text{otherwise,} \end{cases} \\ &\quad \text{if } l_{i,e,f} = m_{i,e,f} = r_{i,e,f}. \end{aligned} \tag{2.d}$$

We computed the parameter $m_{i,e,f}$, which corresponds to the abscissa of the vertex of the triangle, as the weighted average of the $X_{e,f}$ components of the training patterns, the weights being the corresponding membership values. Parameters $l_{i,e,f}$ and $r_{i,e,f}$ were obtained as intersection of

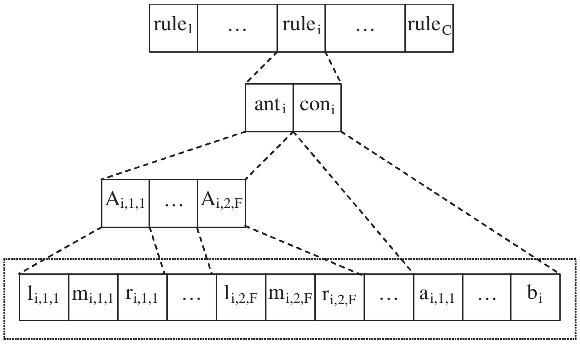

Fig. 1. The chromosome structure.

the $X_{e,f}$ axis with the lines obtained as linear regression of the membership values of the training patterns, respectively, on the left and the right sides of $m_{i,e,f}$. Obviously, if $l_{i,e,f}$ and $r_{i,e,f}$ are beyond the extremes of the definition domain of variable $X_{e,f}$, the sides of the triangles are truncated in correspondence to the extremes. The use of triangular functions allows easy interpretation of the fuzzy sets in linguistic terms. This characteristic will be useful to associate a meaning with the rules, as explained in Section 4. We note that formula (2.d) is used when a cluster is composed of a unique point (singleton): this case is however extremely improbable. Once the antecedent membership functions have been fixed, the consequent parameters $[\underline{a}_{i,1}, \underline{a}_{i,2}, b_i], i = 1 = \ldots C$, of each individual rule i are obtained as a local least squares estimate.

The strategy used so far to build the TS model is aimed at generating a rule base characterized by a number of interesting properties, such as moderate number of rules, membership functions distinguishable from each other, and space coverage, rather than at minimizing the model error. To improve possible poor performance of the system, we apply a genetic algorithm (GA) to tune simultaneously the parameters in the antecedent and consequent parts of each rule in a global optimization. To preserve the good properties of the fuzzy model, we impose that no gap exists in the partition of each input variable. Further, to preserve distinguishability we allow the parameters that define the fuzzy sets to vary within a range around their initial values. Each chromosome represents the entire fuzzy system, rule by rule, with the antecedent and consequent parts (see Fig. 1). Each rule antecedent consists of a sequence of $2 \cdot F$ triplets (l, m, r) of real numbers representing triangular membership functions, whereas each rule consequent contains $2 \cdot F + 1$ real numbers corresponding to the consequent parameters. The fitness value is the inverse of the mean square error (MSE) between the predicted output and the desired output over the training set.

We start with an initial population composed of 70 chromosomes generated as follows: the first chromosome codifies the system generated by the FCM, the others are obtained by perturbing the first chromosome randomly within the ranges fixed to maintain distinguishability. At each generation, the arithmetic crossover and the uniform mutation operators are applied with probabilities 0.8 and 0.6, respectively. Chromosomes to be mated are chosen by using the well-known roulette wheel selection method. At each generation, the offspring are checked against the aforementioned space coverage criterion. To speed up the convergence of the algorithm without significantly increasing the risk of premature convergence to local minima, we adopt the following acceptance mechanism: 40% of the new population is composed of offspring, whereas 60% consists of the best chromosomes of the previous population. When the average of the fitness values of all the individuals in the population is greater than 99.9% of the fitness value of the best individual or a prefixed number of iterations has been executed (6000 in the experiments), the GA is considered to have converged.

Once the TS model has been generated and optimized, we compute the dissimilarity value between each possible pair $(\underline{x}_i, \underline{x}_j)$ of patterns in the data set Q. Such dissimilarity values are provided as an $M \times M$ relation matrix $D = [d_{i,j}]$. The value $d_{i,j}$ represents the extent to which $\underline{x}_i$ is dissimilar to $\underline{x}_j$. Thus, the issue of partitioning patterns described through a set of meaningful features is transformed into the issue of partitioning patterns described through the values of their reciprocal relations. This issue is tackled by *relational clustering* in the literature. As in real applications clusters are generally overlapped and their boundaries are fuzzy rather than crisp, we consider fuzzy relational clustering algorithms.

3. The relational clustering algorithm

The most popular examples of fuzzy relational clustering are the Roubens' fuzzy non-metric model (FNM) [24], the Windham's assignment prototype (AP) model [25], the Hathaway et al.'s relational fuzzy C-means (RFCM) [26], the Hathaway and Bezdek's non-Euclidean relational fuzzy C-means (NERFCM) [27], the Kaufman and Rousseeuw's fuzzy analysis (FANNY) [28], the Krishnapuram et al.'s fuzzy C-medoids (FCMdd) [18], and Davé and Sen's fuzzy relational data clustering (FRC) [29]. All these algorithms assume (at least) that $D = [d_{i,j}]$ is a positive, irreflexive and symmetric fuzzy square binary dissimilarity relation, i.e., $\forall i, j \in [1 \ldots M], d_{i,j} \geqslant 0, d_{i,i} = 0$ and $d_{i,j} = d_{j,i}$. Unfortunately, the relation D produced by the TS model may be neither irreflexive nor symmetric, thus making the existing fuzzy relational clustering algorithms theoretically not applicable to this relation. Actually, as shown in Ref. [16], these algorithms can be applied, but their convergence to a reasonable partition is not guaranteed (see, for instance, Ref. [21]).

To make our approach independent of the relation generated by the TS model, we use the ARCA fuzzy relational clustering method, recently proposed by Corsini et al. [21]. This method is particularly suitable for our problem because it can be applied to any type of relation matrix. ARCA arises from the following observation: in relational clustering, each pattern $\underline{x}_i$ is defined by the values of the relations between $\underline{x}_i$ and all patterns in the data set. If the data set is composed of M patterns, each pattern $\underline{x}_i$ can be represented as a vector $\hat{\underline{x}}_i = [d_{i,1}, \dots, d_{i,M}]$ in $\Re^M$, where $d_{i,j}$ is the extent to which $\underline{x}_i$ is related to $\underline{x}_j$. Since a relational clustering algorithm should group patterns that are "closely related" to each other, and "not so closely" related to patterns in other clusters, as indicated by their relative relational degrees [17], we can obtain clusters by grouping patterns based on their closeness in the space $\Re^M$.

Let $B_1, \dots, B_C$ be a family of fuzzy clusters on Q. Then, the objective function minimized by ARCA is:

$$J_m(U, V) = \sum_{i=1}^{C} \sum_{k=1}^{M} u_{i,k}^m d^2(\hat{\underline{x}}_k, \hat{\underline{v}}_i) \tag{3}$$

under the constraints $u_{i,k} \in [0, 1], \forall i, k$, and $\sum_{i=1}^{C} u_{i,k} = 1, \forall k$, where m is the fuzzification constant, $U = [u_{i,k}]$ is a real $C \times M$ partition matrix, V is the set of cluster prototypes, $u_{i,k}$ is the membership value of $\underline{x}_k$ to cluster B_i, $d(\hat{\underline{x}}_k, \hat{\underline{v}}_i)$ denotes the Euclidean distance between the representations $\hat{\underline{x}}_k$ and $\hat{\underline{v}}_i$ in $\Re^M$ of the generic pattern $\underline{x}_k$ and the prototype $\underline{v}_i$ of cluster B_i. In our case, a prototype is a (possibly virtual) pattern whose relationship with all patterns of the data set is representative of the mutual relationships of a group of similar patterns. The function proposed in Eq. (3) coincides with the objective function of the classical FCM algorithm [22], when the patterns to be clustered are defined in the space $\Re^M$, and therefore it can be minimized by using the same formulas as in FCM. Thus, representing the $M \times M$ relation matrix as M vectors defined in the feature space $\Re^M$ allows transforming a relational clustering problem into an object clustering problem, which can be solved using the FCM algorithm. ARCA was tested on some public data sets, showing that the partitions obtained by ARCA are comparable to the ones generated, when applicable, by the most stable relational algorithms, namely RFCM and NERFCM [21]. Further, ARCA requires no particular constraint on the dissimilarity relation matrix, thus allowing its application to the dissimilarity relation generated by the TS fuzzy system.

In the experiments, we used $m = 2$ and $\varepsilon = 0.001$, where ε is the maximum difference between corresponding membership values in two subsequent iterations. Moreover, we implemented the ARCA algorithm in an efficient way in terms of both memory requirement and computation time, thanks to the use of the technique described in Ref. [30] to speed up the execution of FCM, which is part of ARCA.

4. Experimental results

We tested our approach on the synthetic data set shown in Fig. 2 and on two real data sets, namely the Iris and the Wisconsin Breast Cancer (WBC) data sets. For each data set, we carried out five experiments. We randomly extracted a pool of patterns (called *training pool*) from the data set. This pool was composed of 5%, 10%, 15%, 20% and 25% of the data set, respectively, in the five experiments. We assume to know the dissimilarity degrees between all the pairs that can be built from patterns in the training pool. Then, we build the training set by selecting a given number of pairs of patterns from the training pool. More precisely, assume that C is the number of clusters, which we expect to identify in the data set. Then, for each pattern $\underline{x}_i$ in the training pool, we form $q \cdot C$ pairs $(\underline{x}_i, \underline{x}_j)$, with $q \in [1 \dots 8]$, by randomly selecting $q \cdot C$ patterns $\underline{x}_j$ of the training pool as follows: q patterns are chosen among those with dissimilarity degree lower than 0.5 with $\underline{x}_i$, and the remaining $q \cdot (C-1)$ patterns are chosen among those with dissimilarity degree higher than 0.5. It is obvious that increasing values of q lead to better classification performance, but also to increasing execution times. In the data sets considered in this paper, we observed that $q = 5$ provides a good trade-off between classification accuracy and execution time. Let $d_{i,,j}$ be the degree of dissimilarity between $\underline{x}_i$ and $\underline{x}_j$. We insert both $[\underline{x}_i, \underline{x}_j, d_{i,j}]$ and $[\underline{x}_j, \underline{x}_i, d_{i,j}]$ into the training set.

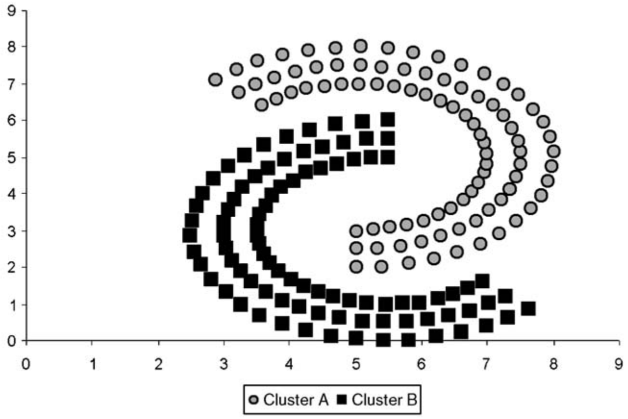

Fig. 2. The synthetic data set.

4.1. Synthetic data set

Fig. 2 shows the synthetic bi-dimensional data set, which consists of two classes. This data set was chosen because it is not easily managed by clustering algorithms based on spatial dissimilarity. For instance, both the Euclidean and the Mahalanobis distances lead the FCM algorithm [22] and the Gustaffson and Kessel's algorithm [31] to partition this data set with percentages of correctly partitioned points of 72.22% and 80.00%, respectively.[1] We carried out the five

[1] We refer to single seed-based algorithms, that is, algorithms which use a single seed to represent a cluster. Actually, multiple seeds-based algorithms can identify correctly the two classes, when the number of seeds is adequate. Anyway, the choice of the number of seeds requires previous estimation of point density and/or identification of cluster border points, which are not very easy to perform [32].

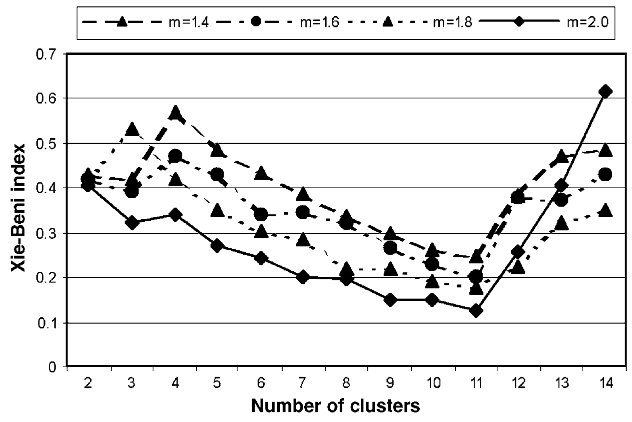

Fig. 3. The Xie–Beni index versus C.

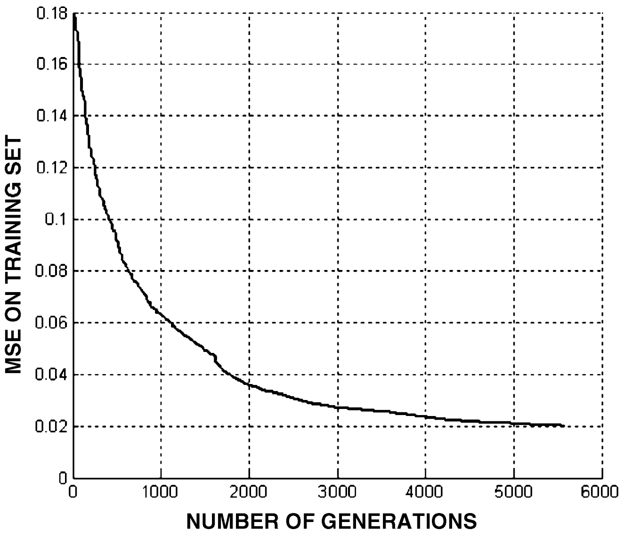

Fig. 4. Mean square error of the best chromosome on the training set.

experiments described above and, for each experiment, we executed ten trials. For the sake of simplicity, in the experiments, we used only 0 and 1 to express the dissimilarity degree of two input points belonging to the same class or to different classes, respectively. Please, note that we use the knowledge about classes just to assign dissimilarity degrees to pairs of points in the training pool. First, for each trial, we executed the FCM algorithm with values of the number C of clusters from 2 to 15 and values of m ranging in $\{1.4, 1.6, 1.8, 2.0\}$. Second, we plotted the Xie–Beni index versus C and chose, as optimal number of clusters, the value of C corresponding to the first distinctive local minimum. Fig. 3 shows an example of this plot for a trial with the training pool composed of 15% of the data. It can be observed that there exists a distinctive global minimum at $C = 11$.

Third, we built the antecedent of the TS model by projecting the rows of the partition matrix U corresponding to the minimum of the Xie–Beni index onto the input variables and approximating the projections by triangular membership functions. Fourth, we computed the consequent parameters of each rule as a local least squares estimate. Fifth, we applied the GA to optimize the TS model so as to reduce the MSE between the known dissimilarity values and the output of the TS model. Fig. 4 shows the MSE of the best chromosome of each generation versus the number of iterations for the same trial as in Fig. 3. We observe that the MSE gets stable around 4500 generations and reaches the termination threshold after 5550 iterations.

To assess the generalization properties, for each trial and each experiment we tested the TS model on all possible pairs of points in the data set and measured the percentage of the point pairs with dissimilarity degree lower than (higher than) 0.5 for pairs of points belonging (not belonging) to the same class. Table 1 shows the percentages of correct dissimilarity values obtained. Here, the columns show, respectively, the percentage of points composing the training pool, the number of rules of the TS model (in the form (mean ± standard deviation)), the percentage of correct dissimilarity values before and after the GA optimization. It can be observed that the application of the GA sensibly improves the percentage of correct dissimilarity values generated by the TS model independently of the cardinality of the training pool. Please, note that the percentage of total pairs of points included in the training set is much lower than the percentage of total points in the training pool. Taking this into account, the 90.4% achieved by the TS model using a training pool with 25% of the points is undoubtedly remarkable. Finally, we note that the number of rules is quite high, especially for lower percentages of points in the training pool. This implies a high number of GA parameters which may cause overfitting.

4.1.1. *Solving overfitting*

To highlight possible overfitting problems, at each iteration we also computed the MSE between the real dissimilarity values and the dissimilarity values output by the TS model on a validation set consisting of point pairs generated (in the same way as the training set) from 15% of the points not included in the training pool. Fig. 5 shows the MSE versus the number of iterations for the same trial as in Fig. 4. Actually, we recognize the presence of overfitting in correspondence with 1500–2000 generations. We experienced that this problem occurs for training pools consisting of up to 20% of the points. Thus, for these cases, we stopped the execution of the GA after 2000 generations. To verify whether overfitting was due to the inappropriate choice of the number of clusters determined by the Xie–Beni index, we performed a thorough experimental activity varying incrementally the number of clusters (starting from 2), executing the GA and measuring the overall performance of the TS model. We observed that the best results were achieved adopting the number of rules fixed by the Xie–Beni index.

Table 1
Percentage of point pairs with correct dissimilarity values (synthetic data set)

Training pool (%)	Number of rules	Correct dissimilarity values before GA (%)	Correct dissimilarity values after GA (%)
5	10.5 ± 3.3	61.8 ± 6.7	69.6 ± 7.6
10	10.1 ± 3.2	66.3 ± 3.8	75.7 ± 5.2
15	11.7 ± 3.2	65.6 ± 6.8	82.6 ± 4.2
20	12.6 ± 2.9	67.8 ± 3.0	85.3 ± 3.9
25	14.2 ± 1.5	69.7 ± 2.8	90.4 ± 3.5

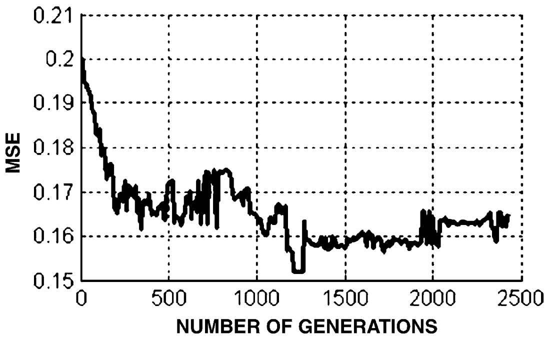

Fig. 5. Mean square error of the best chromosome on the validation set.

Table 2
Number of clusters in the five experiments

Training pool (%)	Number of clusters	Percentage of trials with number of clusters equal to number of classes (%)
5	2.1 ± 0.3	90
10	2.0 ± 0.0	100
15	2.0 ± 0.0	100
20	2.0 ± 0.0	100
25	2.0 ± 0.0	100

4.1.2. Applying ARCA

Finally, we computed the dissimilarity relation and applied ARCA. We used $\varepsilon = 0.001$ and $m = 2$. We executed the fuzzy relational algorithm with C ranging from 2 to 5 and chose the optimal number of clusters based on the Xie–Beni index. Table 2 shows the number of clusters (in the form (mean ± standard deviation)) in the five experiments. It can be observed that the percentage of trials in which the number of clusters is equal to the number of classes increases very quickly (up to 100%) with the increase of the percentage of points in the training pool.

Table 3 shows the percentage of correctly classified points in the five experiments when $C = 2$. Here, the second column indicates the percentage of correctly classified points and the third column the partition coefficient. As expected, the percentage of correctly classified points increases with the increase of points in the training pool. Just for small percentages of points in the training pool, the combination TS system–ARCA is able to trace the boundaries of the classes conveniently. The quality of the approximation improves when the points of the training pool are a significant sample of the overall data set. Table 3 shows that the class shape is almost correctly identified just with 5% of the points of the data set. Note that, as reported in Table 1, the TS system is able to output only 69.6% of correct dissimilarity values, when trained with training pools containing the same percentage of points. Finally, the high values of the partition coefficient highlight that the partition determined by the relational clustering algorithm is quite good.

Table 3
Percentage of correctly classified points of the synthetic data set in the five experiments

Training pool (%)	Correctly classified points (%)	Partition coefficient
5	84.4 ± 6.5	0.84 ± 0.07
10	87.5 ± 5.4	0.89 ± 0.05
15	93.7 ± 3.5	0.90 ± 0.04
20	94.1 ± 2.9	0.92 ± 0.02
25	97.0 ± 1.8	0.94 ± 0.03

4.1.3. Neural network approach vs TS model approach

To further assess the results described above, we applied ARCA to the dissimilarity relation extracted by the three-layer feed-forward neural network proposed in Refs. [15,16]. We performed the same experiments with 5%, 10%, 15%, 20% and 25% of the data set. We obtained 87.12% ± 2.82%, 88.85 ± 2.8%, 93.80% ± 1.54%, 94.60 ± 1.52% and 97.34% ± 1.34%, respectively, of correctly classified points. We can conclude that the two methods to extract the dissimilarity relation achieve similar performance. Unlike the neural network-based approach, however, the TS model-based approach allows describing the dissimilarity relation intuitively. To explain this statement, Figs. 6 and 7 show the antecedent and the consequent of the rules which compose the TS model for the same trial as in Fig. 3 before and after the GA optimization, respectively. Here, we have associated a label with each fuzzy set based on the position of the fuzzy set in the universe of definition.

Since each rule defines its fuzzy sets, which may be different from the other rules, we used the following method to assign a meaningful linguistic label to each fuzzy set. Firstly, we uniformly partition the universes of discourse into

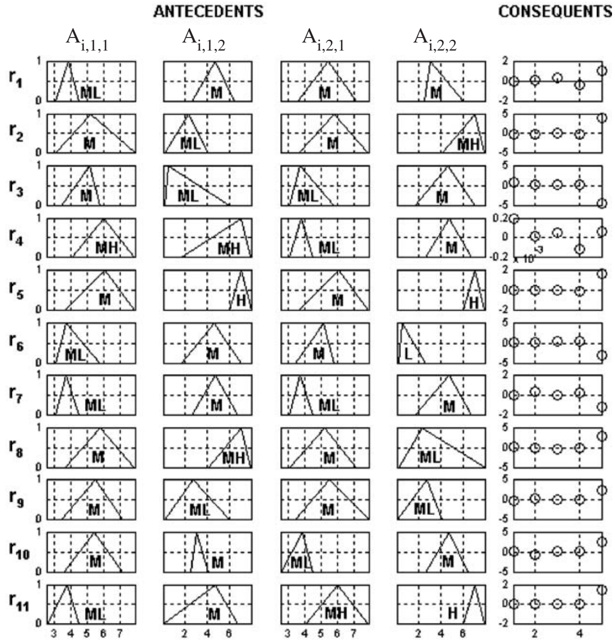

Fig. 6. Rules before GA (synthetic data set).

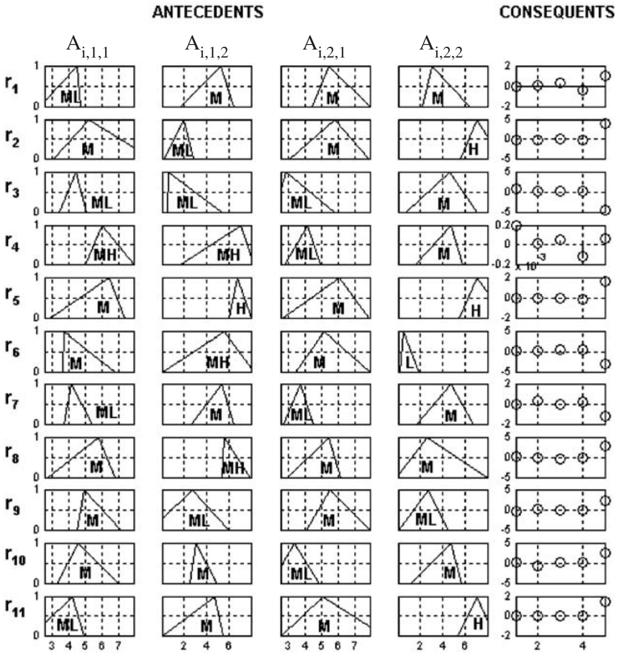

Fig. 7. Rules after GA (synthetic data set).

G fuzzy sets (denoted as *reference terms* in the following) as shown in Fig. 8 and associate a meaningful label with each fuzzy set. In the example, labels L, ML, M, MH, H, denote, respectively, low, medium-low, medium, medium-high, and high. Then, we compute the similarity between each fuzzy set used in the rules and the reference terms using

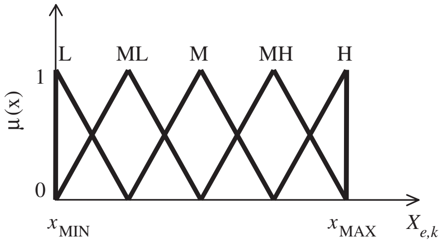

Fig. 8. Reference terms for a generic input variable $X_{e,k}$.

Table 4
The qualitative model before GA

Rule	$X_{1,1}$	$X_{1,2}$	$X_{2,1}$	$X_{2,2}$	$\bar{d}_{i,j}$
r_1	ML	M	M	M	0.80
r_2	M	ML	M	MH	0.41
r_3	M	ML	ML	M	0.50
r_4	MH	MH	ML	M	0.81
r_5	M	H	M	H	0.00
r_6	ML	M	M	L	0.36
r_7	ML	M	ML	M	0.23
r_8	M	MH	M	ML	0.61
r_9	M	ML	M	ML	0.40
r_{10}	M	M	ML	M	0.91
r_{11}	ML	M	MH	H	0.53

the formula:

$$S_{i,l} = \frac{|A_{i,e,k} \cap P_{l,e,k}|}{|A_{i,e,k} \cup P_{l,e,k}|},$$

where $A_{i,e,k}$ and $P_{l,e,k}$ are, respectively, a fuzzy set and a reference term defined on the domain of input $X_{e,k}$ [33]. Finally, if there exists a value of $S_{i,l}$, with $l = 1 \ldots G$, larger than a fixed threshold τ, the reference term $P_{l,e,k}$ is associated with $A_{i,e,k}$ (if there exist more $P_{l,e,k}$ with $S_{i,l} > \tau$, then $A_{i,e,k}$ is associated with the $P_{l,e,k}$ corresponding to the highest $S_{i,l}$); otherwise, $A_{i,e,k}$ is added to the reference terms after associating a meaningful label with it. Once the fuzzy sets of all the rules have been examined, we again compute the similarity between each fuzzy set and the current reference terms in order to associate the most appropriate label with each fuzzy set. To generate the labels associated with the fuzzy sets shown in Figs. 6 and 7, we have used a threshold $\tau = 0.5$. Note that no further reference term has been added.

To interpret the rules, we follow this procedure: for each pattern $\underline{z}_k = [\underline{x}_i, \underline{x}_j, d_{i,j}]$ in the training set, we feed as input the values of the coordinates of $\underline{x}_i$ and $\underline{x}_j$ to the TS model and measure the activation degree of each rule. We aim to discover whether there exists a relation between the activation of a rule and the values of dissimilarity. Tables 4 and 5 show, for each rule, the mean value $\bar{d}_{i,j}$ of dissimilarity $d_{i,j}$ of the pairs $(\underline{x}_i, \underline{x}_j)$ of patterns of the training set which activate this rule more than the other rules before and after applying the GA, respectively. This association between

Table 5
The qualitative model after GA

Rule	$X_{1,1}$	$X_{1,2}$	$X_{2,1}$	$X_{2,2}$	$\bar{d}_{i,j}$
r_1	ML	M	M	M	0.92
r_2	M	ML	M	H	0.73
r_3	ML	ML	ML	M	0.00
r_4	MH	MH	ML	M	1.00
r_5	M	H	M	H	0.00
r_6	M	MH	M	L	0.70
r_7	ML	M	ML	M	0.00
r_8	M	MH	M	M	0.73
r_9	M	ML	M	ML	0.32
r_{10}	M	M	ML	M	0.50
r_{11}	ML	M	M	H	0.38

rules and dissimilarity values helps us interpret the meaning of the rules.

Before applying the GA, we can observe that each rule is often activated both by pairs of points with high dissimilarity and by pairs of points with low dissimilarity. Indeed, the mean value of dissimilarity is quite close to 0.5 rather than to 0 or 1. This means that the antecedents of the rules determine regions of the plane which contain points belonging both to the same class and to different classes. This observation confirms the results shown in Table 1: using 15% of points in the training pool, we achieved only 65.6% of correct classification. To make the relations expressed by the rules more evident, we introduce a visual representation of the rules. To this aim, we assume that a rule is activated more than the other rules when the membership of a pair of points to the region determined by the antecedent of the rule is higher than 0.5. The sub-region which contains these pairs of points corresponds to the region determined by the rectangles produced by α-cutting the membership functions with $\alpha = 0.5$. As an example, Fig. 9 shows these rectangles for rule r_3. For the sake of preciseness, here we use the real fuzzy sets rather than their linguistic approximation through the reference terms. The circled points of the data set represent the points belonging to the training pool. We can observe that the rectangles contain points which belong both to the same class and to different classes. This result could be expected as the mean value $\bar{d}_{i,j}$ of dissimilarity associated with this rule in Table 4 is 0.5.

After applying the GA, we note that rules are activated by pairs of points with either high or low dissimilarity. Indeed, the mean value of dissimilarity is close to 0 or 1. This means that the antecedents of the rules determine regions of the plane which contain points belonging either to the same class or to different classes. This observation confirms the results shown in Table 1: using 15% of points in the training pool, we achieved 82.6% of correct classification. Fig. 10 shows the graphical representation of rule r_3 after the optimization performed by the GA. We can observe that the rectangles contain only points which belong to the same class. This result agrees with the mean

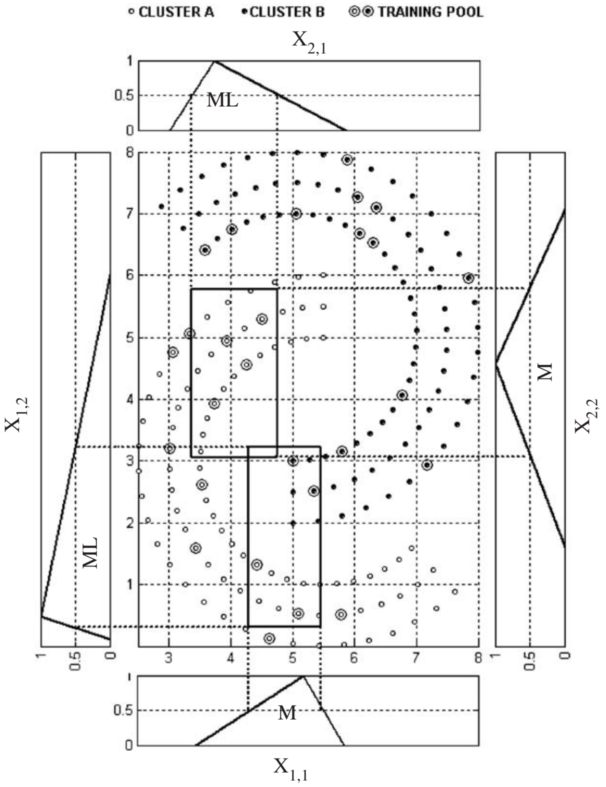

Fig. 9. Rule r_3 before GA.

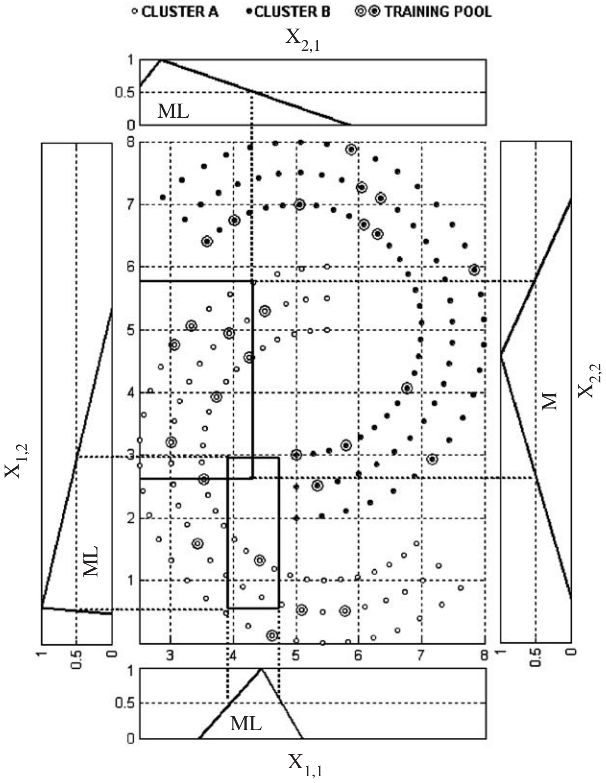

Fig. 10. Rule r_3 after GA.

Table 6
Percentage of pattern pairs with correct dissimilarity values (Iris data set)

Training pool (%)	Number of rules	Correct dissimilarity values before GA (%)	Correct dissimilarity values after GA (%)
5	8.9 ± 2.4	80.0 ± 4.1	80.5 ± 4.5
10	6.4 ± 1.6	82.7 ± 4.8	87.7 ± 3.1
15	4.8 ± 0.6	80.8 ± 3.0	90.2 ± 2.2
20	4.4 ± 0.8	78.5 ± 7.0	91.6 ± 2.0
25	4.7 ± 0.5	80.7 ± 4.7	91.6 ± 1.8

value $\bar{d}_{i,j} = 0$ of dissimilarity associated with this rule in Table 5.

The analysis of the rules, which compose the TS system, can therefore provide an intuitive explanation of the dissimilarity relation, by highlighting interesting associations between the features which describe the patterns and the values of dissimilarity. Actually, using the antecedents of Tables 4 and 5, the plane can be subdivided into regions characterized by different levels of dissimilarity. This characteristic may be considered a considerable advantage with respect to the neural approach proposed in Ref. [16].

4.2. The Iris data set

The second example uses the real Iris data set, provided by the University of California, Irvine (http://www.ics.uci.edu/∼mlearn/MLSummary.html). Iris consists of 150 patterns characterized by four numeric features which describe, respectively, sepal length, sepal width, petal length and petal width. Patterns are equally distributed in three classes of Iris plants, namely Iris Setosa, Iris Versicolor and Iris Virginica. Class Iris Setosa is linearly separable from the other two. However, class Iris Versicolor and Iris Virginica are not separable from each other. We carried out the five experiments described in Section 4.1. Tables 6–8 show the percentage of pattern pairs with correct dissimilarity values, the number of clusters determined by the Xie–Beni index, and the percentage of correctly classified points of the Iris data set in the five experiments, respectively. We observe that the percentage of correct dissimilarity values (after GA) is higher than 90% with just 15% of points in the training pool. Further, in 100% of the trials the number of clusters is equal to the number of classes when the training pool contains 20% of the points. Finally, just with 5% of points in the training pool, the combination TS system–ARCA is able to correctly classify nearly 90% of the points.

Fig. 11 shows the antecedent and the consequent of the rules which compose the TS model for one of the trials performed in the experiments after the optimization process performed by the GA (for the sake of brevity, we omit the representation of the rules before applying the GA). The labels are associated with each fuzzy set using the same reference terms as in Fig. 8 and the same procedure described in Section 4.1.

Table 7
Number of clusters in the five experiments (Iris data set)

Training pool (%)	Number of clusters	Percentage of trials with number of clusters equal to number of classes (%)
5	2.5 ± 0.5	50
10	2.8 ± 0.6	60
15	3.3 ± 0.5	70
20	2.9 ± 0.3	100
25	3.0 ± 0.0	100

Table 8
Percentage of correctly classified points of the Iris data set in the five experiments

Training pool (%)	Correctly classified points (%)	Partition coefficient
5	89.8 ± 5.5	0.74 ± 0.08
10	92.5 ± 4.6	0.86 ± 0.04
15	94.4 ± 3.0	0.91 ± 0.04
20	95.2 ± 2.1	0.93 ± 0.04
25	95.8 ± 1.6	0.92 ± 0.03

Table 9 shows, for each rule, the mean value $\bar{d}_{i,j}$ of dissimilarity $d_{i,j}$ of the pairs $(\underline{x}_i, \underline{x}_j)$ of patterns of the training set which activate this rule more than the other rules after applying the GA.

To allow the reader to verify whether the relations expressed by the rules in Table 9 correspond to reality, Figs. 12a–d show the distribution of the patterns for each feature. In the figures, the X and Y axes represent the patterns (separated for classes) and the feature values, respectively. For the reader's convenience, on the left side of the figures we show the reference terms which partition the feature domain. As an example, let us consider rule r_1. Here, the reader can observe that the fuzzy sets defined on variables X_1 and X_2 define clusters which contain patterns belonging to Setosa, and Versicolor or Virginica irises, respectively. Indeed, rule r_1 is associated with a mean value $\bar{d}_{i,j}$ equal to 1.

4.3. The WBC data set

As third example, we used the WBC data set, also provided by the University of California. The WBC data set consists of 699 patterns belonging to two classes: 458 patterns are members of the "benign" class and the other 241 patterns are members of the "malignant" class. Each pattern is described by nine features: clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli and mitoses. Since 16 patterns have a missing value, we decided to use only 683 patterns in our experiments. We performed the five experiments described in Section 4.1.

Tables 10–12 show the percentage of pattern pairs with correct dissimilarity values, the number of clusters

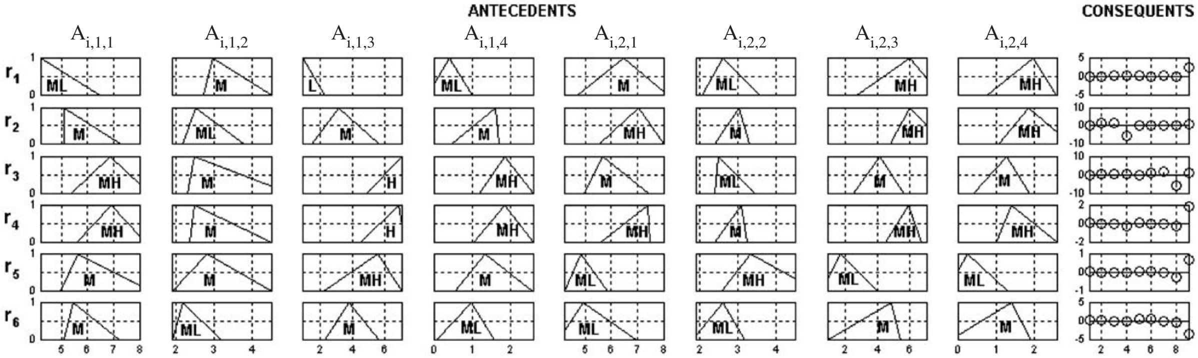

Fig. 11. Rules after GA (Iris data set).

Table 9
The qualitative model after GA

Rule	$X_{1,1}$	$X_{1,2}$	$X_{1,3}$	$X_{1,4}$	$X_{2,1}$	$X_{2,2}$	$X_{2,3}$	$X_{2,4}$	$\bar{d}_{i,j}$
r_1	ML	M	L	ML	M	ML	MH	MH	1.00
r_2	M	ML	M	M	MH	M	MH	MH	1.00
r_3	MH	M	H	MH	M	ML	M	M	1.00
r_4	MH	M	H	MH	MH	M	MH	MH	0.16
r_5	M	M	MH	M	ML	MH	ML	ML	1.00
r_6	M	ML	M	ML	ML	ML	M	M	0.22

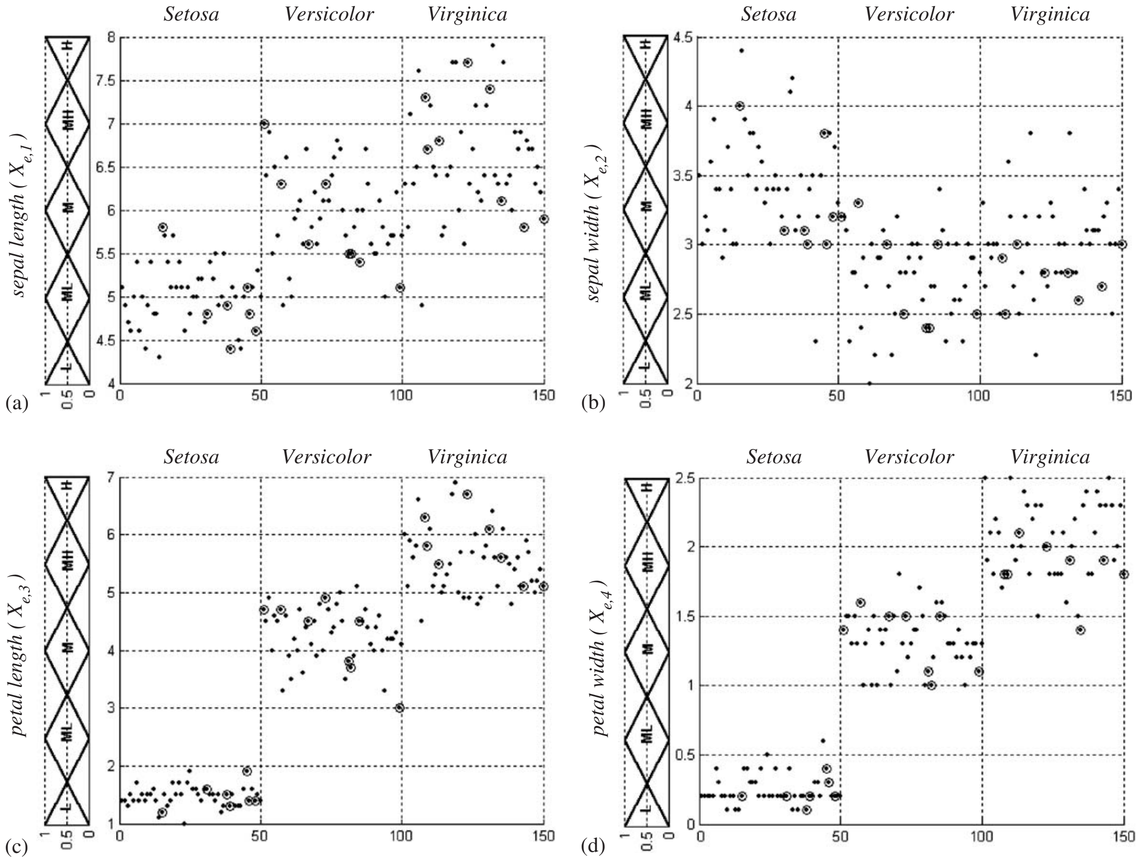

Fig. 12. Distribution of the patterns for each feature (Iris data set).

Table 10
Percentage of pattern pairs with correct dissimilarity values (WBC data set)

Training pool (%)	Number of rules	Correct dissimilarity values before GA (%)	Correct dissimilarity values after GA (%)
5	5.3 ± 0.6	83.0 ± 7.8	86.8 ± 4.8
10	4.0 ± 0.0	85.5 ± 8.0	91.4 ± 0.8
15	3.7 ± 0.6	87.3 ± 3.0	93.2 ± 1.0
20	3.3 ± 0.6	83.0 ± 2.3	92.3 ± 0.4
25	3.7 ± 0.6	84.2 ± 5.3	92.8 ± 0.6

Table 11
Number of clusters in the five experiments (WBC data set)

Training pool (%)	Number of clusters	Percentage of trials with number of clusters equal to number of classes (%)
5	2.0 ± 0.0	100
10	2.0 ± 0.0	100
15	2.0 ± 0.0	100
20	2.0 ± 0.0	100
25	2.0 ± 0.0	100

Table 12
Percentage of correctly classified points of the WBC data set in the five experiments

Training pool (%)	Correctly classified points (%)	Partition coefficient
5	95.9 ± 0.5	0.94 ± 0.03
10	96.1 ± 0.3	0.97 ± 0.00
15	96.8 ± 0.7	0.96 ± 0.00
20	96.8 ± 0.3	0.95 ± 0.01
25	97.1 ± 0.1	0.96 ± 0.01

determined by the Xie–Beni index, and the percentage of correctly classified points of the WBC data set in the five experiments, respectively. We observe that the percentage of correct dissimilarity values is higher than 90% with just 10% of points in the training pool. Further, the number of clusters is always equal to the number of classes. Finally, just with 5% of points in the training pool, the combination TS system–ARCA is able to correctly classify about 96% of the points. This result is very interesting when compared with some results published in the recent literature. In Ref. [34], for instance, seven different classification methods are compared with each other using the WBC data set: half of the 683 patterns are used as training set, and the remaining patterns are used as test set. The best method achieves 95.14% classification rate on the test set. In our case, though we exploit a lower number of patterns to generate the training set, we have 97.1% classification rate. This result proves the effectiveness of our approach.

Fig. 13 shows the antecedent and the consequent of the rules which compose the TS model for one of the trials performed in the experiments after the optimization process performed by the GA. The labels are associated with each fuzzy set using the same reference terms as in Fig. 8 and the same procedure described in Section 4.1.

Table 13 shows, for each rule, the mean value $\bar{d}_{i,j}$ of dissimilarity $d_{i,j}$ of the pairs $(\underline{x}_i, \underline{x}_j)$ of patterns of the training set which activate this rule more than the other rules after applying the GA. Here, from the first rule we can observe that if all the values of the features are low, then the patterns are similar. On the other hand, we can derive from the second rule that if the first input has low values for all the features and the second input has high values for all the features, then the patterns belong to different classes. The third rule shows that if the first input has high values for all the features and the second input has low values, then the dissimilarity is quite "undecided". To allow the reader to verify whether the relations expressed by the rules in Table 13 correspond to reality, Figs. 14a–i show the distribution of the patterns for each feature. The reader can observe, for instance, that the majority of patterns with low values for each feature belong to the same class, the benign class, as it was intuitively described by rule r_1. Similarly, the reader can observe that patterns with high values for each feature and patterns with low values for each feature belong to different classes, as it was represented by rule r_2.

5. Conclusions

Most clustering algorithms partition a data set based on a dissimilarity relation expressed in terms of some distance. If the nature of the relation is conceptual rather than metric, distance functions may fail to correctly model dissimilarity and consequently cluster shapes. For this reason, in this paper, we have proposed to extract the dissimilarity relation directly from the data. To this aim, we exploit a TS fuzzy

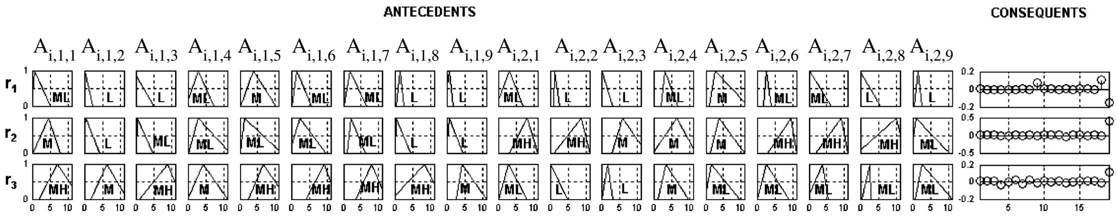

Fig. 13. Rules after GA (WBC data set).

Table 13
The qualitative model after GA

Rule	$X_{1,1}$	$X_{1,2}$	$X_{1,3}$	$X_{1,4}$	$X_{1,5}$	$X_{1,6}$	$X_{1,7}$	$X_{1,8}$	$X_{1,9}$	$X_{2,1}$	$X_{2,2}$	$X_{2,3}$	$X_{2,4}$	$X_{2,5}$	$X_{2,6}$	$X_{2,7}$	$X_{2,8}$	$X_{2,9}$	$\bar{d}_{i,j}$
r_1	ML	L	L	ML	M	ML	ML	L	L	ML	L	L	ML	M	ML	ML	L	L	0.00
r_2	M	L	ML	ML	ML	ML	ML	L	L	MH	MH	M	M	M	MH	MH	MH	ML	0.92
r_3	MH	M	MH	M	MH	MH	MH	MH	M	ML	L	L	M	ML	ML	ML	ML	ML	0.39

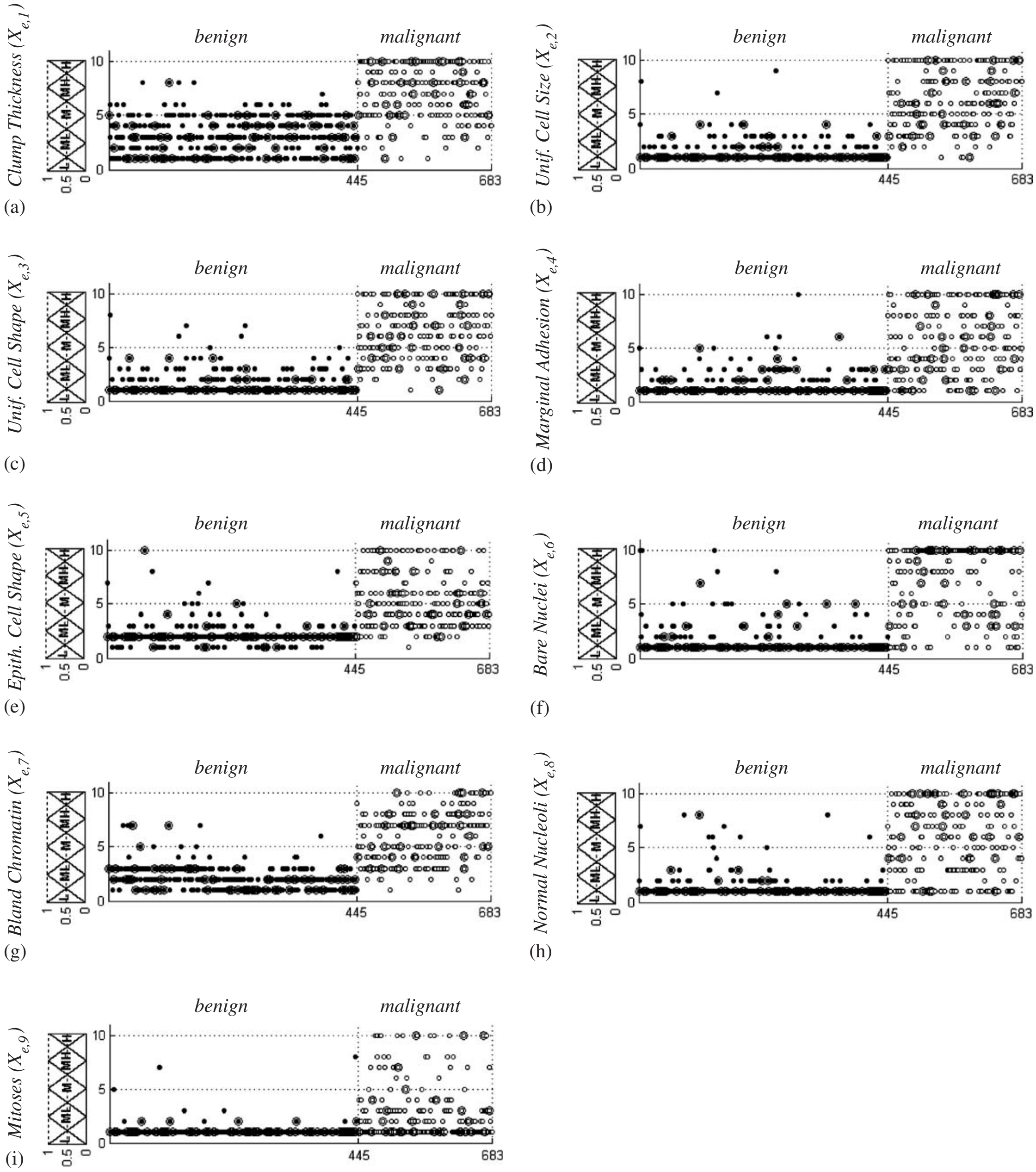

Fig. 14. Distribution of the patterns for each feature (WBC data set).

system appropriately trained with a few known dissimilarities between pattern pairs. The dissimilarity relation modelled by the TS system is fed as input to a fuzzy relational clustering algorithm recently proposed by the authors. The algorithm partitions the data set based on the proximity of the vectors containing the dissimilarity values between each pattern and all the other patterns in the data set.

We have described the application of our method to an artificial data set, which is not easily clustered by classical fuzzy clustering algorithms, and to two well-known real data sets, namely Iris and WBC data sets. Further, we have shown that just using a significantly low percentage of known dissimilarities, our method is able to cluster the data sets almost correctly. Finally, we have described how the TS model can provide an intuitive linguistic description of the dissimilarity relation.

Concluding, we wish to point out that in order to asses the effectiveness of ARCA, we performed several experiments by using the well-known relational clustering algorithms AP, RFCM, NERFCM, FCMdd and FRC to cluster the dissimilarity relations extracted by the TS model from the three data sets described in Section 4. We observed that the behavior of these algorithms is similar to ARCA's for the first two data sets, where the number of objects is quite small. Actually, both AP and FRC are very sensitive to the initialization phase. For the third data set, though we used values of m quite low (1.2–1.4), RFCM, NERFCM and FCMdd tend to converge to partitions with membership of $1/C$ for all the objects. AP and FRC show a stronger dependence on the initialization.

References

[1] A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review, ACM Comput. Surv. 31 (3) (1999) 265–323.

[2] M.C. Su, C.H. Chou, A modified version of the K-means algorithm with a distance based on cluster symmetry, IEEE Trans. Pattern Anal. Mach. Intell. 23 (6) (2001) 674–680.

[3] A.J. Jain, P.J. Flynn (Eds.), Three Dimensional Object Recognition Systems, Elsevier Science Inc., New York, NY, 1993.

[4] B. Kamgar-Parsi, A.K. Jain, Automatic aircraft recognition: toward using human similarity measure in a recognition system, Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognition (1999) 268–273.

[5] L.J. Latecki, R. Lakamper, Shape similarity measure based on correspondence of visual parts, IEEE Trans. Pattern Anal. Mach. Intell. 22 (10) (2000) 1185–1190.

[6] S. Santini, R. Jain, Similarity measures, IEEE Trans. Pattern Anal. Mach. Intell. 21 (9) (1999) 871–883.

[7] D. Valentin, H. Abdi, A.J. O'Toole, G.W. Cottrell, Connectionist models of face processing: a survey, Pattern Recognition 27 (1994) 1208–1230.

[8] R.A. Jarvis, E.A. Patrick, Clustering using a similarity method based on shared near neighbors, IEEE Trans. Comput. C-22 (11) (1973) 1025–1034.

[9] R. Michalski, R.E. Stepp, E. Diday, Automated construction of classifications: conceptual clustering versus numerical taxonomy, IEEE Trans. Pattern Anal. Mach. Intell. 5 (4) (1983) 396–409.

[10] M.S. Yang, K.L. Wu, A similarity-based robust clustering method, IEEE Trans. Pattern Anal. Mach. Intell. 26 (4) (2004) 434–448.

[11] D.W. Jacobs, D. Weinshall, Y. Gdalyahu, Classification with nonmetric distances: image retrieval and class representation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (6) (2000) 583–600.

[12] S. Makrogiannis, G. Economou, S. Fotopoulos, A region dissimilarity relation that combines feature-space and spatial information for color image segmentation, IEEE Trans. Systems Man Cybernet. Part B 35 (1) (2005) 44–53.

[13] W. Pedrycz, G. Succi, M. Reformat, P. Musilek, X. Bai, Expressing similarity in software engineering: a neural model, Proceedings of the Second International Workshop on Soft Computing Applied to Software Engineering, Enschede, The Netherlands, 2001.

[14] T. Hertz, A. Bar-Hillel, D. Weinshall, Learning distance functions for image retrieval, Proceedings of the 2004 IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognition II (2004) 570–577.

[15] P. Corsini, B. Lazzerini, F. Marcelloni, Clustering based on a dissimilarity measure derived from data, Proceedings of KES 2002, Crema, Italy, 2002, pp. 885–889.

[16] P. Corsini, B. Lazzerini, F. Marcelloni, A fuzzy relational clustering algorithm based on a dissimilarity measure extracted from data, IEEE Trans. Systems Man and Cybernet. Part B 34 (1) (2004) 775–782.

[17] J.C. Bezdek, J. Keller, R. Krisnapuram, N.R. Pal, Fuzzy Model and Algorithms for Pattern Recognition and Image Processing, Kluwer Academic Publishing, Boston, 1999.

[18] R. Krishnapuram, A. Joshi, O. Nasraoui, L. Yi, Low-complexity fuzzy relational clustering algorithms for web mining, IEEE Trans. Fuzzy Systems 9 (4) (2001) 595–607.

[19] T. Takagi, M. Sugeno, Fuzzy identification of systems and its application to modeling and control, IEEE Trans. Systems Man Cybernet. 15 (1985) 116–132.

[20] M. Setnes, H. Roubos, GA-fuzzy modeling and classification: complexity and performance, IEEE Trans. Fuzzy Systems 8 (5) (2000) 509–522.

[21] P. Corsini, B. Lazzerini, F. Marcelloni, A new fuzzy relational clustering algorithm based on the fuzzy C-means algorithm, Soft Comput. 9 (6) (2005) 439–447.

[22] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, New York, 1981.

[23] X.L. Xie, G. Beni, A validity measure for fuzzy clustering, IEEE Trans. Pattern Anal. Mach. Intell. 13 (8) (1991) 841–847.

[24] M. Roubens, Pattern classification problems and fuzzy sets, Fuzzy Sets and Systems 1 (1978) 239–253.

[25] M.P. Windham, Numerical classification of proximity data with assignment measures, J. Classification 2 (1985) 157–172.

[26] R.J. Hathaway, J.W. Davenport, J.C. Bezdek, Relational duals of the c-means clustering algorithms, Pattern Recognition 22 (1989) 205–212.

[27] R.J. Hathaway, J.C. Bezdek, NERF c-means: non-Euclidean relational fuzzy clustering, Pattern Recognition 27 (1994) 429–437.

[28] L. Kaufman, P.J. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis, Wiley, New York, 1990.

[29] R.N. Davé, S. Sen, Robust fuzzy clustering of relational data, IEEE Trans. Fuzzy Systems 10 (6) (2002) 713–727.

[30] J.F. Kolen, T. Hutcheson, Reducing the time complexity of the fuzzy C-means algorithm, IEEE Trans. Fuzzy Systems 10 (2) (2002) 263–267.

[31] D.E. Gustafson, W.C. Kessel, Fuzzy clustering with fuzzy covariance matrix, in: M.M. Gupta, R.K. Ragade, R. R Yager (Eds.), Advances in Fuzzy Set Theory and Applications, North-Holland, Amsterdam, 1979, pp. 605–620.

[32] D. Chaudhuri, B.B. Chaudhuri, A novel multiseed nonhierarchical data clustering technique, IEEE Trans. Systems Man Cybernet. Part B 27 (5) (1997) 871–876.

[33] M. Sugeno, T. Yasukawa, A fuzzy logic-based approach to qualitative modelling, IEEE Trans. Fuzzy Systems 1 (1) (1993) 7–31.

[34] H.-M. Lee, C.-M. Chen, J.-M. Chen, Y.-L. Jou, An efficient fuzzy classifier with feature selection based on fuzzy entropy, IEEE Trans. Systems Man Cybernet. Part B Cybernet. 31 (3) (2001) 426–432.

About the Author—MARIO G.C.A. CIMINO received the Laurea degree in Computer Engineering in March 2003 from the University of Pisa, where he has been a Ph.D. student in Information Engineering since January 2004. His main research interests include relational clustering, robust clustering, Information Systems.

About the Author—BEATRICE LAZZERINI is a Full Professor at the Faculty of Engineering of the University of Pisa, Italy. Her main research interests include fuzzy systems, neural networks and evolutionary computation. She has co-authored seven books and has published more than one hundred papers in international journals and conferences.

About the Author—FRANCESCO MARCELLONI is an Associate Professor at the Faculty of Engineering of the University of Pisa. His research interests include object-oriented software development process, object-oriented models, approximate reasoning, fuzzy rule-based systems, fuzzy clustering algorithms and pattern recognition. He is (co-)author of more than 80 papers in international journals and conferences.