Relational Clustering based on a Dissimilarity Relation Extracted from Data by a TS Model*

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
{b.lazzerini,f.marcelloni}@iet.unipi.it

Abstract - *Most clustering algorithms partition a data set based on a dissimilarity relation expressed in terms of some distance function. When the nature of this relation is conceptual rather than metric, distance functions may fail to adequately model dissimilarity. For this reason, we propose to extract dissimilarity relations directly from the data. We exploit some pairs of patterns with known dissimilarity to build a TS fuzzy system which models the dissimilarity relation. Then, we use the TS system to compute a dissimilarity relation between any pair of patterns. The resulting dissimilarity matrix is input to a new unsupervised fuzzy relational clustering algorithm, which partitions the data set based on the proximity of the vectors containing the dissimilarity values between a pattern and all the patterns in the data set. Experimental results to confirm the validity of our approach are shown and discussed.*

Keywords: Similarity/Dissimilarity relations, Fuzzy identification, Fuzzy clustering.

1 Introduction

In the last years, several different clustering algorithms have been proposed to partition a data set into groups of similar objects. Similarity (more often dissimilarity) is typically expressed in terms of some distance function (such as the Euclidean distance or the Mahalanobis distance). In real applications, however, when data distribution is not regular, distance functions cannot adequately model dissimilarity, which appears to be conceptual rather than metric [6][9][10][13]. Consider, for example, the pixels of an image made up of distinguishable elements with irregular-shaped contours. The dissimilarity between pixels should be small (large) when the pixels belong to the same image element (different image elements). In these (frequent) situations, a relation extracted directly from the data rather than a data-independent relation (such as a distance function) may be more effective in modeling dissimilarity.

To this aim, in previous papers, we used a multilayer perceptron with supervised learning to extract the dissimilarity relation from the data exploiting a few pairs of data with known dissimilarity [3][4][5]. Then, we used the dissimilarity measure generated by the network to guide an unsupervised fuzzy relational clustering algorithm. We showed that the clustering algorithm based on the neural dissimilarity outperforms some widely used (possibly partially supervised) clustering algorithms based on spatial dissimilarity. Following this approach, in this paper, we exploit a Takagi-Sugeno (TS) fuzzy rule system to extract the dissimilarity relation from the data [12]. Fuzzy rules composing the TS system are identified from a few pairs of patterns with known dissimilarity by using the method proposed in [11]. First, we apply the classical fuzzy C-means (FCM) algorithm [1] to determine the TS system structure (i.e., number of rules and number of fuzzy sets which partition each input variable). Then, we estimate the parameters which identify the consequent functions. Finally, we apply a genetic algorithm to refine the TS model so as to reduce the model error. Appropriate constraints are enforced during the genetic evolution to preserve the properties of the TS model derived from the clusters produced by the FCM algorithm. At the end of the identification phase, the TS system can associate a dissimilarity degree with each pair of patterns in the data set so as to generate a dissimilarity relation between patterns in the data set. Compared to the neural network-based approach, the TS model-based approach shows similar performance, but better capability of describing the dissimilarity relation (through the rules of the TS model).

Due to the generalization performed starting from a restricted number of known relationship values, the dissimilarity relation D produced by the TS model may be neither irreflexive nor symmetric. Unfortunately, the most popular examples of fuzzy relational clustering algorithms [2] [8], such as the fuzzy nonmetric model, the assignment prototype model, the relational fuzzy C-means, and fuzzy C-medoids, assume that D is at least a positive, irreflexive and symmetric square binary relation. This makes these fuzzy relational clustering algorithms not applicable to our relation. Actually, these algorithms can be applied, but their convergence to a reasonable partition is not guaranteed. To make our approach independent of the characteristics of the relation generated by the TS model, we propose a new fuzzy relational clustering method which can be applied to any type of relation matrix. The algorithm exploits the well-known fuzzy C-means algorithm to partition the data set based on the proximity of the vectors containing the dissimilarity values between a pattern and all the patterns in the data set. We verified that our algorithm 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 our algorithm is based on the fuzzy C-means algorithm [1], which has proved to be one of the most stable fuzzy clustering algorithms, our algorithm is appreciably more stable than the other fuzzy relational clustering algorithms.

To test the effectiveness of our approach we present an example of its application to a synthetic data set and to a public real data set. We show how our relational clustering algorithm, which exploits the dissimilarity relation extracted using a limited number of training samples, achieves very good clustering performance.

2 Dissimilarity modeling

Let $Q=[\underline{x}_1,...,\underline{x}_M]$ be the data set. To model the dissimilarity relation, we use a fuzzy system composed of a set of rules expressed in Takagi-Sugeno form [12]:

r_i: **If** $X_{1,1}$ **is** $A_{i,1,1}$ **and** … $X_{1,F}$ **is** $A_{i,1,F}$ **and** $X_{2,1}$ **is** $A_{i,2,1}$ **and** … $X_{2,F}$ **is** $A_{i,2,F}$

then $d_i=\underline{a}_{i,1}^T\underline{X}_1+\underline{a}_{i,2}^T\underline{X}_2+b_i \quad i=1..C$

where $\underline{X}_e=[X_{e,1},...,X_{e,F}]$, with e=1,2, are the two input variables of F components which represent the pair of patterns whose dissimilarity has to be evaluated, $A_{i,e,1}$, ..., $A_{i,e,F}$ are fuzzy sets defined on the domain of $X_{e,1}$, ..., $X_{e,F}$, respectively, and $\underline{a}_{i,e}^T=[a_{i,e,1},...,a_{i,e,F}]$, with $a_{i,e,f}\in\Re$. 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:

$$d=\frac{\sum_{i=1}^{C}\beta_i d_i}{\sum_{i=1}^{C}\beta_i} \qquad (1)$$

where $\beta_i=\prod_{e=1}^{2}\prod_{f=1}^{F}A_{i,e,f}(\underline{x}_j)$ is the degree of activation of the i-th 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 [11]. Let $T=\{T_1,...,T_N\}$ be the set of known data, where $T_i=[\underline{x}_i,\underline{x}_j,d_{i,j}]\in\Re^{2F+1}$, with $d_{i,j}$ the known dissimilarity between $\underline{x}_i$ and $\underline{x}_j$. First, the fuzzy c-means algorithm (FCM) is applied to T to determine a partition U of the input/output space [1]. The optimal number of clusters is determined 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 [14]. 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 f^{th} component of the input variable $\underline{X}_e$ and approximating the projections by triangular membership functions. Once the antecedent membership functions have been fixed, the consequent parameters $[\underline{a}_{i,1},\underline{a}_{i,2},b_i]$, i=1..C, of each individual rule i are obtained as a local least squares estimate.

The strategy used so far to build the TS model has 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 which characterize 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. 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 between the predicted output and the desired output over the training set.

We start with an initial population composed of 80 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: only 25% of the new population is composed of offspring, whereas 75% 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% 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 optimised, we compute the dissimilarity value between each possible pair $(\underline{x}_i, \underline{x}_j)$ of patterns in 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 $\underline{x}_j$.

3 Relational clustering algorithm

To partition a set of patterns described by their reciprocal relationships, several fuzzy relational clustering algorithms have been introduced. The most popular examples are the fuzzy nonmetric model (FNM), the assignment prototype model (AP), the relational fuzzy C-means (RFCM), the non-Euclidean RFCM (NERFCM) [2] and the fuzzy C-medoids (FCMdd) [8]. All these algorithms assume that, at least, $D = [d_{i,j}]$ is a positive, irreflexive and symmetric fuzzy square binary dissimilarity relation, i.e., $\forall i, j \in [1..M]$, $d_{i,j} \geq 0$, $d_{i,i} = 0$ 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, these algorithms can be applied, but their convergence to a reasonable partition is not guaranteed.

To make our approach independent of the relation generated by the TS model, we propose a new fuzzy relational clustering method which can be applied to any type of relation matrix. The basic idea of our method 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 $\hat{\underline{x}}_i = [d_{i,1}, \ldots, d_{i,M}]$ 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 [2], we can obtain clusters by grouping patterns based on their closeness in the space $\Re^M$. Representing the $M \times M$ relation matrix as M vectors defined in the feature space $\Re^M$ allows us to transform a relational clustering problem into an object clustering problem. In this way, we can use the most popular and stable fuzzy object clustering algorithm, namely, the FCM algorithm [1].

Let $B_1, \ldots, B_C$ be a family of fuzzy clusters on Q. Then, the objective function minimised by the FCM algorithm is, in our case, $J_m(U,V) = \sum_{i=1}^{C}\sum_{k=1}^{M} u_{i,k}^m d^2(\hat{\underline{x}}_k, \hat{\underline{v}}_i)$, where m is the fuzzificazion constant, $U = [u_{i,k}]$ is a real $C \times M$ partition matrix, $u_{i,k}$ is the membership value of $\underline{x}_k$ to 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.

We tested our method on some public data sets and verified that the partitions obtained by our method are comparable to the ones generated by RFCM or NERFCM, when applicable. Further, we note that our approach 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. Further, we implemented the FCM algorithm in an efficient way in terms of both memory requirement and computation time, thanks to the use of the technique described in [7].

4 Experimental results

To assess the validity of our approach, we used the synthetic data set shown in Figure 1 and the Iris data set. For each data set, we repeated 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. From the training pool, we selected the pairs used to build the training set. 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 C pairs $(\underline{x}_i, \underline{x}_j)$ by randomly selecting C patterns $\underline{x}_j$ of the training pool as follows: one pattern is chosen among those with dissimilarity degree lower than 0.5 with $\underline{x}_i$, and the remaining C-1 patterns are chosen among those with dissimilarity degree higher than 0.5. 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.

4.1 Synthetic data set

Figure 1 shows the synthetic data set, which is composed of two classes. As it can be noted by analyzing the distribution of the points, object clustering algorithms based on the Euclidean or Mahalanobis distances cannot identify the classes correctly. We carried out the five 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 the fuzzification constant 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. Figure 2 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 mean square error between the known dissimilarity values and the output of the TS model.

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 I shows the percentages of correct dissimilarity values obtained. Here, the first column indicates the percentage of points composing the training pool, the second column shows the number of rules of the TS model (in the form (mean ± standard deviation)), the third and fourth columns present the percentage of correct dissimilarity values before and after the optimization process performed by the GA. 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 87.2% achieved by the TS model using a training pool with 25% of the points is undoubtedly remarkable.

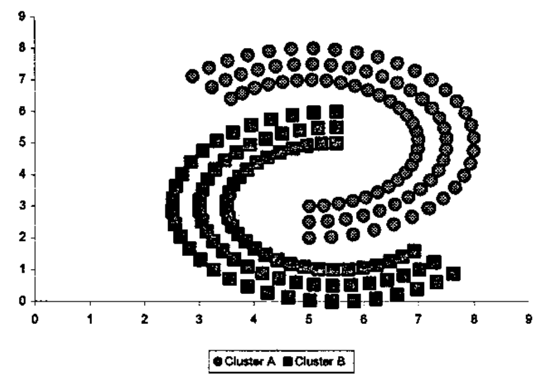

Figure 1. The synthetic data set

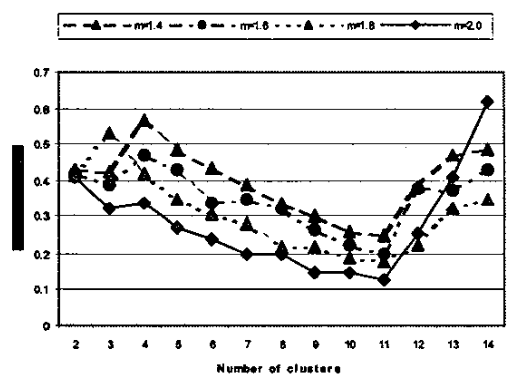

Figure 2. The Xie-Beni index versus C

Finally, we computed the dissimilarity relation and applied the new fuzzy relational algorithm. We used $\varepsilon = 0.001$ and m=2. As our relational algorithm is in fact based on an object clustering algorithm, to determine the number of clusters we can adopt the same indexes as in object clustering algorithms. Thus, we decided to use again the Xie-Beni index. We executed the fuzzy relational algorithm with C ranging from 2 to 8 and plotted the Xie-Beni index versus C. We chose, as optimal number of clusters, the value of C corresponding to the first distinctive local minimum. Table II 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 (up to 100%) with the increase of the percentage of points in the training pool.

Table I. Percentage of point pairs with correct dissimilarity values

Training pool	Number of rules	Correct dissimilarity values before GA	Correct dissimilarity values after GA
5%	9.2 ± 4.5	61.2% ± 4.6%	67.4% ± 5.2%
10%	11.0 ± 3.7	63.9% ± 4.8%	73.1% ± 5.5%
15%	10.8 ± 4.1	66.5% ± 6.3%	79.1% ± 5.5%
20%	12.4 ± 3.1	64.8% ± 3.2%	80.8% ± 4.9%
25%	12.5 ± 2.6	63.7% ± 5.4%	87.2% ± 3.6%

Table III shows the percentage of correctly classified points of the synthetic data set 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 – relational clustering algorithm 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 III shows that the class shape is almost correctly identified with 25% of the points of the data set. Note that, as reported in Table I, the TS system is able to output only 87.2% 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 II. 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%	2.5 ± 1.3	80%
10%	2.4 ± 1.3	90%
15%	2.1 ± 0.3	90%
20%	2.3 ± 0.9	90%
25%	2.0 ± 0.0	100%

In [5], we used a three-layer feed-forward neural network instead of the TS model to compute the dissimilarity relation. We performed three experiments with 5%, 15% and 25% of the data set. We obtained 86.22%, 91.% and 96%, respectively, of correctly classified points. Thus, we can conclude that the methods to extract the dissimilarity relation achieve similar performance.

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

Training pool	Correctly classified points	Partition Coefficient
5%	82.4% ± 7.0%	0.83 ± 0.09
10%	84.6% ± 6.3%	0.84 ± 0.05
15%	90.8% ± 5.0%	0.85 ± 0.06
15%	92.7% ± 5.2%	0.85 ± 0.04
25%	95.7% ± 2.5%	0.90 ± 0.02

4.2 The Iris data set

As second example, we used the real data base Iris, provided by the University of California, Irvine (http://www.ics.uci.edu/AI/ML/MLDBRepository.html). Iris contains three classes of Iris plants, namely Iris Setosa, Iris Versicolor and Iris Virginica. Each class consists of 50 patterns characterised by 4 numeric features which describe, respectively, sepal length, sepal width, petal length and petal width. Class Iris Setosa is linearly separable from the other two. However, class Iris Versicolor and Iris Virginica are not separable from each other. Tables IV, V and VI 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 is higher than 90% with just 20% of points in the training pool. Further, in 90% of the trials the number of clusters is equal to the number of classes when the training pool contains 25% of the points. Finally, just with 10% of points in the training pool, the combination TS system–relational clustering algorithm is able to correctly classify nearly 90% of the points.

5 Conclusions

In this paper, we have adopted a combination of supervised and unsupervised learning for clustering data without assuming any preliminary knowledge of the cluster shape. First, we extract a dissimilarity relation directly from the available data by using a TS fuzzy system appropriately trained with a few known dissimilarities between pattern pairs. Then, the dissimilarity relation is input to a new fuzzy relational clustering algorithm which partitions the data set based on the proximity of the vectors containing the dissimilarity values between a pattern and all the 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 the well-known real data set Iris. We have shown that just using a significantly low percentage of known dissimilarities, our method is able to cluster the data sets almost correctly.

Table IV. Percentage of pattern pairs with correct dissimilarity values

Training pool	Number of rules	Correct dissimilarity values before GA	Correct dissimilarity values after GA
5%	8.5 ± 4.1	76.6% ± 4.3%	77.4% ± 3.8%
10%	6.4 ± 3.3	77.3% ± 8.4%	84.3% ± 6.8%
15%	4.3 ± 1.2	77.6% ± 5.5%	87.2% ± 6.1%
20%	4.9 ± 1.5	79.3% ± 4.7%	90.7% ± 2.7%
25%	5.2 ± 1.8	81.7% ± 4.2%	91.6% ± 2.0%

Table V. 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%	3.3 ± 1.1	20%
10%	3.1 ± 0.9	60%
15%	2.8 ± 0.4	80%
20%	3.0 ± 0.5	80%
25%	2.9 ± 0.3	90%

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

Training pool	Correctly classified points	Partition Coefficient
5%	79.0% ± 8.6%	0.67 ± 0.07
10%	89.3% ± 9.5%	0.78 ± 0.09
15%	91.5% ± 8.7%	0.87 ± 0.07
15%	94.3% ± 3.3%	0.89 ± 0.04
25%	95.1% ± 2.7%	0.92 ± 0.04

References

[1] J.C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, New York: Plenum, 1981.

[2] 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.

[3] P. Corsini, B. Lazzerini, F. Marcelloni, "Clustering based on a dissimilarity measure derived from data," Proceedings of KES 2002, Crema, ITALY, pp. 885-889, 2002.

[4] P. Corsini, B. Lazzerini, F. Marcelloni, "A fuzzy relational clustering algorithm based on a dissimilarity measure extracted from data", to be published in *IEEE Transactions on Systems, Man and Cybernetics Part B.*

[5] P. Corsini, B. Lazzerini, F. Marcelloni, "Combining Supervised and Unsupervised Learning for Data Clustering", to be published in *Neural Computing and Applications.*

[6] B. Kamgar-Parsi, A.K. Jain, "Automatic aircraft recognition: toward using human similarity measure in a recognition system," Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 268-273, 1999.

[7] J.F. Kolen, T. Hutcheson, "Reducing the time complexity of the fuzzy C-means algorithm," *IEEE Transactions on Fuzzy Systems*, Vol. 10, No. 2, pp. 263-267, 2002.

[8] Krishnapuram R, Joshi A, Nasraoui O, Yi L, "Low-complexity fuzzy relational clustering algorithms for web mining," *IEEE Transactions on Fuzzy* Systems, Vol. 9, No. 4, pp. 595-607, 2001.

[9] L.J. Latecki, R. Lakamper, "Shape similarity measure based on correspondence of visual parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No 10, pp. 1185 –1190, 2000;.

[10] S. Santini, R. Jain, "Similarity measures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 9, pp. 871 –883, 1999.

[11] M. Setnes, H. Roubos, "GA-fuzzy modeling and classification: Complexity and performance," *IEEE Transactions on Fuzzy Systems*, Vol. 8, No. 5, pp. 509-522, 2000.

[12] T. Takagi, M. Sugeno, "Fuzzy identification of systems and its application to modeling and control," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 15, pp. 116-132, 1985.

[13] D. Valentin, H. Abdi, A.J. O'Toole, G.W. Cottrell, "Connectionist models of face processing: A survey," *Pattern Recognition*, Vol. 27, pp. 1208-1230, 1994.

[14] X.L. Xie, G. Beni, "A validity measure for fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 8, pp. 841-847, 1991.