Supervised Clustering and Fuzzy Decision Tree Induction for the Identification of Compact Classifiers

Ferenc Peter Pach, Janos Abonyi *, Sandor Nemeth and Peter Arva

University of Veszprem, Department of Process Engineering,
P.O. Box 158, H-8201 Veszprem, Hungary
www.fmt.vein.hu/softcomp

Abstract. Fuzzy decision tree induction algorithms require the fuzzy quantization of the input variables. This paper demonstrates that supervised fuzzy clustering combined with similarity-based rule-simplification algorithms is an effective tool to obtain the fuzzy quantization of the input variables, so the synergistic combination of supervised fuzzy clustering and fuzzy decision tree induction can be effectively used to build compact and accurate fuzzy classifiers.

Fuzzy Decision Trees, Fuzzy Clustering, Input Quantization, Fuzzy Classifier

1 Introduction

Decision trees are widely used in pattern recognition, machine learning and data mining applications [12]. Decision trees are important tools in data mining thanks to the understandable representation of the detected information. Hence, the application of decision trees for the initialization of fuzzy and neural models has been already investigated by some researchers [1,13,5,8,14]. In this paper fuzzy decision trees are used to generate interpretable rule-based fuzzy classifiers.

Decision trees were popularized by Quinlan [12] with the ID3 program. The decision tree classifies an example by propagating it along a path from the root node down to a leaf node which contains the classification for this example. One disadvantage of classical crisp decision trees is their brittleness. A wrong path taken down the tree can lead to widely erroneous results. One method to overcome this difficulty is to make the decision tree fuzzy. A fuzzy non-terminal node will allow more than one path to be followed down the tree. The FID tree-building procedure proposed by Janikow is the same as that of ID3. The only difference is based on the fact that a training example can be found in a node to any degree [6].

ID3 and FID assume discrete and fuzzy domains with small cardinalities. This is a great advantage as it increases comprehensibility of the induced knowledge, but may require an *a priori* partitioning. Some research has been done in the area of domain partitioning while constructing a symbolic decision tree. For example, Dynamic- ID3 [3] clusters multivalued ordered domains, and Assistant [7] produces

*Author whom correspondence should be addressed: abonyij@fmt.vein.hu binary trees by clustering domain values (limited to domains of small cardinality). However, most research has concentrated on a priori partitioning techniques [9].

This paper will investigate how supervised clustering can be used for the effective partitioning of the input domains. The application of fuzzy clustering for the quantization of the input variables is not completely new idea. In [11] it has been shown that the results of the clustering coming in the form of a series of prototypes can be directly used to complete a quantization of the continuous attributes. In contrast with most discretization of continuous variables that deal with a single variable only, this approach concerns all the variables discussed at same time. The discretization mechanism is straightforward: project the cluster prototypes on the respective axes (coordinates) and construct the discretization intervals.

Our approach differs from the previously presented methods in the following main issues:

- *Extended fuzzy classifier (Section 2.1)*
  The classical fuzzy classifier consists of rules each one describing one of the classes. In this paper a new fuzzy model structure is applied where each rule can represent more than one classes with different probabilities. The obtained classifier can be considered as an extension of the quadratic Bayes classifier that utilizes mixture of models for estimating the class conditional densities [10].
- *Supervised clustering (Section 2.2)*
  Usually, if the data driven quantization of the domans of the input variables is needed, classical (fuzzy) clustering algorithms are used to estimate the distribution of the data. These algorithms do not utilize the class label of each data point available for the identification, hence the resulted partioning will do not represent the classification problem, but only the distribution of the data. In [2] a supervised clustering algorithm has been worked out that estimates the distributions of the classes and is able for the direct identification an extended fuzzy classifier. The main idea of the paper is the application of this clustering algorithm to generate the fuzzy sets used by the fuzzy decision tree induction algorithms.
- *Similarity-driven simplification (Section 2.3)*
  Since FID assumes fuzzy domains with small cardinalities, the performance and the size of the inducted trees are highly determined by the quality and the number of the membership functions extracted from the clustering. Hence, to obtain a parsimonious and interpretable fuzzy classifiers similarity-driven rule base simplification algorithm was applied [15] to merge the similar fuzzy sets on each input domain.
- *Obtaining fuzzy data and fuzzy partitioning (Section 2.4)*
  Contrary to other clustering based input partitioning approaches, the results of the applied supervised clustering algorithm can be directly used by fuzzy decision tree induction algorithms.

  Beside the effective utilization of the class label information, the main benefit of the applied clustering algorithm is that the clusters are represented by Gaussian membership functions, hence there is not need to project the resulted clusters

into the axis, so there is no projection error that decreases the performance of the model building procedure.
According to the utilized information of the results of the fuzzy clustering algorithm two approaches can be followed at the step of the induction of the decision tree.

  - SC-FID1: The obtained membership functions are only used for the quantization of the input variables and the FID algorithm generates a fuzzy decision tree based on the original training data and the obtained fuzzy partitioning.
  - SC-FID2: The fuzzy clustering is considered as a tool to obtain a compact representation of the class distribution in terms of a fuzzy model. In this approach, each rule of the fuzzy classifier identified by the clustering algorithm is considered as a fuzzy data, so the entire rulebase is a compact ”fuzzy database”. In this approach the FID algorithm does not utilize the original crisp dataset, but it generates the fuzzy decision tree based on fuzzy data generated by the clustering algorithm.

- *Rule-based interpretation and rule-reduction (Section 2.5)*
  In this paper only the SC-FID1 algorithm is applied. After the induction of the tree the resulted decision tree is transformed into a rule-based fuzzy system without any approximation error. Since FID is based on the ID3 algorithm (Section 2.4), the generated fuzzy decision tree is often too complex than needed. This is because every branches of the tree represent fuzzy tests with each membership function on a domain of the selected variable. This often leads to unnecessarily complex classifier due to the addition of meaningless rules. Hence, there is a place for rule-reduction tools to.

The proposed approach is applied for three well-known classification problems available from the Internet: to the Wisconsin Breast Cancer, the Iris, and the Wine classification problems in Section 3. Finally, conclusions are given in Section 4.

2 Supervised Clustering and Fuzzy Decision Tree Induction

2.1 Structure of the Fuzzy Classifier

The utilized fuzzy rule-based classifier consists of fuzzy rules that describe the N_c classes in the given data set. The rule antecedent defines the operating region of the rule in the n-dimensional feature space and the consequent of the fuzzy rule contains the probabilities of the given rule represents the $c_1,\dots,c_C$ classes:

$$\begin{aligned} r_i: \ & \textbf{If } x_1 \text{ is } A_{i,1}(x_{1,k}) \textbf{ and } \dots x_n \text{ is } A_{i,n}(x_{n,k}) \textbf{ then} \\ & \hat{y}_k = c_1 \text{ with } P(c_1|r_i) \dots, \hat{y}_k = c_C \text{ with } P(c_C|r_i) \end{aligned} \tag{1}$$

R is the number of rules, n is the number of features, $x = [x_1, x_2, \dots, x_n]^T$ is the input vector, g_i is the ith rule output and $A_{i,1},\dots,A_{i,n}$ are the antecedent fuzzy sets. The **and** connective is modeled by the product operator allowing for interaction between the propositions in the antecedent. Hence, the degree of activation of the ith rule is calculated as:

$$\beta_i(\mathbf{x}) = \prod_{j=1}^{n} A_{i,j}(x_j), \;\; i = 1, 2, \ldots, R. \tag{2}$$

Similarly to Takagi-Sugeno fuzzy models [16], the rules of the fuzzy model are aggregated using the normalized fuzzy mean formula and the output of the classifier is determined by the *winner takes all* strategy, i.e. the output is the class related to the consequent of the rule that has the highest degree of activation.

$$\hat{y}_k = c_{i^*} \;,\; i^* = \arg\max_{1\le i\le C} \frac{\sum\limits_{l=1}^{R} \beta_l(\mathbf{x}_k) P(c_i|r_l)}{\sum\limits_{i=1}^{R} \beta_l(\mathbf{x}_k)} \tag{3}$$

2.2 Supervised Clustering [2]

Fuzzy clustering algorithms are often used to estimate the distribution of the data. Since these algorithms do not utilize the class label of each data point available for the identification, the fuzzy sets extracted from the clusters do not related to the classification problem. Furthermore, these clusters cannot be directly used to build a classifier. In this paper the proposed cluster prototype and the related clustering algorithm allows the direct supervised identification of fuzzy classifiers presented in the previous section.

To represent the $A_{i,j}(x_{j,k})$ fuzzy set, we use Gaussian membership functions

$$A_{i,j}(x_{j,k}) = \exp\left(-\frac{1}{2} \frac{\left(x_{j,k} - v_{i,j}\right)^2}{\sigma_{i,j}^2} \right) \tag{4}$$

where $v_{j,i}$ represents the center and $\sigma_{i,j}^2$ stands for the variance of the Gaussian function. The parameters of the fuzzy model can be obtained by the following algorithm:

Initialization Given a set of data specify R, choose a termination tolerance $\varepsilon > 0$, and a fuzzy exponent m. Initialize the $\mathbf{U} = [\mu_{i,k}]_{R\times N}$ partition matrix randomly, where $\mu_{i,k}$ denotes the membership that the $\mathbf{z}_k = \{\mathbf{x}_k, y_k\}$ data is generated by the ith cluster.

Repeat for $l = 1, 2, \ldots$

Step 1 Calculate the parameters of the clusters

- Calculate the centers and standard deviation of the Gaussian membership functions:

$$\mathbf{v}_i^{(l)} = \frac{\sum\limits_{k=1}^{N} \left(\mu_{i,k}^{(l-1)}\right)^m \mathbf{x}_k}{\sum\limits_{k=1}^{N} \left(\mu_{i,k}^{(l-1)}\right)^m}, \sigma_{i,j}^{2\,(l)} = \frac{\sum\limits_{k=1}^{N} \left(\mu_{i,k}^{(l-1)}\right)^m (x_{j,k} - v_{j,k})^2}{\sum\limits_{k=1}^{N} \left(\mu_{i,k}^{(l-1)}\right)^m} \tag{5}$$

- Estimate the consequent probability parameters,

$$p(c_i|r_j) = \frac{\sum_{k|y_k=c_i} \left(\mu_{j,k}^{(l-1)}\right)^m}{\sum_{k=1}^{N} \left(\mu_{j,k}^{(l-1)}\right)^m}, 1 \le i \le C, 1 \le j \le R \tag{6}$$

- *A priori* probability of the cluster and the weight (impact) of the rules:

$$P(r_i) = \frac{1}{N} \sum_{k=1}^{N} \left(\mu_{i,k}^{(l-1)}\right)^m, \; w_i = P(r_i) \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi\sigma_{i,j}^2}} \tag{7}$$

Step 2 Compute the distance measure $D_{i,k}^2(\mathbf{z}_k, r_i)$ by

$$\frac{1}{D_{i,k}^2(\mathbf{z}_k, r_i)} = \underbrace{P(r_i) \prod_{j=1}^{n} \exp\left(-\frac{1}{2}\frac{\left(x_{j,k} - v_{i,j}\right)^2}{\sigma_{i,j}^2}\right)}_{\text{Gath-Geva clustering}} P(c_j = y_k|r_i) \tag{8}$$

This distance measure consists of two terms. The first term is based on the geometrical distance between the $\mathbf{v}_i$ cluster centers and the $\mathbf{x}_k$ observation vector, while the second is based on the probability that the r_i-th cluster describes the density of the class of the k-th data, $P(c_j = y_k|r_i)$ It is interesting to note that this distance measure only slightly differs from the unsupervised Gath–Geva clustering algorithm which can also be interpreted in a probabilistic framework [4]. However, the novelty of the proposed approach is the second term, which allows the use of class labels.

Step 3 Update the partition matrix

$$\mu_{i,k}^{(l)} = \frac{1}{\sum\limits_{j=1}^{R} \left(D_{i,k}(\mathbf{z}_k, r_i)/D_{j,k}(\mathbf{z}_k, r_j)\right)^{2/(m-1)}}, 1 \le i \le R, 1 \le k \le N \tag{9}$$

until $||\mathbf{U}^{(l)} - \mathbf{U}^{(l-1)}|| < \varepsilon$.

2.3 Transformation and Merging of the Membership Functions

The previously presented clustering algorithm obtains a rule-based fuzzy classifier defined with Gaussian membership functions. Since the public program of the FID algorithm uses trapezoid fuzzy membership functions to describe the fuzzy sets $A_{i,j}(x_j)$:

$$\mu_{i,j}(x_j; a, b, c, d) = \max\left(0, \min\left(\frac{x_j - a}{b - a}, 1, \frac{d - x_j}{d - c}\right)\right). \tag{10}$$

there is a need to transform the obtained Gaussian membership functions into trapezoidals. For this transformation we used the values of the Gaussian function at the points $v_{i,j} \pm 3 * \sigma_{i,j}$.

Reduction of the fuzzy classifier is achieved by a rule-base simplification method based on a similarity measure to quantify the redundancy among the fuzzy sets in the rule-base and subsequent set-merging [15]. A similarity measure based on the set-theoretic operations of intersection and union is applied:

$$S(A_{i,j},A_{k,j}) = \frac{|A_{i,j} \cap A_{k,j}|}{|A_{i,j} \cup A_{k,j}|} \tag{11}$$

where $|.|$ denotes the cardinality of a set, and the $\cap$ and $\cup$ operators represent the intersection and union, respectively. If $S(A_{i,j},A_{k,j}) = 1$, then the two membership functions $A_{i,j}$ and $A_{k,j}$ are equal. $S(A_{i,j},A_{k,j})$ becomes 0 when the membership functions are non-overlapping. During the rule-base simplification procedure similar fuzzy sets are merged when their similarity exceeds a user-defined threshold $\theta \in [0,1]$ (θ=0.5 is applied). Merging reduces the number of different fuzzy sets (linguistic terms) used in the model and thereby increases the transparency. The similarity measure is also used to detect “don’t care” terms, i.e., fuzzy sets in which all elements of a domain have a membership close to one. If all the fuzzy sets for a feature are similar to the universal set, or if merging let to only one membership function for a feature, then this feature is eliminated from the model. The complete rule-base simplification algorithm is given in [15].

This method has been extended with an additional rule pruning step, where rules that are only responsible for a few number of classifications are deleted form the rule-base, because they are only cover exceptions or noise in the data. This pruning is based on the activity of the rules measured by the sum of the certainty degree.

2.4 Fuzzy Decision Tree Induction - Implementation

The induction of decision trees can be based on two types of data. The first approach generates a decision tree based on the obtained membership functions and the original database of crisp data. Instead of crisp data, fuzzy data can also be used for the tree induction. This results in two approaches:

- SC-FID1: The obtained membership functions are only used for the quantization of the input variables and the FID algorithm generates a fuzzy decision tree based on the original training data and the obtained fuzzy partitioning.
- SC-FID2: The fuzzy clustering is considered as a tool to obtain a compact representation of the class distribution in terms of a fuzzy model. In this approach, each rule of the fuzzy classifier identified by the clustering algorithm is considered as a fuzzy data, so the entire rulebase is a compact ”fuzzy database”. In this approach the FID algorithm does not utilize the original crisp dataset, but it generates the fuzzy decision tree based on fuzzy data generated by the clustering algorithm.

Since for the induction of the decision trees we used the standard FID algorithm without any modifications (The FID 3.3 is downloadable from `http://www.cs.umsl.edu/ janikow/fid/`) the principles of tree induction algorithm is not shown in this paper, the details are discoverable in [6].

To make the application of this program easy in the MATLAB programming environment, we developed an interface which supports all the functions of FID (included the testing and using of the generated trees). This MATLAB toolbox is available from our website: `http://www.fmt.vein.hu/softcomp`. The main feature of this interface program is that the generated tree can be transformed into a rule-based fuzzy model in the format of the Fuzzy Model Identification Toolbox
(`http://www.dcsc.tudelft.nl/ babuska/`).

2.5 Rule Base Generation from Fuzzy Decision Tree

Trees can be represented in terms of logical rules, where each concept is represented by one disjunctive normal form, and where the antecedent consists of a sequence of attribute value tests. These attribute value tests partition the input domains of the classifier into intervals represented by the fuzzy sets, and the operating region of the rules is formulated by **and** connective of these domains.

The previous considerations can be generalized to form an algorithm that can be used for the transformation of decision trees into initial fuzzy systems.

1. $i = 1, \dots, R$, where R is identical to the number of leafs (terminal nodes), i.e. the number of rules of the fuzzy classifier.
2. Select a terminal node and collect the attribute value tests $A_{i,j}$ related to the chosen terminal node.
3. The T_i attribute value tests define the antecedent part of the i-th rule, while the consequent part is given by the labels and the $P(c_1|r_i) \dots, P(c_C|r_i)$ probabilities given by FID algorithm.

This method has been extended with an additional rule pruning step, where rules that are only responsible for a few number of classifications are deleted form the rule-base, because these only cover exceptions or noise in the data.

3 Comparative Application Study

This section is intended to provide a comparative study based on a set of multivariate classification problems to present how the performance and the complexity of the classifier is changing trough the step-wise model building procedure of the SC-FID1 algorithm.

The chosen Iris, Wine and Wisc data, coming from the UCI Repository of Machine Learning Databases (*http://www.ics.uci.edu*), are example of classification problems with different complexity, e.g. large and small number of features (see Table 1).

During the experiments, the performance of the classifiers are measured by tenfold cross validation. This means, that the data is divided into ten sub-sets, and each sub-set is left out once, while the other nine are applied for the construction of the classifier which is subsequently validated for unseen cases in the left-out sub-set.

Table 1. Complexity of the classification problems.

Problem	♯Samples	♯Features	♯Classes
Iris	150	4	3
Wine	178	13	3
Wisc	699	9	2

To demonstrate the effectiveness of the applied supervised clustering, the obtained results are compared to the performances and complexities of the classifiers obtained by the unsupervised Gath-Geva clustering and uniform partition (i.e. Ruspini partition: triangular membership functions). The performances of the classifiers during the model building procedures are shown in (Figure 1, Figure 2, and Figure 3). The model building procedures are monitored by logging the number of rules, conditions and performances of the classifiers. As Table 2 shows, with the use of the proposed techniques, transparent and compact fuzzy classifiers are resulted, and the input partitioning obtained by the supervised clustering gives the best classifiers.

Table 2. Classification rates (acc.) achieved on the WINE classification problem. Average results of tenfold validation at the number of clusters: 3-7.

Clusters	Supclust	SC-FID1	GGclust	C-FID1	Ruspini
c=3	96.63	96.08	93.85	94.90	96.60
c=4	95.00	96.05	92.74	95.98	96.05
c=5	94.41	96.60	90.46	95.46	93.79
c=6	95.49	95.49	92.71	96.05	92.65
c=7	97.22	95.46	94.44	96.60	89.77

As it appears from the figures, the best performances are usually obtained by the rule-based fuzzy classifiers by the supervised clustering algorithm. The accuracy of these models decreases considerably after the transformation of the Gaussian membership function into trapezoidal ones. However, after the merging of membership functions and the induction of the decision tree accurate, yet compact classifiers can be obtained.

The fuzzy decision trees induced based on uniform partition of the input variables gives lower accuracy compacted to the clustering based performances, so the effectiveness of rule reduction method appears.

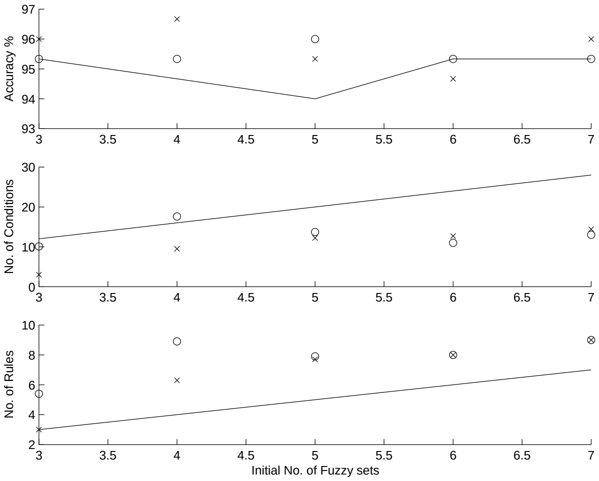

Fig. 1. Classification performances and complexities of the classifiers on the IRIS data sets in the function of the number of clusters. (0: Uniform partitioning, X: SC-FID1, –: Supervised clustering based classifier)

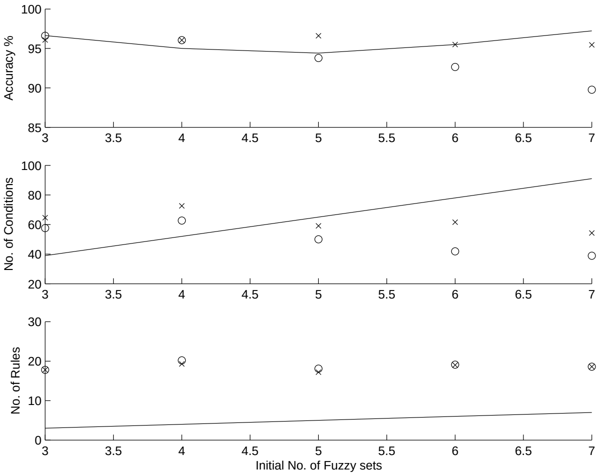

Fig. 2. Classification performances and complexities of the classifiers on the WINE data sets in the function of the number of clusters. (0: Uniform partitioning, X: SC-FID1, –: Supervised clustering based classifier)

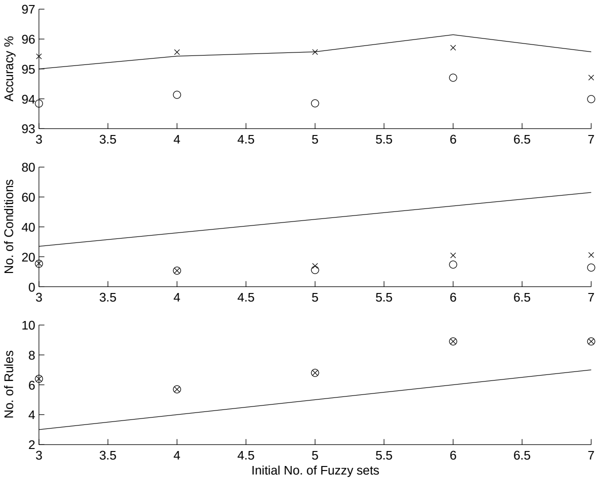

Fig. 3. Classification performances and complexities of the classifiers on the WISC data sets in the function of the number of clusters. (0: Uniform partitioning, X: SC-FID1, –: Supervised clustering based classifier)

4 Conclusion

In this paper a new approach to the identification of compact and accurate fuzzy classifiers has been presented. The novelty of this approach is that each rule can represent more than one classes with different probabilities. For the identification of the fuzzy classifier a supervised clustering method has been used to provide input partitioning to the fuzzy decision tree induction algorithm. The proposed identification approach is demonstrated by the Wisconsin Breast Cancer, the Iris and the Wine benchmark classification problems. The comparison to the uniform partitioning based decision tree induction method indicates that the proposed supervised clustering method effectively utilizes the class labels and able to lead to compact and accurate fuzzy systems with the help of decision tree induction.

References

1. J. Abonyi, H. Roubos, and F. Szeifert. Data-driven generation of compact, accurate, and linguistically sound fuzzy classifiers based on a decision tree initialization. *International Journal of Approximate Reasoning*, pages 1–21, 2003.
2. J. Abonyi and F. Szeifert. Supervised fuzzy clustering for the identification of fuzzy classifiers. *Pattern Recognition Letters*, 24(14):2195–2207, 2003.
3. R. Gallion, D.C.St. Clair, and W.E Bond C.Sabharwahl. Dynamic id3: A symbolic learning algorithm for many-valued attribute domains. In *in Proc. 1993 Symp.Applied Computing.*, pages 14–20, New York, ACM Press, 1993.
4. Gath I. and Geva. A.B. Unsupervised optimal fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7:773781, 1989.
5. I. Ivanova and M. Kubat. Initialization of neural networks by means of decision trees. *Knowledge-Based Systems*, 8:333–344, 1995.
6. C.Z. Janikow. Fuzzy decision trees: Issues and methods. *IEEE Trans. SMC-B*, 28:1–14, 1998.
7. I. Konenko and E. Roskar I. Bratko. *Experiments in automatic learning of medical diagnostic rules*. Tech. Rep., J. Stefan Inst. Yugoslavia, 1994.
8. M. Kubat. Decision trees can initialize radial-basis-function networks. *IEEE Trans. NN*, 9:813–821, 1998.
9. M. Lebowitz. Categorizing numeric information for generalization. *Cognitive Science*, 9:285–308, 1985.
10. Kambhatala N. *Local Models and Gaussian Mixture Models for Statistical Data Processing*. Ph.D. Thesis, Oregon Gradual Institute of Science and Technology, 1996.
11. W. Pedrycz and A. Zenon Sosnowskic. The design of decision trees in the framework of granular data and their application to software quality models. *Fuzzy Sets and Systems*, 123:271290, 2001.
12. Quinlan. Induction on decision trees. *Machine Learning*, 1(1):81–106, 1986.
13. L.K. Sethi. Entropy nets: From decision trees to neural networks. *Proc. IEEE*, 78:1605–1613, 1990.
14. R. Setiono and W.K. Leow. On mapping decision trees and neural networks. *Knowledge Based Systems*, 13:95–99, 1999.
15. M. Setnes, R. Babuška, U. Kaymak, and H.R. van Nauta Lemke. Similarity measures in fuzzy rule base simplification. *IEEE Trans. SMC-B*, 28:376–386, 1998.
16. T. Takagi and M. Sugeno. Fuzzy identification of systems and its application to modeling and control. *IEEE Trans. SMC*, 15:116–132, 1985.