ELSEVIER

Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE DIRECT®

International Journal of Approximate Reasoning 32 (2003) 1–21

INTERNATIONAL JOURNAL OF APPROXIMATE REASONING

www.elsevier.com/locate/ijar

Data-driven generation of compact, accurate, and linguistically sound fuzzy classifiers based on a decision-tree initialization

Janos Abonyi [a,*], Johannes A. Roubos [b], Ferenc Szeifert [a]

[a] *Department of Process Engineering, University of Veszprem, P.O. Box 158, H-8200, Veszperm, Hungary*

[b] *Department of Information Technology and Systems, Systems and Control Engineering, Delft University of Technology, P.O. Box 5031, 2600 GA Delft, The Netherlands*

Received 1 July 2001; accepted 1 March 2002

Abstract

The data-driven identification of fuzzy rule-based classifiers for high-dimensional problems is addressed. A binary decision-tree-based initialization of fuzzy classifiers is proposed for the selection of the relevant features and effective initial partitioning of the input domains of the fuzzy system. Fuzzy classifiers have more flexible decision boundaries than decision trees (DTs) and can therefore be more parsimonious. Hence, the decision tree initialized fuzzy classifier is reduced in an iterative scheme by means of similarity-driven rule-reduction. To improve classification performance of the reduced fuzzy system, a genetic algorithm with a multiobjective criterion searching for both redundancy and accuracy is applied. The proposed approach is studied for (i) an artificial problem, (ii) the Wisconsin Breast Cancer classification problem, and (iii) a summary of results is given for a set of well-known classification problems available from the Internet: Iris, Ionosphere, Glass, Pima, and Wine data.

[*] Corresponding author. Tel.: +36-88-422-022/4201.
E-mail addresses: abonyij@fmt.vein.hu (J. Abonyi), hans@ieee.org (J.A. Roubos).
URLs: http://www.fmt.vein.hu/softcomp (J. Abonyi), http://LCEwww.et.tudelft.nl/ (J.A. Roubos).

0888-613X/02/$ - see front matter © 2002 Elsevier Science Inc. All rights reserved.
PII: S0888-613X(02)00076-2

Keywords: Classification; Fuzzy classifier; Decision tree; Genetic algorithm; Model reduction

1. Introduction

As a result of the increasing complexity and dimensionality of classification problems, it becomes necessary to deal with structural issues of the identification of classifier systems. Important aspects are the selection of the relevant features and the determination effective initial partition of the input domain [1]. Moreover, when the classifier is identified as part of an expert system, the linguistic interpretability is also an important aspect which must be taken into account. The first two aspects are often approached by an exhaustive search or educated guesses, while the interpretability aspect is often neglected. Only recently people recognized the importance of all these aspects [2,3], which makes the automatic data-based identification of classification systems that are compact, interpretable and accurate, a challenging topic.

We propose fuzzy logic rule-based classifiers to handle the interpretability aspect. Fuzzy logic helps to improve the interpretability of knowledge-based classifiers through its semantics that provide insight in the classifier structure and decision making process. Fuzzy logic, however, is not a guarantee for interpretability, as was also recognized in [2,3]. Real effort must be made to keep the resulting rule-base transparent [4–6]. For this purpose, two main approaches are followed in the literature: (i) selection of a low number of input variables in order to create a compact classifier [4,7], and (ii) construction of a large set of possible rules by using all inputs, and subsequently use this set to make a useful selection out of these rules [6,8]. Often genetic algorithms are applied for this rule-selection. In both approaches, further model reduction can obtained by generalization and/or similarity-driven set-reduction techniques [3].

For high-dimensional classification problems, the initialization step of the identification procedure of the fuzzy model becomes very significant. Common initializations methods such as grid-type partitioning [8] and *rule generation on extrema* initialization [6], result in complex and non-interpretable initial models and the rule-base simplification and reduction step become computationally demanding. To obtain compact initial fuzzy models fuzzy clustering algorithms [4] or similar but less complex covariance-based initialization techniques [7] were put forward, where the data is partitioned by ellipsoidal regions (multivariable membership functions). Normal fuzzy sets can then be obtained by an orthogonal projection of the multivariable membership functions onto the input–output domains. The projection of the ellipsoids results in hyperboxes in the product space. The information loss at this step makes the model suboptimal resulting in a much worse performance than the initial model defined by multivariable membership functions. However, gaining linguistic interpretability is the main advantage derived from this step.

To avoid problems associated with the described approaches, a crisp decision-tree-based initialization technique is proposed. This proposal is motivated by the high performance and computational efficiency of the existing decision tree generation methods that are effective in the selection of the relevant features and initial partitioning of the input domain [9]. The application of decision and regression trees for the initialization of fuzzy and neural models has been already investigated by some researcher. In [10] a decision tree was mapped into a feedforward neural network. A variation of this method is given in [11] where the decision tree was used for the input domains discretization only. This approach was extended with a model pruning method in [12]. In [13], the decision tree was applied to initialize radial-basis functions for a neural network, because feedforward neural networks are expensive to train, and the abundance of their parameters may render the training procedure inefficient if the training set is small. This method was based on the placement of radial-basis functions to the center or the edge of the rectangular regions defined by the decision tree. The complexity of the resulted model can be controlled by the complexity of the decision tree [13] or by the number of the added basis functions [14]. As radial-basis functions are functionally equivalent to fuzzy inference systems [15,16], this approach is identical to LOLIMOT [17] that initializes fuzzy models from regression trees. A similar approach is the simple fuzzification of the decisions in the regression tree. This results in a fuzzy CART model [18], where the antecedent part of the fuzzy model is build up from fuzzy inequalities.

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

Initialization of the fuzzy system. Contrary to other methods, the crisp binary decision tree is transformed into a fuzzy system without any approximation error by a one-to-one mapping. This is possible because the proposed fuzzy classifiers utilize trapezodial membership functions. The membership functions are chosen during the initialization in such a way that they are equivalent to crisp sets. The initial fuzzy system is therefore equivalent to a crisp rule-based classifier, which is only an alternative representations of the decision tree.

No tuning of the fuzzy system. Most methods for transformation of DTs into fuzzy systems deteriorate the classification. Usually a tuning step is necessary to recover the accuracy. This often leads to increased complexity of the fuzzy classifier due to the addition of rules and/or fuzzy sets to compensate for this negative transformation effect. The proposed initialization approach does not introduce an approximation error, such that there is no need to increase the complexity of the fuzzy model.

DT-based classifiers perform a rectangular partitioning of the input space, while fuzzy models generate non-axis parallel decision boundaries [19]. Hence, the main advantage of rule-based fuzzy classifiers over crisp DTs is the greater flexibility of the decision boundaries. Therefore fuzzy classifiers can be more parsimonious than DTs and one may conclude that the fuzzy classifiers, based on the transformation of DTs only [17,18], will usually be more complex than necessary. This suggest that the simple transformation of a DT into a fuzzy model may be successfully followed by model-reduction steps to reduce the model's complexity and improve its interpretability. We propose rule-base optimization and simplification steps for this purpose. Hence, to obtain a parsimonious and interpretable fuzzy classifiers the following approach is taken. First the initial fuzzy classifiers is obtained by an exact transformation of the decision tree. Then we apply similarity-driven rule-base simplification algorithm [3] and a genetic algorithm (GA)-based parameter optimization in an iterative way to improve the classification accuracy and compactness, while ensuring the transparency classifier.

In the sequel, we focus on the decision tree based initialization step. For the second step, the classifier tuning, several notes are given while the details can be found elsewhere [7]. Section 2 explains the structure of the fuzzy classifier. In Section 3, the transformation of decision trees to fuzzy models is discussed. The model simplification techniques are reviewed in Section 4. Section 5 considers several classification problems. The proposed approach is studied for a two-class artificial geometric problem, followed by the Wisconsin Breast Cancer classification problem, and subsequently, a summary of results is given for a set of well-known classification problems available from the Internet: Iris, Ionospehere, Glass, Pima, and Wine data. Finally, conclusions are given in Section 6.

2. Structure of the fuzzy classifier

The 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 rule consequent is a crisp (non-fuzzy) class label from the set $g_i \in \{1, 2, \ldots, N_c\}$:

$$R_i : \textbf{If } x_1 \text{ is } A_{i1} \textbf{ and } \ldots x_n \text{ is } A_{in} \textbf{ then } g_i, \quad i = 1, \ldots, M, \tag{1}$$

where M is the number of rules, n is the number of features, $\vec{x} = [x_1, x_2, \ldots, x_n]^T$ is the input vector, g_i is the ith rule output and $A_{i1}, \ldots, A_{in}$ 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:

$$\beta_i(\vec{x}) = \prod_{j=1}^{n} A_{ij}(x_j), \quad i = 1, 2, \ldots, M. \tag{2}$$

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:

$$y = g_i^*, \quad i^* = \underset{1 \leqslant i \leqslant M}{\arg\max}, \beta_i. \tag{3}$$

The certainty degree of the decision is given by the normalized degree of firing of the rule:

$$\mathrm{CF} = \beta_{i^*} / \sum_i^M \beta_i. \tag{4}$$

3. Initialization of the fuzzy classifier by a decision tree

3.1. Construction of decision trees

Throughout the paper, binary decision trees are applied to create the initial classifier rule-base. A binary decision tree consists of two type of nodes: (i) internal nodes having two children and (ii) terminal nodes without children. Each internal node is associated with a decision function to indicate which node to visit next. Each terminal node represents the output of a given input that leads to this node, i.e., in classification problems each terminal node contains the label of the predicted class (Fig. 1).

The decision tree construction algorithms generate decision trees from a set D of cases. Theses algorithms partition the data set D into subsets $D_1, D_2, \ldots, D_M$ by a set of tests T with mutually outcomes $T_1, T_2, \ldots, T_M$, where D_i contains those cases that have outcome T_i. The C4.5 [9] is such an binary decision tree generating algorithm and is applied in the following. For numeric (continuous) attributes the attribute test is written as $x_j < t$. The t-thresholds are selected based on a splitting criterion. The default splitting criterion used by C4.5 is the *gain ratio*, as an information-based measure that takes into account different probabilities of the outcomes. The gain ratio is explained as follows. The residual uncertainty about the class to which a case in D belongs can be expressed as:

$$\mathrm{Info}(D) = -\sum_{j=1}^{M} p(D,j) \times \log_2(p(D,j)), \tag{5}$$

where $p(D,j)$ denotes the proportion of classes in D that belong to the jth class. The information gained by a test is strongly effected by the number of outcomes and is maximal when there is one class in each subset D_i:

$$\mathrm{Gain}(D,T) = \mathrm{Info}(D) - \sum_{i=1}^{M} \frac{|D_i|}{|D|} \times \mathrm{Info}(D_i), \tag{6}$$

where $|D_i|$ denotes the cardinality of the D_i data set.

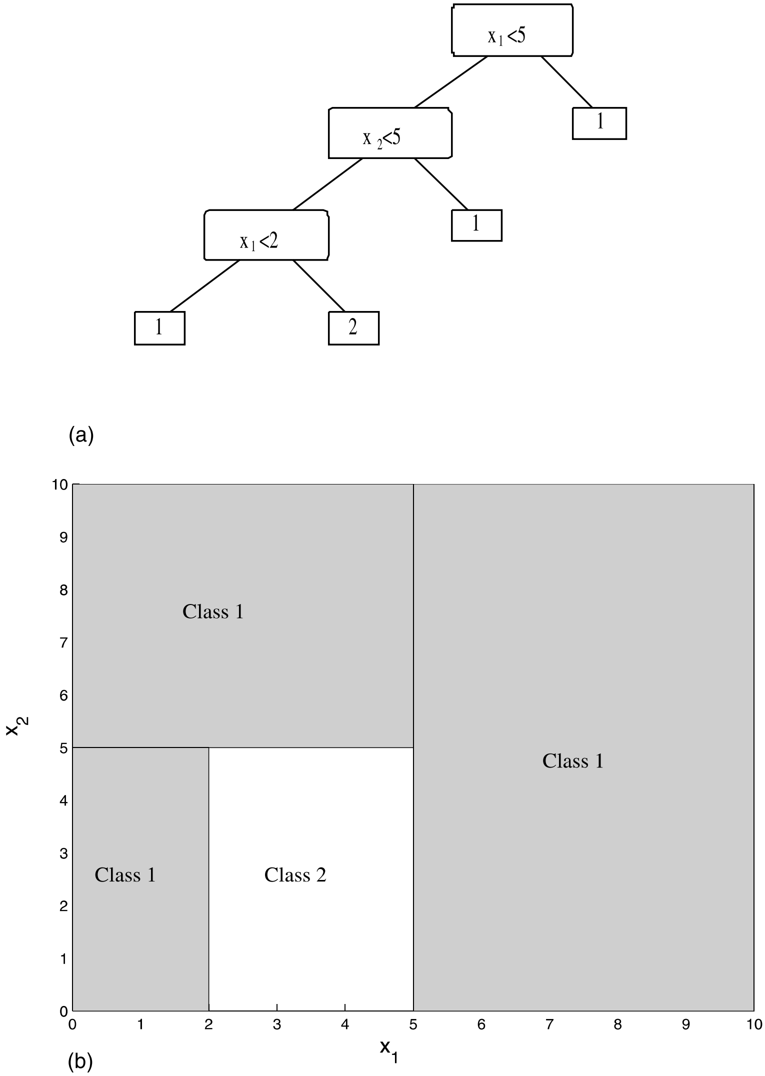

Fig. 1. Example of a binary decision tree: (a) Binary decision tree. (b) The decomposed features space.

On the other hand, the potential information obtained by partitioning a set of cases is based on knowing the subset D_i, into which a case falls. This *split information* is:

$$\mathrm{Split}(D,T) = -\sum_{i=1}^{M} \frac{|D_i|}{|D|} \times \log_2\left(\frac{|D_i|}{|D|}\right), \tag{7}$$

which tends to increase with the number of outcomes of a test. The gain ratio criterion assesses the desirability of a test as the ratio of its information gain to its split information. The gain ratio of every possible test is determined, and among those with at least average gain, the split with maximum gain ratio is selected [9]. The recursive partition strategy results in trees that are consistent with the training data. In practical applications, data contains often noise, which leads generally to too complex trees. Hence, most decision tree construction methods prune the initial tree by identifying sub-trees that contribute only a little to the predictive accuracy by replacing these by a leaf.

3.2. Transformation of the decision tree into a fuzzy model

Binary trees can be represented in terms of crisp logical rules, where each concept is represented by one disjunctive normal form, and where the antecedent consists of a sequence of attribute value tests, e.g., $x_j < 5$. As attributes can appear more than once in a tree, the attribute value tests partitions the input domains of the classifier into intervals. These intervals can be represented by crisp characteristic sets, and the operating region of the rules are formulated by **and** connective of these domains.

These crisp characteristic sets are the extremum case of trapezoidal fuzzy membership functions, μ_{ij}, that are often used to describe fuzzy sets $A_{ij}(x_j)$:

$$\mu_{ij}(x_j; a, b, c, d) = \max\left(0, \min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right)\right). \tag{8}$$

Thus, decision trees can be represented by fuzzy rules with trapeziodal membership functions. For example, the rectangular region of class 2, defined by the depicted decision tree (Fig. 1) can be represented by the fuzzy rule:

$$R_1 : \textbf{If } x_1 \text{ is } A_{11} \textbf{ and } x_2 \text{ is } A_{12} \textbf{ then } g_1 = 2, \tag{9}$$

where A_{11} and A_{12} are defined as $\mu_{11}\{x_1, 2, 2, 5, 5\}$ and $\mu_{12}\{x_2, 0, 0, 5, 5\}$, respectively.

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. $i = 1, \ldots, M$.
2. Select a terminal node of the DT defines D_i data set.
3. Collect the attribute value tests T_i related to the chosen terminal node.
4. The T_i attribute value tests define a hypercube that contains the D_i data set and can be used to formulate the ith rule and define the characteristic points of the fuzzy sets.

4. Reduction and tuning of the initialized fuzzy classifier

4.1. Motivation for the model reduction

The crisp decision tree is thus transformed into a crisp rule base with the same structure as the fuzzy rule base that we have in mind. There are basically two reasons for the transformation from the crisp decision tree/rule-base into a fuzzy rule-base: (i) fuzzy classifiers in comparison with crisp classifiers contain additional information about the certainty degree of the classifier decision (4) and (ii) fuzzy systems can easily define non-axis parallel decision boundaries, while DTs always approximate such systems in a step-wise manner [19]. An example is given in Fig. 2. As this figure suggests, for an accurate approximation of a non-axis parallel class, many crisp decision rules are needed, while a fuzzy model with two rules provides a perfect solution:

$$\begin{aligned} R_1 &: \textbf{If } x_1 \text{ is } A_{11} \textbf{ and } x_2 \text{ is } A_{12} \textbf{ then } g_1 = 1, \\ R_2 &: \textbf{If } x_1 \text{ is } A_{21} \textbf{ and } x_2 \text{ is } A_{22} \textbf{ then } g_2 = 2. \end{aligned} \tag{10}$$

As it is shown in Fig. 2, the obtained membership functions overlap. Because of the interpolation effect of the fuzzy inference between overlapping, non-rectangular fuzzy sets, the resulted classification boundary can be smooth and non-axis parallel. These advantageous properties of fuzzy systems makes the fuzzy rule-based classifier much more parsimonious than crisp decision trees. This suggests that the transformation of a DT into a fuzzy model should be followed by a series of rule-base simplification and membership function tuning steps. In the following subsection it will be shown that the algorithm starts from rectangular membership functions extracted from the DTs. These rectangular membership functions are parameterized as extreme cases of trapezoids, and then tuned by using genetic algorithm to provide optimal non-axis parallel decision boundaries.

4.2. Reduction and tuning algorithm

In the previous subsection, it was shown that the fuzzy model obtained from the binary decision tree, may contain unnecessary complexity since fuzzy classifiers are able to define non-axis-parallel decision boundaries while crisp decision trees cannot. An iterative optimization-model reduction method is proposed to reduce the classifier while maintaining the accuracy. The accuracy usually decreases in each reduction step but can be regained to some extent by tuning the membership functions. A genetic algorithm (GA) is applied to tune the antecedent membership functions [20]. The user has to decide how much accuracy loss allows for a certain gain in transparency.

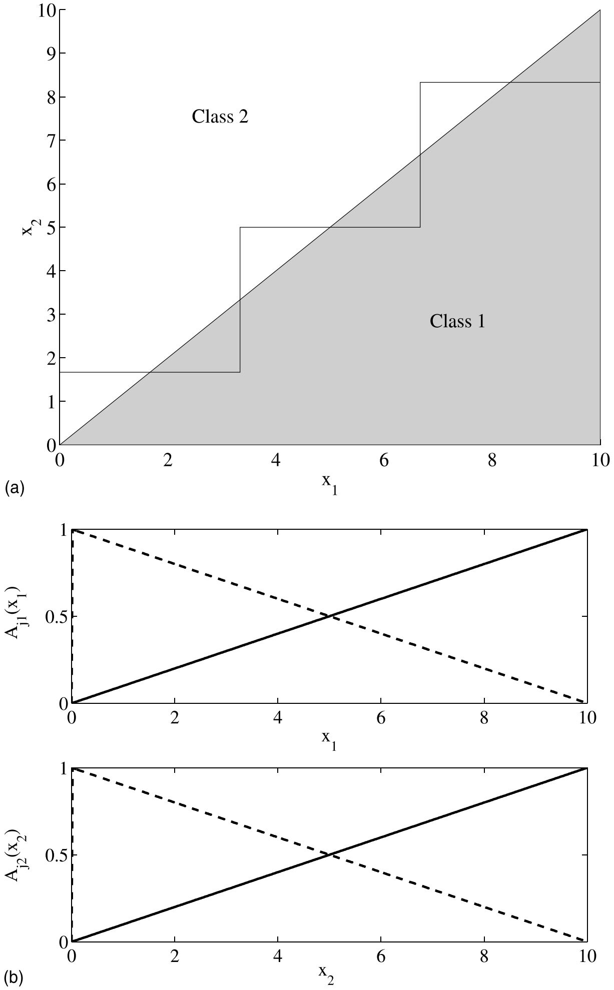

Fig. 2. Solution of a linearly separable classification problem by a decision tree and a fuzzy model: (a) The classification problem and the approximate decision boundary of a crisp rule-based system. (b) Membership functions of the fuzzy model that gives a perfect classification.

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 [3]. A similarity measure based on the set-theoretic operations of intersection and union is applied:

$$S(A_{ij}, A_{kj}) = \frac{|A_{ij} \cap A_{kj}|}{|A_{ij} \cup A_{kj}|}, \tag{11}$$

where $|\cdot|$ denotes the cardinality of a set, and the $\cap$ and $\cup$ operators represent the intersection and union, respectively. If $S(A_{ij}, A_{kj}) = 1$, then the two membership functions A_{ij} and A_{kj} are equal. $S(A_{ij}, A_{kj})$ 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 $\theta \in [0, 1]$ ($\theta = 0.5$ 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 led 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 [3].

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. This pruning is based on the activity of the rules measured by the sum of the certainty degree (4). The proposed rule-base simplification method is illustrated in Fig. 3.

The combination of the parameter optimization and rule-base simplification algorithm resulted a three-step modeling scheme (Fig. 4).

After the DT-based initialization phase, in the model reduction phase the GA is forced to emphasize the redundancy in the model to increase the number of the possible removable fuzzy sets and rules as proposed in [7,21]. To reward similarity during the iterative process, the misclassification rate is combined with a similarity measure in the GA objective function. The achieved redundancy is then used to remove unnecessary fuzzy sets in the next iteration. In the

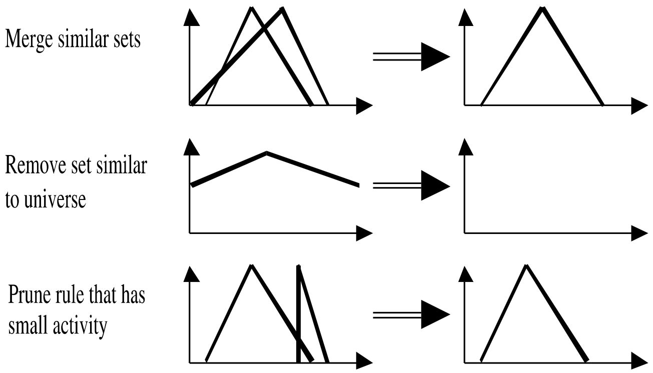
Merge similar sets
Remove set similar to universe
Prune rule that has small activity

Fig. 3. Simplification of the fuzzy classifier.

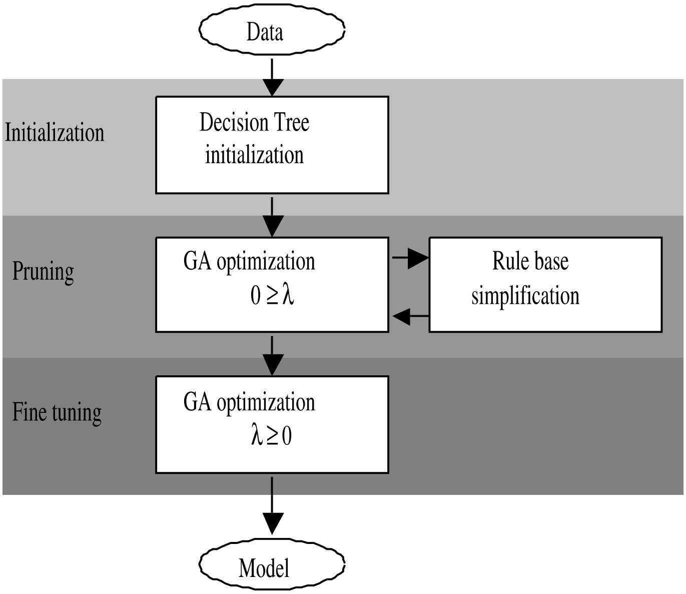

Fig. 4. Scheme of the complete DT identification approach.

fine-tuning step, the combined similarity among fuzzy sets was penalized to obtain a distinguishable term set for linguistic interpretation. The tradeoff between similarity rewarding-penalizing results in the following multiobjective function to be minimized by the GA:

$$J = (1 + \lambda S^*) \cdot \mathrm{MCE}, \tag{12}$$

where MCE represents the mean classification error of the model, $S^* \in [0, 1]$ is the average of the maximum pairwise similarity that is present in each input, i.e., S^* is an aggregated similarity measure for the total model, and the weighting function $\lambda \in [-1, 1]$ determines whether similarity is rewarded ($\lambda < 0$) or penalized ($\lambda > 0$).

The absolute value of λ determines the trade-off between the similarity objective and the accuracy. Normally some experience is necessary to decide about a good value, however the final results seems to be not highly sensitive for the exact value. Generally, good results were obtained with $|\lambda|$ values in the range $[0, 2]$ [22].

Details of the applied real-coded GA can be found in [4]. The GA was applied with a population size $L = 40$, number of chromosomes $n_C = 10$, domain parameters $\alpha_1 = 25\%$ and $\alpha_2 = 25\%$ and number of generations $T = 50$ in the final optimization and $T = 100$ in the complexity reduction step. The threshold $\lambda = 1$ for redundancy searches and $\lambda = -1$ in the final optimization.

The threshold for set merging was $\theta = 0.5$ and $\theta = 0.8$ for removing sets similar to the universal set ("don't care" terms).

5. Performance evaluation

In order to examine the performance of the proposed identification method a set of examples is presented in this section. The first example is an artificial problem with geometrical data to demonstrate the capabilities of the algorithm. The second more detailed example is the Wisconsin Breast Cancer classification problem, which is a benchmark problem from the literature. Finally, a comparative study based on a set of well-known multidimensional classification problem is presented. This study is performed to evaluate the performance of the proposed method for several problems varying in complexity, e.g., an increasing number of classes and features.

5.1. Example 1: Geometrical data

A simple two-dimensional two-class geometric classification problem has been defined to investigate the capabilities of the proposed classifier generation algorithm. The domain of class two is represented by the shaded area of Fig. 5. The training and the testing set were generated by taking 1000 and 500 uniformly distributed samples in the $[0, 10] \times [0, 10]$ domain.

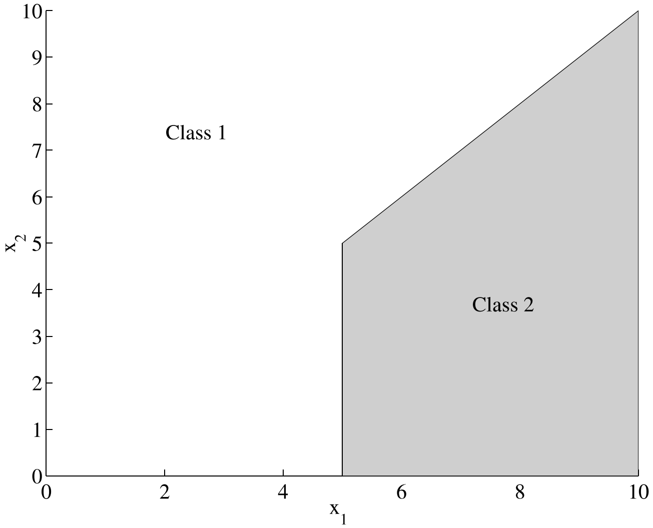

Fig. 5. The geometric classification problem.

```
feature_1 <= 5.00707 : 1
feature_1 > 5.00707 :
|    feature_2 <= 7.32887 :
|    |    feature_2 <= 5.68109 : 2
|    |    feature_2 > 5.68109 :
|    |    |    feature_1 > 6.62893 : 2
|    |    |    feature_1 <= 6.62893 :
|    |    |    |    feature_2 > 5.9933 : 1
|    |    |    |    feature_2 <= 5.9933 :
|    |    |    |    |    feature_1 <= 5.85027 : 1
|    |    |    |    |    feature_1 > 5.85027 : 2
|    feature_2 > 7.32887 :
|    |    feature_1 <= 8.46436 : 1
|    |    feature_1 > 8.46436 :
|    |    |    feature_2 <= 8.83811 : 2
|    |    |    feature_2 > 8.83811 :
|    |    |    |    feature_2 > 9.41078 : 1
|    |    |    |    feature_2 <= 9.41078 :
|    |    |    |    |    feature_1 <= 8.97657 : 1
|    |    |    |    |    feature_1 > 8.97657 : 2
```

Fig. 6. Decision tree generated by C4.5 for the geometric problem.

An initial decision tree was generated by the C4.5 algorithm. Because of the non-axis parallel decision problem, a complex tree resulted with 20 internal and 11 terminal nodes as shown in Fig. 6.

Because of the large number of parameters and the noise-free conditions, the performance of the resulted tree was excellent, the recognition rate was 99.9% on the training set and 99.2% on the test set. However, as can be seen from Fig. 6, the resulted model is not really transparent.

To enhance interpretability and compactness, the resulted decision tree is transformed into a fuzzy model and the previously presented model optimization-pruning algorithm has been applied. Surprisingly, after two rule-base reduction and optimization step, the following simple rule-base resulted:

$$R_1 : \textbf{If } x_1 \text{ is } A_{11} \textbf{ then } g_1 = 1,$$
$$R_2 : \textbf{If } x_1 \text{ is } A_{21} \textbf{ and } x_2 \text{ is } A_{22} \textbf{ then } g_2 = 2.$$

This model has zero missclassifications and the generated membership functions are close to their idealistic shape as is shown in Fig. 7.

This simple example showed that in certain situations, because of the superior approximation capabilities of fuzzy systems over crisp classifiers, fuzzy models generated based on DTs can be significantly reduced. Therefore, DT-based identification algorithms that simply fuzzify the decision boundaries [13,15,17] does not use the advantages of fuzzy systems in an optimal way.

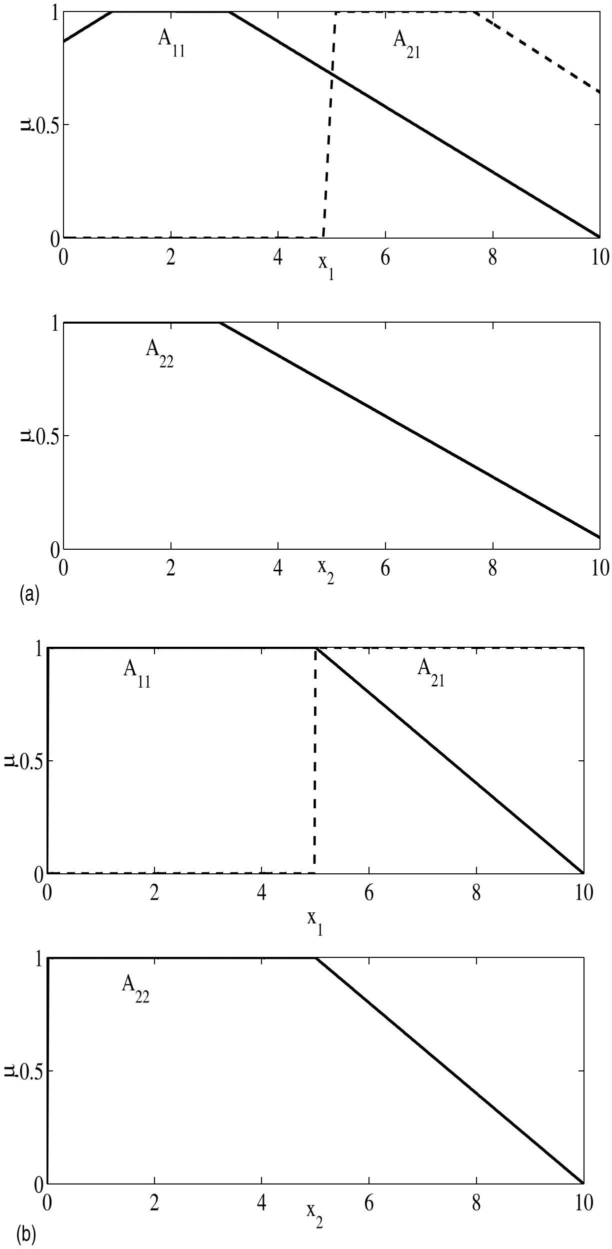

Fig. 7. Membership functions for the geometric classification problem: (a) The obtained membership functions. (b) The idealistic solution.

5.2. Example 2: The Wisconsin Breast Cancer classification problem

The previous case study showed that it is possible to obtain a good rule structure by the proposed rule fuzzification–simplification–optimization procedure. However, the real advantage of the DT-based initialization was not shown. This will be done by the following real classification problem.

The Wisconsin Breast Cancer data (WBCD) is available from the University of California, Irvine (URL: http://www.ics.uci.edu/~mlearn/). The aim of the classification is to distinguish between *benign* and *malignant* cancers based on the available nine measurements: x_1 clump thickness, x_2 uniformity of cell size, x_3 uniformity of cell shape, x_4 marginal adhesion, x_5 single epithelial cell size, x_6 bare nuclei, x_7 bland chromatin, x_8 normal nuclei, and x_9 mitosis (data shown in Fig. 8). The attributes have integer value in the range [1, 10]. The original database contains 699 instances however 16 of these are omitted because these are incomplete, which is common with other studies. The class distribution is 65.5% benign and 34.5% malignant, respectively.

The performance of the classifiers was measured by 10-fold cross validation. The data divided into 10 sub-sets of cases that have similar size and class distributions. Each subset 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 subset.

The advanced version of C4.5 gives missclassification of 5.26% on 10-fold cross validation (94.74% correct classification) with tree size 25 ± 0.5 [23]. An example for such a DT is shown in Fig. 9, where the DT classifier has 7 terminal and 12 internal nodes.

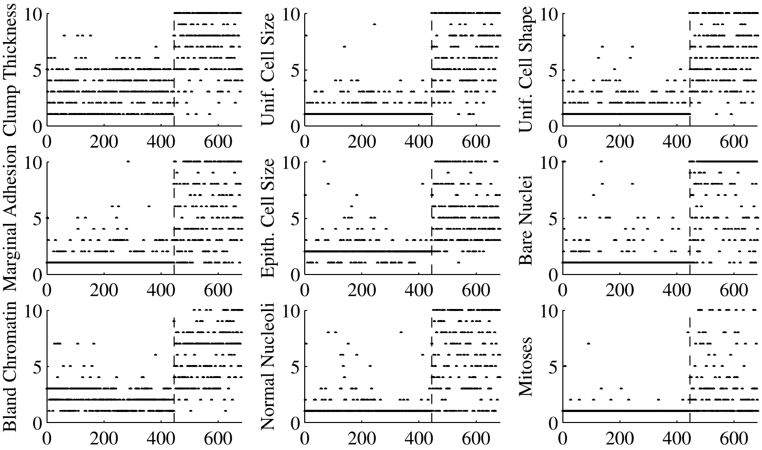

Fig. 8. Wisconsin Breast Cancer data: 2 classes and 9 attributes (Class 1: 1–445, Class 2: 446–683).

```
feature_2 <= 2 :
|   feature_6 <= 3 : 1
|   feature_6 > 3 :
|   |   feature_1 <= 3 : 1
|   |   feature_1 > 3 : 2
feature_2 > 2 :
|   feature_2 > 4 : 2
|   feature_2 <= 4 :
|   |   feature_6 > 2 : 2
|   |   feature_6 <= 2 :
|   |   |   feature_4 <= 3 : 1
|   |   |   feature_4 > 3 : 2
```

Fig. 9. Decision tree generated by C4.5 for the WBCD problem.

The constructed decision trees were transformed into fuzzy models as proposed in Section 3. The number of the fuzzy sets becomes less than the number of the attribute value test of the decision tree because there is more than one interval test for some of the input domains. For instance, the previously presented decision tree (Fig. 9) resulted in a fuzzy model with seven rules and 11 tarapezoidal membership functions.

The model reduction procedure for this initial fuzzy model was started. The first similarity-driven simplification step led to a reduction with four fuzzy sets. In addition, the rules that had a contribution of less than five percent were also deleted. Thereafter the reduced classifier with three rules and four membership functions was optimized with the GA using the objective function given in (12). The obtained classifier was again subjected to the similarity-driven simplification, and the reduced classifier with again one fuzzy sets less was optimized again in 100 GA iterations in the fine-tuning phase. Finally, a very transparent and compact fuzzy model resulted with a recognition rate of 96.5%.

$$R_1 : \textbf{If } x_1 \text{ is } A_{12} \textbf{ and } x_2 \text{ is } A_{16} \textbf{ then } \text{Class} = 1,$$

$$R_2 : \textbf{If } x_1 \text{ is } A_{22} \textbf{ then } \text{Class} = 2.$$

Comparing the fuzzy sets in Fig. 10 with the data in Fig. 8 shows that the obtained rules are highly interpretable.

The 10-fold validation experiment showed 96.82% average classification accuracy, with 94.29% as the worst and 100% as the best performance. This is really good for such a small classifier as compared with previously reported results. The Wisconsin Breast Cancer data are widely used to test the effec-

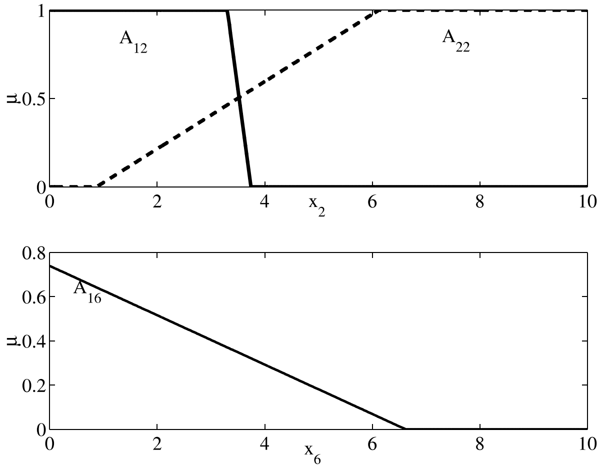

Fig. 10. The resulted membership functions by using the proposed modeling scheme.

tiveness of classification and rule extraction algorithms (Table 1). As the error estimates are either obtained from 10-fold cross validation or from testing the solution once by using the 50% of the data as training set, the results given in Table 1 are only roughly comparable.

Nauck and Kruse [5] combined neuro-fuzzy techniques with interactive strategies for rule pruning to obtain a fuzzy classifier. An initial rule-base was made by applying two sets for each input, resulting in $2^9 = 512$ rules which was reduced to 135 by deleting the non-firing rules. A heuristic data-driven learning method was applied instead of gradient descent learning, which is not applicable for triangular membership functions. Semantic properties were taken into account by constraining the search space. They final fuzzy classifier could be

Table 1
Classification rates and model complexity for classifiers constructed for the Wisconsin Breast Cancer problem

Author	Method	♯ Rules	♯ Conditions	Accuracy
Setiono [25]	NeuroRule 1e	1	4	97.36%
Setiono [25]	NeuroRule 1f	4	4	97.36%
Setiono [25]	NeuroRule 2a	3	11	98.1%
Peña-Reyes and Sipper [24]	Fuzzy-GA1	1	4	97.07%
Peña-Reyes and Sipper [24]	Fuzzy-GA2	3	16	97.36%
Nauck and Kruse [5]	NEFCLASS	2	10–12	95.06% ♮
This paper	DT based FC	2	3-4	96.82% ♮

♮ denotes results from averaging a 10-fold validation.

reduced to two rules with five to six features only, with a misclassification of 4.94% on 10-fold validation (95.06% classification accuracy).

Rule-generating methods that combine GA and fuzzy logic were also applied to this problem [24]. In this method the number of rules to be generated needs to be determined a priori. This method constructs a fuzzy model that has four membership functions and one rule with an additional *else* part. Setiono [25] has generated similar compact classifier by a two-step rule extraction from a feedforward neural network trained on preprocessed data.

As Table 1 shows, our fuzzy rule-based classifier is one of the most compact models in the literature with such high accuracy.

5.3. Example 3: Comparative study

This section is intended to provide a comparative study based on a set of multidimensional classification problem to present how the performance and the complexity of the classifier is changing though the tuning procedure. The chosen Iris, Ionosphere, Glass, Pima and Wine 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 and classes (see Table 2).

During the experiments, the performance of the classifiers were measured by fivefold cross validation. For all classification problems, the initial fuzzy classifier, constructed from a decision tree, was reduced by the presented similarity-driven simplification procedure. Thereafter, the reduced classifier was optimized in 50 GA generations with the GA using the objective function given in Section 4 to enhance performance and similarity. The obtained classifier was again subjected to similarity-driven simplification, and the reduced classifier, was again optimized in 50 GA-iterations. In this step, the distinguishability of the fuzzy sets is preferred ($\lambda < 0$). This step is followed by a fine-tuning phase that consists of 200 GA-iterations ($\lambda > 0$). This model building procedure was monitored by logging the number of the rules, the conditions, and the performance of the classifiers. As Table 3 shows, with the use of the proposed technique, extremely transparent and compact fuzzy classifiers were

Table 2
Complexity of the classification problems

Problem	♯ Samples	♯ Features	♯ Classes
Iris	150	4	3
Ionosphere	351	34	2
Glass	214	9	7
Pima	768	8	2
Wine	178	13	3

Table 3
Classification rates (Acc.) and model complexity (♯ Rules and ♯ Conditions) for the fuzzy (FC) and the initial decision tree (DT) classifiers

Problem	♯ Rules DT	♯ Rules FC	♯ Conditions DT	♯ Conditions FC	Acc. DT	Acc. FM
Iris	4.6	3	7.2	4	95.46%	96.11%
Ionosphere	12.2	3.4	56.6	10.2	91.53%	86.47%
Glass	23	19.2	110.8	90.8	68.32%	66.03%
Pima	24.4	11.2	104.8	40	73.31%	73.05%
Wine	5.6	3.6	14.4	8.8	90.69%	91.22%

Results of fivefold validation.

obtained. During the tuning phase, the number of rules and conditions in the rule-base have been decreased by 50%, while the classification performance has been improved or slightly decreased. This effect is much bigger than the effect of the standard transformation technique [13] to the model complexity and performance.

Concluding, the generated fuzzy classifiers have a comparable performance as those of recently ones, but they are much more simple and transparent [8,26].

6. Conclusions

A decision-tree-based initialization of fuzzy rule-based classifiers is proposed for high-dimensional classification problems. The initial model is derived by means of the C4.5 algorithm which is a crisp binary decision tree algorithm. Contrary to other DT-based initialization methods, an exact transformation technique is applied to obtain the initial fuzzy classifier, which is subsequently reduced and optimized in a iterative scheme by means of similarity-driven rule-reduction and a genetic algorithm with a multiobjective criterion searching for both redundancy and accuracy.

The proposed approach is demonstrated for an artificial problem and the Wisconsin Breast Cancer. Subsequently, a summary of results is given for several classification problems known from literature: Iris, Ionosphere, Glass, Pima, and Wine data. The geometrical classification example demonstrated the superior approximation capabilities of fuzzy systems over crisp classifiers. This indicates that decision-tree-based identification algorithms that fuzzify the decision boundaries and subsequently tune the accuracy by adding rules, do not make optimal use of the fuzzy system structure and lead to unnecessary complex fuzzy classifiers. Moreover, it is shown that a proper rule structure is obtained by the proposed rule-fuzzification, rule-simplification and rule-optimization procedure. The obtained classifier are very compact and well interpretable while the accuracy is still comparable to the best results reported in the literature. The proposed approach could be also used in the regression tree based identification of Takagi–Sugeno fuzzy models, that is one of the topic of our future research.

References

[1] K.J. Cios, W. Pedrycz, R.W. Swiniarski, Data Mining Methods for Knowledge Discovery, Kluwer Academic Publishers, Boston, MA, 1998.

[2] J.V. de Oliveira, Semantic constraints for membership function optimization, IEEE Trans. FS 19 (1999) 128–138.

[3] M. Setnes, R. Babuška, U. Kaymak, H.R. van Nauta Lemke, Similarity measures in fuzzy rule base simplification, IEEE Trans. SMC-B 28 (1998) 376–386.

[4] M. Setnes, J.A. Roubos, Ga-fuzzy modeling and classification: complexity and performance, IEEE Trans. FS 8 (2000) 509–522.

[5] D. Nauck, R. Kruse, Obtaining interpretable fuzzy classification rules from medical data, Artif. Intell. Med. 16 (1999) 149–169.

[6] Y. Jin, Fuzzy modeling of high-dimensional systems, IEEE Trans. FS 8 (2000) 212–221.

[7] J.A. Roubos, M. Setnes, J. Abonyi, Learning fuzzy classification rules from data, in: R. John, R. Birkenhead (Eds.), Developments in Soft Computing, Springer, Berlin, 2001, pp. 108–115.

[8] H. Ishibuchi, T. Nakashima, T. Murata, Performance evaluation of fuzzy classifier systems for multidimensional pattern classification problems, IEEE Trans. SMC–B 29 (1999) 601–618.

[9] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufman, San Mateo, 1993.

[10] L.K. Sethi, Entropy nets: from decision trees to neural networks, Proc. IEEE 78 (1990) 1605–1613.

[11] I. Ivanova, M. Kubat, Initialization of neural networks by means of decision trees, Knowl. Based Syst. 8 (1995) 333–344.

[12] R. Setiono, W. Leow, On mapping decision trees and neural networks, Knowl. Based Syst. 13 (1999) 95–99.

[13] M. Kubat, Decision trees can initialize radial-basis-function networks, IEEE Trans. NN 9 (1998) 813–821.

[14] M. Orr, Combining regression trees and RBFs, International Journal of Neural Systems 10 (6) (2000) 453–465.

[15] J.-S. Jang, C.-T. Sun, Functional equivalence between radial basis function networks and fuzzy inference systems, IEEE Trans. NN 4 (1993) 156–159.

[16] L.T. Kóczy, D. Tikk, T.D. Gedeon, On functional equivalence of certain fuzzy controllers and RBF type approximation schemes, Int. J. Fuzzy Syst. 2 (3) (2000) 164–175.

[17] O. Nelles, M. Fischer, Local linear model trees (LOLIMOT) for nonlinear system identification of a cooling blast, in: European Congress on Intelligent Techniques and Soft Computing (EUFIT), Aachen, Germany, 1996.

[18] J.-S. Jang, Structure determination in fuzzy modeling: A fuzzy CART approach, in: Proceedings of IEEE International Conference on Fuzzy Systems, Orlando, FL, USA, 1994.

[19] F. Hoppner, F. Klawonn, R. Kruse, T. Runkler, Fuzzy Cluster Analysis – Methods for Classification, Data Analysis and Image Recognition, Wiley, New York, 1999.

[20] M. Setnes, J.A. Roubos, Transparent fuzzy modeling using fuzzy clustering and GA's, in: NAFIPS, New York, USA, 1999, pp. 198–202.

[21] J.A. Roubos, M. Setnes, Compact fuzzy models through complexity reduction and evolutionary optimization, in: Proceedings of IEEE International Conference on Fuzzy Systems, San Antonio, USA, 2000, pp. 762–767.

[22] J.A. Roubos, M. Setnes, Compact and transparent fuzzy models and classifiers through iterative complexity reduction, IEEE Trans. Fuzzy Syst. 9 (4) (2001) 516–524.

[23] J.R. Quinlan, Improved use of continuous attributes in C4.5, J. Artif. Intell. Res. 4 (1996) 77–90.

[24] C.A. Peña-Reyes, M. Sipper, A fuzzy genetic approach to breast cancer diagnosis, Artif. Intell. Med. 17 (2000) 131–155.

[25] R. Setiono, Generating concise and accurate classification rules for breast cancer diagnosis, Artif. Intell. Med. 18 (2000) 205–219.

[26] O. Cordon, F.H.M.J. Jesus, A proposal on reasoning methods in fuzzy rule-based classification systems, Int. J. Approx. Reason. 20 (1999) 21–45.