# A Fuzzy Association Rule-Based Classification Model for High-Dimensional Problems With Genetic Rule Selection and Lateral Tuning

Jesús Alcalá-Fdez, Rafael Alcalá, and Francisco Herrera, *Member, IEEE*

*Abstract*—The inductive learning of fuzzy rule-based classification systems suffers from exponential growth of the fuzzy rule search space when the number of patterns and/or variables becomes high. This growth makes the learning process more difficult and, in most cases, it leads to problems of scalability (in terms of the time and memory consumed) and/or complexity (with respect to the number of rules obtained and the number of variables included in each rule). In this paper, we propose a fuzzy association rule-based classification method for high-dimensional problems, which is based on three stages to obtain an accurate and compact fuzzy rule-based classifier with a low computational cost. This method limits the order of the associations in the association rule extraction and considers the use of subgroup discovery, which is based on an improved weighted relative accuracy measure to preselect the most interesting rules before a genetic postprocessing process for rule selection and parameter tuning. The results that are obtained more than 26 real-world datasets of different sizes and with different numbers of variables demonstrate the effectiveness of the proposed approach.

*Index Terms*—Associative classification, classification, data mining, fuzzy association rules, genetic algorithms (GAs), genetic fuzzy rule selection, high-dimensional problems.

## I. INTRODUCTION

**F**UZZY rule-based classification systems (FRBCSs) [1], [2] are useful and well-known tools in the machine learning framework, since they can provide an interpretable model for the end user [3]–[6]. There are many real applications in which FRBCSs have been employed, including anomaly intrusion detection [7], image processing [8], among others. In most of these areas, the available or useful data consist of a high number of patterns (instances or examples) and/or variables. In this situation, the inductive learning of FRBCSs suffers from exponential growth of the fuzzy rule search space. This growth makes the learning process more difficult, and in most cases, it leads to problems of scalability (in terms of the time and memory consumed) and/or complexity (with respect to the number of rules obtained and the number of variables included in each rule) [9], [10].

Association discovery is one of the most common data mining techniques that are used to extract interesting knowledge from large datasets [11]. Much effort has been made to use its advantages for classification under the name of associative classification [12]–[19]. Association discovery aims to find interesting relationships between the different items in a database [20], while classification aims to discover a model from training data that can be used to predict the class of test patterns [21]. Both association discovery and classification rules mining are essential in practical data mining applications [11], [22], and their integration could result in greater savings and convenience for the user.

A typical associative classification system is constructed in two stages:

1) discovering the association rules inherent in a database;
2) selecting a small set of relevant association rules to construct a classifier.

In order to enhance the interpretability of the obtained classification rules and to avoid unnatural boundaries in the partitioning of the attributes, different studies have been presented to obtain classification systems, which is based on fuzzy association rules [23]–[28]. For instance, in [24], the authors have made use of a genetic algorithm (GA) [29], [30] to automatically determine minimum support and confidence thresholds, mining for each chromosome a fuzzy rule set for classification by means of an algorithm, which is based on the Apriori algorithm [31], and adjusting the fuzzy confidence of these rules with the approach that was proposed by Nozaki *et al.* in [32]. Consequently, this approach can only be used for small problems since its computational cost is very high when we consider problems that consist of a high number of patterns and/or variables. On the other hand, in [25], the authors used an algorithm that is based on the Apriori algorithm to mine association rules only up to a certain level and to select the $K$ most confident ones for each class among them, in order to finally employ a genetic rule-selection method that obtains a classifier from them. However, many patterns may be uncovered if we only consider the confidence measure to select the candidate rules.

In this paper, we present a fuzzy association rule-based classification method for high-dimensional problems (FARC-HD) to obtain an accurate and compact fuzzy rule-based classifier with a low computational cost. This method is based on the following three stages:

The authors are with the Department of Computer Science and Artificial Intelligence, Research Center on Information and Communications Technology (CITIC-UGR), University of Granada, 18071 Granada, Spain (e-mail: jalcala@decsai.ugr.es; alcala@decsai.ugr.es; herrera@decsai.ugr.es).

1) *Fuzzy association rule extraction for classification*: A search tree is employed to list all possible frequent fuzzy item sets and to generate fuzzy association rules for classification, limiting the depth of the branches in order to find a small number of short (i.e., simple) fuzzy rules.

2) *Candidate rule prescreening*: Even though the order of the associations is limited in the association rule extraction, the number of rules generated can be very large. In order to decrease the computational cost of the genetic postprocessing stage, we consider the use of subgroup discovery based on an improved weighted relative accuracy measure (wWRAcc') to preselect the most interesting rules by means of a pattern weighting scheme [33].

3) *Genetic rule selection and lateral tuning*: Finally, we make use of GAs to select and tune a compact set of fuzzy association rules with high classification accuracy in order to consider the known positive synergy that both techniques present (selection and tuning). Several works have successfully combined the selection of rules with the tuning of membership functions (MFs) within the same process [34], [35], taking advantage of the possibility of different coding schemes that GAs provide. The successful application of GAs to identify fuzzy systems has led to the so-called genetic fuzzy systems (GFSs) [36]–[38].

In order to assess the performance of the proposed approach, we have used 26 real-world datasets with a number of variables ranging from 4 to 90 and a number of patterns ranging from 150 to 19 020. We have developed the following studies. First, we have shown the results that are obtained from comparison with three other GFSs [38]. Second, we have compared the performance of our approach with two approaches to obtain fuzzy associative classifiers. Third, we have shown the results that are obtained from the comparison with four other classical approaches for associative classification and with the C4.5 decision tree [39]. Furthermore, in these studies, we have made use of some nonparametric statistical tests for pairwise and multiple comparison [40]–[43] of the performance of these classifiers. Then, we have shown a study on the influence of the depth of the trees and the number of evaluations in the genetic selection and tuning process. Finally, we have analyzed the scalability of the proposed approach.

This paper is arranged as follows. Section II introduces the type of rules, rule weights, and inference model, which are used, and the basic definitions for fuzzy association rules and associative classification. Section III describes in detail each stage of the proposed approach. Section IV presents the experimental setup. Section V shows and discusses the results that are obtained on 26 real-world datasets. Finally, in Section VI, some concluding remarks are made.

## II. PRELIMINARIES

In this section, we first describe FRBCSs. Then, we introduce the basic definitions for fuzzy association rules. Finally, we present fuzzy association rules for classification.

### A. Fuzzy Rule-Based Classification Systems

Any classification problem consists of $N$ training patterns, i.e., $x_p = (x_{p1}, \ldots, x_{pm})$, $p = 1, 2, \ldots, N$, from $S$ classes, where $x_{pi}$ is the $i$th attribute value ($i = 1, 2, \ldots, m$) of the $p$th training pattern. In this paper, we use fuzzy rules of the following form for our classifier:

$$\text{Rule } R_j: \text{IF } x_1 \text{ is } A_{j1} \text{ and } \cdots \text{ and } x_m \text{ is } A_{jm}$$

$$\text{THEN Class } = C_j \text{ with } \text{RW}_j$$

where $R_j$ is the label of the $j$th rule, $x = (x_1, \ldots, x_m)$ is an $m$-dimensional pattern vector, $A_{ji}$ is an antecedent fuzzy set, $C_j$ is a class label, and $\text{RW}_j$ is the rule weight.

The rule weight of each fuzzy rule $R_j$ has a great effect on the performance of fuzzy rule-based classifiers [44]. Different specifications of the rule weight have been proposed and examined in the literature. In [45], we can find some heuristic methods for rule weight specification. In this paper, we employ the most common one, i.e., the fuzzy confidence value or certainty factor (CF) [46]:

$$\text{RW}_j = \text{CF}_j = \frac{\sum_{x_p \in \text{Class} C_j} \mu_{A_j}(x_p)}{\sum_{p=1}^{N} \mu_{A_j}(x_p)} \qquad (1)$$

where $\mu_{A_j}(x_p)$ is the matching degree of the pattern $x_p$ with the antecedent part of the fuzzy rule $R_j$. We use the fuzzy reasoning method of the weighted vote or additive combination [46] to classify new patterns by the rule base (RB). With this method, each fuzzy rule casts a vote for its consequent class. The total strength of the vote for each class is computed as follows:

$$V_{\text{Class}_h}(x_p) = \sum_{R_j \in \text{RB}; \; C_j = h} \mu_{A_j}(x_p) \cdot \text{CF}_j$$

$$h = 1, 2, \ldots, S, \qquad R_j \in \text{RB}. \qquad (2)$$

The new pattern $x_p$ is classified as the class with the maximum total strength of the vote. If multiple class labels have the same maximum value for $x_p$ or no fuzzy rule is compatible with $x_p$, this pattern is classified as the class with most patterns in the training data.

### B. Fuzzy Association Rules

Association rules are used to represent and identify dependences between items in a database [11], [20]. They are expressions of the type $A \rightarrow B$, where $A$ and $B$ are sets of items, and $A \cap B = \oslash$. This means that if all the items in $A$ exist in a transaction, then all the items in $B$ with a high probability are also in the transaction, and $A$ and $B$ should have no common items [31]. There are many previous studies to mine association rules that are focused on databases with binary or discrete values; however, data in real-world applications usually consist of quantitative values. Designing data mining algorithms, which are able to deal with various types of data, presents a challenge to workers in this research field.

Fuzzy set theory has been used more and more frequently in data mining because of its simplicity and similarity to human reasoning [1]. The use of fuzzy sets to describe associations
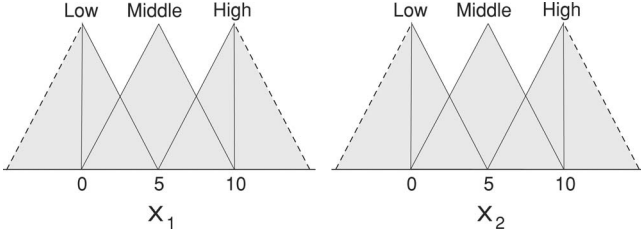
Fig. 1. Attributes and linguistic terms for the attributes $X_1$ and $X_2$.

between data extends the types of relationships that may be represented, facilitates the interpretation of rules in linguistic terms, and avoids unnatural boundaries in the partitioning of the attribute domains. For this reason, in recent years, different studies have proposed methods to mine fuzzy association rules from quantitative data [47]–[54].

Let us consider a simple database $T$ with two attributes ($X_1$ and $X_2$) and three linguistic terms with their associated MFs (see Fig. 1). Based on this definition, a simple example of fuzzy association rule is that $X_1$ is Middle $\rightarrow X_2$ is High.

Support and confidence are the most common measures of interest of an association rule. These measures can be defined for fuzzy association rules as follows:

$$\text{Support}(A \rightarrow B) = \frac{\sum_{x_p \in T} \mu_{AB}(x_p)}{|N|} \quad (3)$$

$$\text{Confidence}(A \rightarrow B) = \frac{\sum_{x_p \in T} \mu_{AB}(x_p)}{\sum_{x_p \in T} \mu_A(x_p)} \quad (4)$$

where $|N|$ is the number of transactions in $T$, $\mu_A(x_p)$ is the matching degree of the transaction $x_p$ with the antecedent part of the rule, and $\mu_{AB}(x_p)$ is the matching degree of the transaction $x_p$ with the antecedent and consequent of the rule.

### C. Fuzzy Association Rules for Classification

Over the past few years, different studies have proposed methods to obtain fuzzy association rule-based classifiers [23]–[28]. The task of classification is to find a set of rules in order to identify the classes of undetermined patterns. A fuzzy association rule can be considered to be a classification rule if the antecedent contains fuzzy item sets, and the consequent part contains only one class label ($C = \{C_1, \ldots, C_j, \ldots, C_S\}$). A fuzzy associative classification rule, i.e., $A \rightarrow C_j$, could be measured directly in terms of support and confidence as follows:

$$\text{Support}(A \rightarrow C_j) = \frac{\sum_{x_p \in \text{Class}C_j} \mu_A(x_p)}{|N|} \quad (5)$$

$$\text{Confidence}(A \rightarrow C_j) = \frac{\sum_{x_p \in \text{Class}C_j} \mu_A(x_p)}{\sum_{x_p \in T} \mu_A(x_p)}. \quad (6)$$

## III. FUZZY ASSOCIATION RULE-BASED CLASSIFIER FOR HIGH-DIMENSIONAL PROBLEMS

In this section, we will describe our proposal to obtain a fuzzy association rule-based classifier for high-dimensional problems. This method is based on the following three stages:

1) *Fuzzy association rule extraction for classification*: A search tree is employed to list all the possible frequent fuzzy item sets and to generate fuzzy association rules for classification.
2) *Candidate rule prescreening*: A rule evaluation criterion is used to preselect candidate fuzzy association rules.
3) *Genetic rule selection and lateral tuning*: The best cooperative rules are selected and tuned by means of a GA, considering the positive synergy between both techniques within the same process.

Finally, we add a default rule considering the class with the most patterns in the training data. In the following, we will introduce the three mentioned stages, which explain in detail all their characteristics (see Sections III-A–C and present a flowchart of the algorithm (see Section III-D).

### A. Stage 1. Fuzzy Association Rule Extraction for Classification

To generate the RB, we employ a search tree to list all the possible fuzzy item sets of a class. The root or level 0 of a search tree is an empty set. All attributes are assumed to have an order (in our case, the order of appearance in the training data), and the one-item sets that correspond to the attributes are listed in the first level of the search tree according to their order. If an attribute has $j$ possible outcomes ($q_j$ linguistic terms for each quantitative attribute), it will have $j$ one-item sets that are listed in the first level. The children of a one-item node for an attribute $A$ are the two-item sets that include the one-item set of attribute $A$ and a one-item set for another attribute behind attribute $A$ in the order, and so on. If an attribute has $j > 2$ possible outcomes, it can be replaced by $j$ binary variables to ensure that no more than one of these $j$ binary attributes can appear in the same node in a search tree. An example with two attributes $V_1$ and $V_2$ with two linguistic terms $L$ and $H$ is detailed in Fig. 2.

An item set with a support higher than the minimum support is a frequent item set. If the support of an $n$-item set in a node $J$ is less than the minimum support, it does not need to be extended more because the support of any item set in a node in the subtree, which is led by node $J$, will also be less than the minimum support. Likewise, if a candidate item set generates a classification rule with confidence higher than the maximum confidence, this rule has reached the quality level that is demanded by the user, and it is again unnecessary to extend it further. These properties greatly reduce the number of nodes needed for searching.

The fuzzy support of an item set can be calculated as follows:

$$\text{Support}(A) = \frac{\sum_{x_p \in T} \mu_A(x_p)}{|N|} \quad (7)$$

where $\mu_A(x_p)$ is the matching degree of the pattern $x_p$ with the item set. The matching degree $\mu_A(x_p)$ of $x_p$ to the different fuzzy regions is computed by the use of a conjunction operator, in our case, the product $T$-norm.

Once all frequent fuzzy item sets have been obtained, the candidate fuzzy association rules for classification can be generated, setting the frequent fuzzy item sets in the antecedent of the rules and the corresponding class in the consequent. This process is repeated for each class.
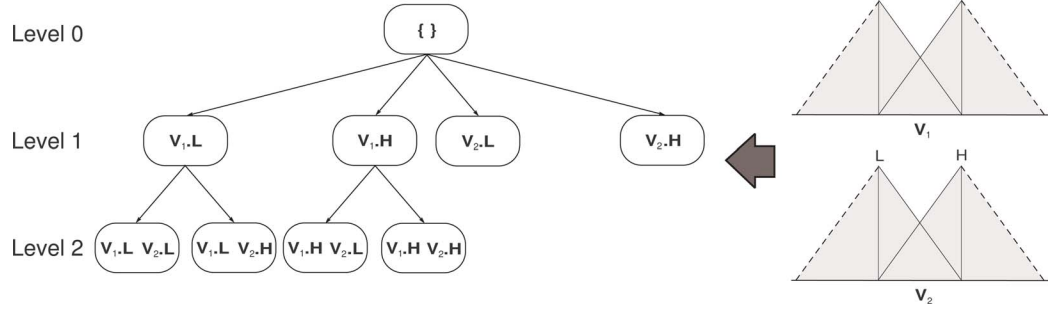
Fig. 2. Search tree for two quantitative attributes $V_1$ and $V_2$ with two linguistic terms $L$ and $H$.

The number of frequent fuzzy item sets that are extracted depends directly on the minimum support. The minimum support is usually calculated considering the total number of patterns in the dataset; however, the number of patterns for each class in a dataset can be different. For this reason, our algorithm determines the minimum support of each class by the distributions of the classes over the dataset. Thus, the minimum support for class $C_j$ is defined as

$$\text{MinimumSupport}_{C_j} = \text{minSup} * f_{C_j} \quad (8)$$

where minSup is the minimum support determined by the expert, and $f_{C_j}$ is the pattern ratio of the class $C_j$.

In this stage, we can generate a large number of candidate fuzzy association rules for classification. It is, however, very difficult for human users to handle such a large number of generated fuzzy rules and to intuitively understand long fuzzy rules with many antecedent conditions. For this reason, we only generate short fuzzy rules and with only a small number of antecedent conditions. Thus, the depth of the trees is limited to a fixed value $\text{Depth}_{\max}$ that is determined by an expert.

### B. Stage 2. Candidate Rule Prescreening

In the previous stage, we can generate a large number of candidate rules. In order to decrease the computational costs of stage 3, we consider the use of subgroup discovery to preselect the most interesting rules from the RB, which are obtained in the previous stage by means of a pattern weighting scheme [33]. This scheme treats the patterns in such a way that covered positive patterns are not deleted when the current best rule is selected. Instead, each time a rule is selected, the algorithm stores a count $i$ for each pattern of how many times (with how many of the selected rules) the pattern has been covered.

Weights of positive patterns covered by the selected rule decrease according to the formula $w(e_j, i) = \frac{1}{i+1}$. In the first iteration, all target class patterns are assigned the same weight, i.e., $w(e_j, 0) = 1$, while in the following iterations the contributions of patterns are inversely proportional to their coverage by previously selected rules. This way, the patterns that are already covered by one or more selected rules decrease their weights while uncovered target class patterns whose weights have not been decreased will have a greater chance of being covered in the following iterations. Covered patterns are completely eliminated when they have been covered more than $k_t$ times.

TABLE I
FIVE PATTERNS IN THIS EXAMPLE

| ID | $X_1$ | $X_2$ | Class | Weight |
|----|-------|-------|-------|--------|
| ID1 | 0.0 | 10.0 | $C_1$ | 1.0 |
| ID2 | 2.5 | 4.0 | $C_2$ | 1.0 |
| ID3 | 3.2 | 1.0 | $C_2$ | 0.0 |
| ID4 | 9.0 | 5.0 | $C_2$ | 1.0 |
| ID5 | 2.5 | 10 | $C_1$ | 0.5 |

Thus, in each iteration of the process, the rules are ordered according to a rule evaluation criterion from best to worst. The best rule is selected, covered patterns are reweighted, and the procedure repeats these steps until one of the stopping criteria is satisfied: either all patterns have been covered more than $k_t$ times, or there are no more rules in the RB. This process is to be repeated for each class.

wWRAcc′ was used to evaluate the quality of intervalar rules in APRIORI-SD [33]. This measure was defined as follows:

$$\text{wWRAcc}'(A \rightarrow C_j) = \frac{n'(A)}{N'} \cdot \left( \frac{n'(A \cdot C_j)}{n'(A)} - \frac{n(C_j)}{N} \right) \quad (9)$$

where $N'$ is the sum of the weights of all patterns, $n'(A)$ is the sum of the weights of all covered patterns, $n'(A \cdot C_j)$ is the sum of the weights of all correctly covered patterns, $n(C_j)$ is the number of patterns of class $C_j$, and $N$ is the number of all patterns. For instance, let us consider a simple database with two attributes $X_1$ and $X_2$, two classes $C_1$ and $C_2$, and five training patterns. Table I shows the five training patterns and their weights in the $p$th iteration of the process. In this iteration, the wWRAcc′ value of a simple rule, i.e, $R = $ IF $X_1$ is $[0.0, 5.0[$ and $X_2$ is $[5.0, 10.0] \rightarrow C_1$, is calculated as follows:

$$\text{wWRAcc}'(R) = \frac{1.0 + 0.5}{1.0 + 1.0 + 0.0 + 1.0 + 0.5} \cdot \left( \frac{1.0 + 0.5}{1.0 + 0.5} - \frac{2}{5} \right)$$
$$= 0.257.$$

We have modified this measure to enable the handling of fuzzy rules. The new measure is defined as follows:

$$\text{wWRAcc}''(A \rightarrow C_j) = \frac{n''(A \cdot C_j)}{n'(C_j)} \cdot \left( \frac{n''(A \cdot C_j)}{n''(A)} - \frac{n(C_j)}{N} \right) \quad (10)$$

where $n''(A)$ is the sum of the products of the weights of all covered patterns by their matching degrees with the antecedent part of the rule, $n''(A \cdot C_j)$ is the sum of the products of the weights of all correctly covered patterns by their matching degrees with the antecedent part of the rules, and $n'(C_j)$ is the sum of the weights of patterns of class $C_j$. Moreover, the first term in the definition of wWRAcc$'$ has been replaced by $\frac{n''(A \cdot C_j)}{n(C_j)}$ to reward rules that cover unelimated patterns of class $C_j$.

Let us consider three linguistic terms for the attributes $X_1$ and $X_2$ (see Fig. 1). Based on this definition, a simple example of fuzzy association rule for classification is: $R = $ If $X_1$ is Low and $X_2$ is High $\rightarrow C_1$. This rule covers the training patterns in Table I with degrees (ID1, 1.0), (ID2, 0.0), (ID3, 0.0), (ID4, 0.0), and (ID5, 0.5). In this situation, the wWRAcc$''$ value of this rule is calculated as follows:

$$\text{wWRAcc}''(R) = \frac{1.0 * 1.0 + 0.5 * 0.5}{1.0 + 0.5}$$
$$\cdot \left( \frac{1.0 * 1.0 + 0.5 * 0.5}{1.0 * 1.0 + 0.5 * 0.5} - \frac{2}{5} \right) = 0.5.$$

This measure can obtain positive or negative values in the interval $[-1.0, 1.0]$. A rule with a wWRAcc$''$ value near to 1 may be more useful for the classification.

### C. Stage 3. Rule Selection and Lateral Tuning

We consider the use of GAs to select and tune a compact set of fuzzy association rules with high classification accuracy from the RB, which are obtained in the previous stage. We consider the approach that is proposed in [35], where rules are based on the linguistic two-tuple representation [55]. This representation allows the lateral displacement of the labels considering only one parameter (symbolic translation parameter), which involves a simplification of the tuning search space that eases the derivation of optimal models, particularly, when it is combined with a rule selection within the same process enabling it to take advantage of the positive synergy that both techniques present. This way, this process to contextualize the MFs enables them to achieve a better covering degree while maintaining the original shapes, which results in accuracy improvements without a significant loss in the interpretability of the fuzzy labels. The symbolic translation parameter of a linguistic term is a number within the interval $[-0.5, 0.5)$ that expresses the domain of a label when it is moving between its two lateral labels. Let us consider a set of labels $S$ representing a fuzzy partition. Formally, we have the pair $(S_i, \alpha_i)$, $S_i \in S$, $\alpha_i \in [-0.5, 0.5)$. An example is illustrated in Fig. 3, where we show the symbolic translation of a label that is represented by the pair $(S_2, -0.3)$.

Let us consider the simple problem presented in the previous section. Based on this definition, examples of classic rule and linguistic two-tuple represented rule are as follows.

*Classic Rule*:

$$\text{IF } X_1 \text{ is Low and } X_2 \text{ is Middle}$$
$$\text{THEN Class is } C_1$$



Fig. 3. Symbolic translation of a linguistic label and lateral displacement of the involved MF. (a) Simbolic translation of a linguistic term. (b) Lateral displacement of an MF.

*Two-Tuple Representation*:

$$\text{IF } X_1 \text{ is (Low, 0.1) and } X_2 \text{ is (Middle, } -0.3)$$
$$\text{THEN Class is } C_1.$$

In [35], two different rule representation approaches were proposed: a global approach and a local approach. In our particular case, the tuning is applied to the level of linguistic partitions (global approach). This way, the pair $(X_i, \text{label})$ takes the same $\alpha$ value in all the rules where it is considered, i.e., a global collection of two tuples is considered by all the fuzzy rules. For example, $X_1$ is (High, 0.3) that will present the same value for those rules in which the pair "$X_1$ is High" was initially considered. This proposal decreases the tuning problem complexity, greatly easing the derivation of optimal models. Another important issue is that from the parameters $\alpha$ that are applied to each label, we could obtain the equivalent triangular MFs, by which an FRBCS that is based on linguistic two tuples could be represented as a classical Mamdani FRBCS. Notice that the class label and RW of the rule are not modified.

In the following, the main characteristics of the genetic approach that combines rule selection and lateral tuning are presented: genetic model, codification and initial gene pool, chromosome evaluation, crossover operator, and restarting approach.

*1) CHC Genetic Model:* The approach that is proposed in [35] considers the use of a specific GA, the CHC algorithm [56].

Fig. 4. Scheme of the CHC algorithm.

The CHC algorithm is a GA that presents a good trade-off between exploration and exploitation, making it a good choice for problems with complex search spaces [57]. This genetic model makes use of a mechanism of selection of populations in order to perform an adequate global search. $P$ parents and their corresponding offspring compete to select the best $P$ individuals to take part in the next populatio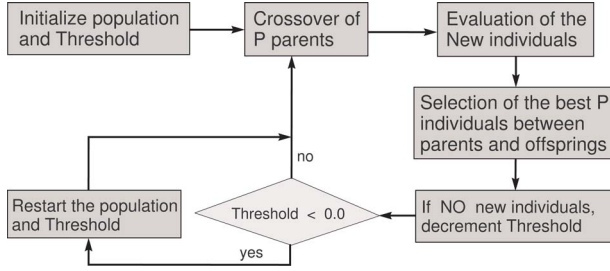n. The CHC approach makes use of an incest prevention mechanism and a restarting process to encourage diversity in the population, instead of the well-known mutation operator.

This incest prevention mechanism will be considered in order to apply the crossover operator, i.e., two parents are crossed if their hamming distance divided by 2 is more than a predetermined threshold $L$. This threshold value is initialized as the maximum possible distance between two individuals (the number of genes in the chromosome) divided by 4. Following the original CHC scheme, $L$ is decremented by 1 when there are no new individuals in the population in one generation. In order to make this procedure independent of the number of genes in the chromosome, in our case, $L$ will be decremented by $\varphi\%$ of its initial value (with $\varphi$ determined by the user, usually 10%). When $L$ is below zero, the algorithm restarts the population (for more information, see [58]).

A scheme of this algorithm is shown in Fig. 4.

*2) Codification and Initial Gene Pool:* To combine the rule selection with the global lateral tuning, a double coding scheme for both rule selection $C_S$ and lateral tuning $C_T$ is used

1) For the $C_S$ part, each chromosome is a binary vector that determines when a rule is selected or not (alleles "1" and "0," respectively). Considering the $M$ rules that are contained in the candidate rule set, the corresponding part, i.e., $C_S = \{c_1, \ldots, c_M\}$, represents a subset of rules composing the final RB so that IF $c_i = 1$ THEN ($R_i \in$ RB) *else* ($R_i \notin RB$), with $R_i$ being the corresponding $i$th rule in the candidate rule set and RB being the final RB.

2) For the $C_T$ part, a real coding is considered. This part is the joint of the $\alpha$ parameters of each fuzzy partition. Let us consider the following number of labels per variable: $(m^1, m^2, \ldots, m^n)$ with $n$ being the number of system variables. Then, this part has the following form, where each gene is associated with the tuning value of the corresponding label: $C_T = (c_{11}, \ldots, c_{1m^1},$ $c_{21}, \ldots, c_{2m^2}, \ldots, c_{n1}, \ldots, c_{nm^n})$.

Finally, a chromosome $C$ is coded in the following way: $C = C_S C_T$. To make use of the available information, all the candidate rules are included in the population as an initial solution. To do this, the initial pool is obtained with the first individual having all genes with value "1" in the $C_S$ part and all genes with value "0.0" in the $C_T$ part. The remaining individuals are generated at random.

*3) Chromosome Evaluation:* To evaluate a determined chromosome penalizing a large number of rules, we compute the classification rate and the following function is maximized:

$$\text{Fitness}(C) = \frac{\#\text{Hits}}{N} - \delta \cdot \frac{\text{NR}_{\text{initial}}}{\text{NR}_{\text{initial}} - \text{NR} + 1.0} \qquad (11)$$

where #Hits is the number of patterns that are correctly classified (see Section II-C), $\text{NR}_{\text{initial}}$ is the number of candidate rules, NR is the number of selected rules, and $\delta$ is a weighting percentage given by the system expert that determines the tradeoff between accuracy and complexity. If there is at least one class without selected rules or if there are no covered patterns, the fitness value of a chromosome will be penalized with the number of classes without selected rules and the number of uncovered patterns.

*4) Crossover Operator:* The crossover operator will depend on the chromosome part where it is applied.

1) For the $C_T$ part, we consider the Parent Centric BLX (PCBLX) operator [59] (an operator that is based on BLX-$\alpha$). This operator is based on the concept of neighborhood, which allows the offspring genes to be around the genes of one parent or around a wide zone that is determined by both parent genes. Let us assume that $X = (x_1, \ldots, x_n)$, and $Y = (y_1, \ldots, y_n)$, where $x_i, y_i \in [a_i, b_i] \subset \Re, i = 1, \ldots, n$, are two real-coded chromosomes that are going to be crossed. We generate the following two offspring.

a) $O_1 = (o_{11} \cdots o_{1n})$, where $o_{1i}$ is a randomly (uniformly) chosen number from the interval $[l_i^1, u_i^1]$, with $l_i^1 = \max\{a_i, x_i - I_i \cdot \alpha\}$, $u_i^1 = \min\{b_i, x_i + I_i \cdot \alpha\}$, and $I_i = |x_i - y_i|$.

b) $O_2 = (o_{21} \cdots o_{2n})$, where $o_{2i}$ is a randomly (uniformly) chosen number from the interval $[l_i^2, u_i^2]$, with $l_i^2 = \max\{a_i, y_i - I_i \cdot \alpha\}$, and $u_i^2 = \min\{b_i, y_i + I_i \cdot \alpha\}$.

2) In the $C_S$ part, the half-uniform crossover scheme (HUX) is employed [58]. The HUX crossover exactly interchanges the mid of the alleles that are different in the parents (the genes to be crossed are randomly selected from among those that are different in the parents). This operator ensures the maximum distance of the offspring to their parents (exploration).

In this case, four offspring are generated by the combination of the two from the part $C_T$ with the two from the part $C_S$. The two best offspring obtained in this way are considered as the two corresponding descendents.

Notice that since we consider a real coding scheme for the $C_T$ part, the incest prevention mechanism has to transform each gene considering a Gray code (binary code) with a fixed number of bits per gene (BITSGENE) that is determined by the expert
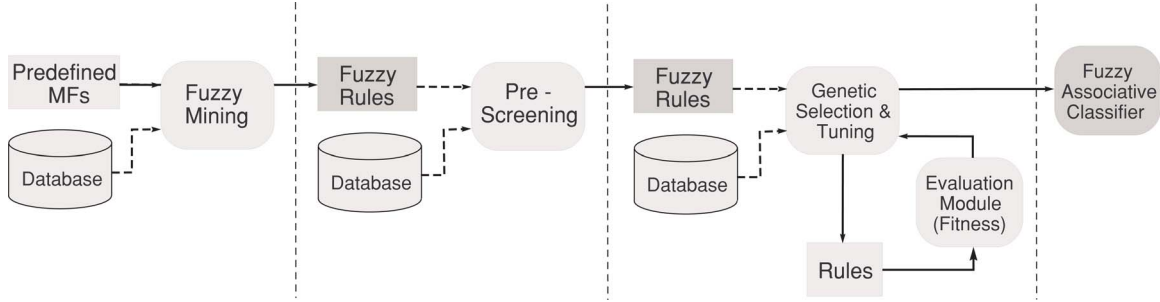
Fig. 5. Scheme of the FARC-HD method.

to calculate the hamming distance between two individuals in order to apply the crossover operators.

*5) Restarting Approach:* To get away from local optima, this algorithm uses a restart approach. In this case, the best chromosome is maintained, and the remaining are generated at random. The restart procedure is applied when the threshold value $L$ is below zero, which means that all the individuals coexisting in the population are very similar.

### D. Flowchart

In accordance with the previous description, the proposed algorithm to obtain a fuzzy association rule-based classifier is described in the following.

INPUT: A dataset with size $T$ and $m$ attributes, each with $q_j$ predefined linguistic terms.

OUTPUT: A fuzzy associative classifier.

*Stage 1. Fuzzy Association Rule Extraction for Classification. For each class $C_j$:*

*Step 1:* Calculate the minimum support of class $C_j$ according to (9).

*Step 2:* Create the levels 0 and 1 of the tree.

*Step 3:* Create a new level in the tree.

*Step 4:* Prune nodes.

*Step 5:* If there are more than two nodes in the new level, and the depth of the tree is less than $\text{Depth}_{\max}$, go to Step 3.

*Step 6:* Generate the rules with class $C_j$ on the right-hand side.

Stage 2. *Candidate Rule Prescreening.*

*For each class $C_j$:*

*Step 7:* Set the weight of the patterns as 1.

*Step 8:* Calculate the wWRAcc″ value for each rule.

*Step 9:* Select the best rule as a part of the initial RB for Stage 3 and remove it from the candidate rule set.

*Step 10:* Decrease the weight of the patterns covered by the selected rule.

*Step 11:* If any pattern has been covered less than $k_t$ times and there are more rules in the candidate rule set, go to Step 8.

Stage 3. *Rule Selection and Lateral Tuning.*

*Step 12:* Generate the initial population with $P$ chromosomes.

*Step 13:* Evaluate the population.

*Step 14:* Initialize the threshold value taking into account Gray codings, i.e., $L = L_{\text{initial}}$.

*Step 15:* Generate the next population as following.

1) Shuffle the population.

2) Select the parents two by two. Each pair is crossed if the hamming distance between the parent Gray codings divided by 2 is more than $L$.

3) Evaluate the new individuals.

4) Join the parents with their offspring, and select the best $P$ individuals to take part in the next population.

*Step 16:* If the best chromosome does not change or there are no new individuals in the population, then $L = L - (L_{\text{initial}} * 0.1)$.

*Step 17:* If $L < 0$, restart the population and initialize $L$.

*Step 18:* If the maximum number of evaluations is not reached, go to Step 15.

A scheme of this algorithm is shown in Fig. 5.

## IV. EXPERIMENTAL SETUP

Several experiments have been carried out in this paper to evaluate the usefulness of our proposal. In the following, first, we describe the real-world databases that are used in these experiments; second, we introduce a brief description of the methods considered for comparison; third, we show the configuration of the methods (determining all the parameters used); and finally, we describe the statistical analysis that is adopted in this study.

### A. Datasets

In order to analyze the performance of the proposed approach, we have considered 26 real-world datasets. Table II summarizes the main characteristics of the 26 datasets and shows the link to the Knowledge Extraction based on Evolutionary Learning (KEEL)-dataset repository [60] from which they can be downloaded, where "Attributes$(R/I/N)$" is the number of (Real/Integer/Nominal) attributes in the data, "Patterns" is the number of patterns, and "Classes" is the number of classes. Notice that we have removed the instances with any missing value in the datasets (Cleveland and Crx), and 12 datasets have a number of variables greater than or equal to 15.

To develop the different experiments, we consider a *tenfold cross-validation model*, i.e., we randomly split the dataset into ten folds, each containing 10% of the patterns of the dataset, and use nine folds for training and one for testing.[1] For each of the ten partitions, we executed three trials of the algorithms.

---

[1]The corresponding data partitions (tenfold) for these datasets are available at the KEEL-dataset repository [60]: http://sci2s.ugr.es/keel/datasets.php

TABLE II
DATASETS CONSIDERED FOR THE EXPERIMENTAL STUDY

| Name | Attributes (R/I/N) | Patterns | Classes | Name | Attributes (R/I/N) | Patterns | Classes |
|---|---|---|---|---|---|---|---|
| Iris | 4 (4/0/0) | 150 | 3 | Vowel | 13 (10/3/0) | 990 | 11 |
| Phoneme | 5 (5/0/0) | 5404 | 2 | Crx | 15 (3/3/9) | 653(690) | 2 |
| Monks | 6 (0/6/0) | 432 | 2 | Pen-based | 16 (0/16/0) | 10992 | 10 |
| Appendicitis | 7 (7/0/0) | 106 | 2 | German | 20 (0/7/13) | 1000 | 2 |
| Ecoli | 7 (7/0/0) | 336 | 8 | Twonorm | 20 (20/0/0) | 7400 | 2 |
| Pima | 8 (8/0/0) | 768 | 2 | Ringnorm | 20 (20/0/0) | 7400 | 2 |
| Yeast | 8 (8/0/0) | 1484 | 10 | Wdbc | 30 (30/0/0) | 569 | 2 |
| Glass | 9 (9/0/0) | 214 | 7 | SatImage | 36 (0/36/0) | 6435 | 6 |
| Page-blocks | 10 (4/6/0) | 5472 | 5 | Texture | 40 (40/0/0) | 5500 | 11 |
| Magic | 10 (10/0/0) | 19020 | 2 | Spectfheart | 44 (0/44/0) | 267 | 2 |
| Wine | 13 (13/0/0) | 178 | 3 | Spambase | 57 (57/0/0) | 4597 | 2 |
| Heart | 13 (1/12/0) | 270 | 2 | Sonar | 60 (60/0/0) | 208 | 2 |
| Cleveland | 13 (13/0/0) | 297(303) | 5 | Movementlibras | 90 (90/0/0) | 360 | 15 |

Available at `http://sci2s.ugr.es/keel/datasets.php`

For each dataset, we therefore consider the average results of 30 runs.

### B. Methods Considered for Comparison

In these experiments, we compare the proposed approach with other ten methods, which are available in the KEEL software tool [61]. A brief description of these methods is as follows.

1) *C4.5* [39]: This is a well-known algorithm used to generate a decision tree from a set of training data in the same way as the ID3 algorithm [62]. The extensions or improvements with respect to ID3 are that it accounts for unavailable or missing values in data, it handles continuous attribute value ranges, it chooses an appropriate attribute selection measure (maximizing the gain ratio), and it prunes the resulting decision trees.

2) *Classification based on associations (CBA)* [12]: This method consists of two parts. In the first part, an algorithm based on the Apriori algorithm [31] is used to mine the interval association rules for classification. In the second part, this sorts the generated rules according to their precedence relation and chooses a set of high precedence rules to cover the training data.

3) *CBA2* [13]: This method is the second version of the CBA algorithm, which improves the previous system by the use of multiple class minimum support in rule generation.

4) *Classification based on multiple association rules (CMAR)* [14]: This method extends an efficient frequent pattern (FP) mining method, i.e., FP-Growth [63], constructs a class distribution-associated FP-tree, and mines large databases efficiently. Moreover, it applies a CR-tree structure to store and retrieve mined interval association rules efficiently, and it prunes rules effectively based on confidence, correlation (by using a weighted chi-square method), and database coverage. The classification is performed based on a weighted chi-square analysis using multiple strong association rules.

5) *Structural learning algorithm on vague environment (2SLAVE)* [64]: This method is a modification of the GA of the SLAVE algorithm [65] in order to include a feature selection process. This is an inductive learning algorithm based on the iterative rule learning approach, in which each chromosome represents a rule, to obtain a set of disjunctive normal form (DNF) fuzzy rules. Chromosomes compete in every GA run, choosing the best rule per run. The global solution is formed by the best rules obtained when the algorithm is run multiple times.

6) *Learning algorithm to discover fuzzy association rules for classification (LAFAR)* [24]: This method uses a GA to automatically determine the minimum fuzzy support and the minimum fuzzy confidence. To evaluate a determined chromosome, this method finds frequent fuzzy grids and generates fuzzy classification rules from them. Once the whole classifier is obtained, the fitness value can be calculated, which maximizes the classification accuracy rate and minimizes the number of fuzzy rules. When reaching the termination condition, the chromosome with the maximum fitness value is used to test the performance of the proposed method.

7) *Classification based on predictive association rules (CPAR)* [15]: This method adopts a greedy algorithm to generate interval association rules directly from training data. In this process, this algorithm selects multiple literals with similar gains to build multiple rules simultaneously in order to avoid missing important rules. To perform the classification, this uses expected accuracy to evaluate each rule and uses the best $k$ rules in prediction.

8) *Fuzzy hybrid genetic-based machine learning algorithm (FH-GBML)* [66]: This method follows a genetic cooperative–competitive learning (GCCL) approach and consists of two processes. The first process is used to generate good fuzzy rules while the second one is used to find good combinations of generated fuzzy rules. This method simultaneously uses multiple fuzzy partitions with different granularities for fuzzy rule extraction, using four

TABLE III
PARAMETERS CONSIDERED FOR COMPARISON

| Method | Parameters |
|--------|------------|
| **C4.5** | $Pruned = yes, Confidence = 0.25, InstancesPerLeaf = 2$ |
| **CBA** | $Minsup = 0.01, Minconf = 0.5, Pruned = yes, RuleLimit = 80,000$ |
| **CBA2** | $Minsup = 0.01, Minconf = 0.5, Pruned = yes, RuleLimit = 80,000$ |
| **CMAR** | $Minsup = 0.01, Minconf = 0.5, \delta = 4, DifferenceThreshold = 20\%$ |
| **2SLAVE** | $Pop = 100, Iter_{change} = 500, P_c = 0.6, P_m = 0.01, \kappa_1 = 0, \kappa_2 = 1, Op_{or} = 0.1,$ $Op_{and} = 0.1, P_{gen} = 0.1, P_{rot} = 0.1, \alpha = 0.15$ |
| **LAFAR** | $t_{max} = 50, N_{pop} = 30, \mid s \mid = \mid c \mid = 10, W_{CAR} = 10, W_V = 1, P_c = 1.0, P_m = 0.01,$ $J_{max} = 100, \eta_1 = 0.001, \eta_2 = 0.1$ |
| **CPAR** | $\delta = 0.05, min\_gain = 0.7, \alpha = 0.66, k = 5$ |
| **FH-GBML** | $N_{rules} = 20, F_{sets} = 200, Gen = 1000, P_c = 0.9, P_{dcare} = \{0.5, 0.8, 0.95\},$ $P_{michigan} = 0.5$ |
| **SGERD** | $Q = heuristics$ |
| **CFAR** | $Minpsup = \{0.05, 0.1\}, Minpconf = 0.85, MS = 0.15$ |
| **FARC-HD** | $Minsup = 0.05, Maxconf = 0.8, Depth_{max} = 3, k_t = 2,$ $Pop = 50, Evaluations = 15,000, BITSGENE = 30, \delta = 0.2$ |

homogeneous fuzzy partitions with triangular fuzzy sets and a don't-care condition.

9) *Steady-state GA for extracting fuzzy classification rules from data (SGERD)* [67]: It is a steady-state GA to generate a prespecified number of $Q$ rules per class following a GCCL approach. In each iteration, parents and their corresponding offspring compete to select the best $Q$ rules for each class. This method also simultaneously uses multiple fuzzy partitions with different granularities and a don't-care condition for fuzzy rule extraction.

10) *Classification with fuzzy association rules (CFAR)* [27]: This method uses the Apriori algorithm to mine all the fuzzy association rules for classification and remove the conflicting and redundant rules to generate a compact set of rules denoted as CompSet. Then, this method selects the best rules to build the classifier by means of two processes. In the first process, for each pattern, CompSet is sorted by matching and confidence degree, rewarding the best rule that classify this pattern and punishing the rules that do not classify it. In the second process, the worst rules from CompSet are removed. These processes are iterated until the error rate in the training set increases.

### C. Parameters of the Methods

The parameters of the analyzed methods are shown in Table III.[2] Notice that only the rules with a number of antecedent conditions less than or equal to 3 are examined for our proposal. This restriction is intended to facilitate the discovery of a small number of short (i.e., simple) fuzzy rules. The parameters of the remaining methods were selected according to the recommendation of the corresponding authors within each proposal,

which are the default parameter settings included in the KEEL software tool [61]. Notice that in the FH-GBML algorithm, the authors used three different probabilities of don't care (0.5, 0.8, and 0.95 depending on the size of the dataset) to obtain fuzzy rules with a few antecedent fuzzy sets. In these experiments, we have used these three probabilities of don't care in each dataset and have shown in the tables the best average result obtained for each one. Furthermore, in the CFAR algorithm, the authors used 0.1 as the minimum support and this could be very high for some datasets (we are using 0.05 as the minimum support in our proposal). Likewise, we have used these two minimum supports in each dataset (0.1 and 0.05), and we have shown in the tables the best average result obtained in each one.

The initial linguistic partitions for our proposal and the fuzzy methods analyzed are comprised of *five linguistic terms* with uniformly distributed triangular MFs giving meaning to them, except in the FH-GBML and SGERD algorithms, where the partitions are comprised of *two, three, four, and five linguistic terms* for each attribute. The discretization of the continuous attributes for the CBA, CBA2, CMAR, and CPAR algorithms is done by the use of the entropy method [68]. Notice that we use a crisp label for each value of the nominal variables.

### D. Statistical Analysis

In order to assess whether significant differences exist among the results, we adopt statistical analysis [41]–[43] and, in particular, nonparametric tests, according to the recommendations made in [40], where a set of simple, safe, and robust nonparametric tests for statistical comparisons of classifiers has been introduced.

For pairwise comparison, we use Wilcoxon's Signed-Ranks test [69], [70], and for multiple comparison we employ Friedman's test [71], Iman and Davenport's test [72], and Holm's method [73]. In order to perform multiple comparisons, it is necessary to check whether all the results obtained by the algorithms present any significant difference (Friedman's test and

---

[2]With these values for our proposal, we have tried to facilitate comparisons, selecting standard common parameters that work well in most cases instead of searching for very specific values.

TABLE IV
RESULTS OBTAINED BY THE ANALYZED METHODS

| Dataset | 2SLAVE | | | | FH-GBML | | | | SGERD | | | | FARC-HD | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | #R | #C | Tra | Tst | #R | #C | Tra | Tst | #R | #C | Tra | Tst | #R | #C | Tra | Tst |
| Iris | 4.0 | 3.2 | 94.32 | 94.44 | 14.9 | 3.3 | **98.89** | 94.00 | **3.4** | 2.0 | 95.14 | 94.89 | 4.0 | **1.1** | 98.59 | **96.00** |
| Phoneme | 11.5 | 24.2 | 77.52 | 76.41 | 17.4 | 4.5 | 79.57 | 79.66 | **3.6** | **1.9** | 75.74 | 75.55 | 17.8 | 2.2 | **83.52** | **82.14** |
| Monks | 3.0 | **1.3** | 97.22 | 97.26 | 14.7 | 2.1 | 98.36 | 98.18 | **2.2** | 1.4 | 80.56 | 80.65 | 14.2 | 2.0 | **99.92** | **99.77** |
| Appendicitis | 4.4 | 7.5 | 91.20 | 82.91 | 13.8 | 7.0 | 93.19 | **86.00** | **2.5** | 2.0 | 87.88 | 84.48 | 6.8 | **1.8** | **93.82** | 84.18 |
| Ecoli | 12.6 | 9.6 | 89.51 | **84.53** | 10.3 | 4.2 | 75.83 | 69.38 | 9.4 | **1.6** | 76.53 | 74.05 | 33.8 | 2.4 | **92.33** | 82.19 |
| Pima | 7.8 | 8.8 | 76.35 | 73.71 | 10.6 | 6.0 | 77.18 | 75.26 | **3.1** | 2.0 | 74.01 | 73.37 | 22.7 | 2.4 | **82.90** | 75.66 |
| Yeast | 23.6 | 9.8 | 55.54 | 51.27 | **7.5** | 5.9 | 52.31 | 51.42 | 11.3 | **1.5** | 39.83 | 38.77 | 35.2 | 2.6 | **63.81** | **58.50** |
| Glass | 15.1 | 9.3 | 74.25 | 58.05 | 9.4 | 5.0 | 64.85 | 57.99 | **6.9** | 2.0 | 61.31 | 58.49 | 22.7 | 2.5 | **81.10** | **70.24** |
| Page-blocks | 7.5 | 10.3 | 91.39 | 91.39 | 7.4 | 8.1 | 94.37 | 94.21 | **6.5** | 2.0 | 90.83 | 90.72 | 19.1 | 2.3 | **95.62** | **95.01** |
| Magic | 4.1 | 10.5 | 73.97 | 73.96 | 9.9 | 8.2 | 81.23 | 81.30 | **3.1** | 2.0 | 72.17 | 72.06 | 43.3 | 2.5 | **85.36** | **84.51** |
| Wine | 5.5 | 10.3 | 92.52 | 89.47 | 9.2 | 4.7 | 95.51 | 92.61 | **4.2** | 2.0 | 93.67 | 91.88 | 8.7 | **1.6** | **99.94** | **94.35** |
| Heart | 4.3 | 10.7 | 75.35 | 71.36 | 12.7 | 3.2 | 84.65 | 75.93 | **2.7** | **1.9** | 74.83 | 73.21 | 27.0 | 2.6 | **93.91** | **84.44** |
| Cleveland | 11.9 | 12.8 | 54.24 | 48.82 | 6.9 | 4.5 | 58.29 | 53.51 | **6.4** | 2.0 | 56.55 | 51.59 | 61.3 | 2.9 | **88.18** | **55.24** |
| Vowel | 63.1 | 15.6 | **82.06** | 71.11 | **9.2** | 13.0 | 67.41 | 67.07 | 18.0 | **1.9** | 72.99 | 65.83 | 72.3 | 2.9 | 80.48 | **71.82** |
| Crx | 2.4 | 6.7 | 74.36 | 74.06 | 11.6 | 6.2 | 86.32 | **86.60** | **2.1** | **1.9** | 85.04 | 85.03 | 25.4 | 2.6 | **91.17** | 86.03 |
| Pen-based | 40.0 | 18.9 | 81.32 | 81.16 | 18.4 | 8.0 | 50.69 | 50.45 | **15.9** | 2.0 | 68.17 | 67.93 | 152.8 | 2.8 | **97.04** | **96.04** |
| German | 6.5 | 8.4 | 72.44 | 70.53 | 5.1 | 4.0 | **87.11** | **87.01** | **3.4** | 2.0 | 68.54 | 67.97 | 85.7 | 2.8 | 86.81 | 72.80 |
| Twonorm | 26.5 | 15.5 | 87.45 | 86.99 | 12.0 | 7.6 | 86.26 | 85.97 | **3.1** | 2.0 | 74.49 | 73.98 | 60.9 | 2.6 | **96.64** | **95.28** |
| Ringnorm | **4.6** | 23.7 | 80.12 | 79.63 | 6.9 | 11.3 | 87.34 | 86.92 | 6.8 | 2.0 | 73.21 | 72.63 | 24.0 | **1.9** | **95.13** | **94.08** |
| Wdbc | 5.2 | 8.1 | 92.43 | 92.33 | 7.2 | 4.9 | 95.12 | 92.26 | **3.7** | 2.0 | 91.79 | 90.68 | 10.4 | **1.7** | **98.57** | **95.25** |
| SatImage | 57.9 | 25.1 | 84.03 | 81.69 | 16.5 | 36.0 | 74.90 | 74.72 | 12.2 | 2.0 | 77.15 | 77.10 | 76.1 | 2.7 | **88.68** | **87.32** |
| Texture | 34.9 | 23.9 | 82.87 | 81.57 | **14.6** | 40.0 | 69.91 | 70.15 | 18.6 | 2.0 | 72.12 | 71.66 | 54.5 | 2.7 | **93.71** | **92.89** |
| Spectfheart | 6.1 | 21.7 | 80.71 | 79.17 | 10.8 | 44.0 | 79.28 | 72.36 | **2.1** | **1.9** | 79.05 | 78.16 | 12.9 | **1.8** | **91.43** | **79.83** |
| Spambase | 7.9 | 11.4 | 69.87 | 70.14 | 3.9 | 18.5 | 77.86 | 77.22 | **3.7** | 2.0 | 72.90 | 72.98 | 29.8 | 2.4 | **92.37** | **91.93** |
| Sonar | 9.6 | 17.5 | 77.92 | 71.42 | 10.3 | 4.7 | 80.56 | 68.24 | **3.2** | 2.0 | 74.22 | 71.90 | 18.0 | 2.3 | **98.77** | **80.19** |
| Movementlibras | 47.4 | 26.5 | 90.13 | 67.04 | **12.1** | 90.0 | 77.87 | 68.89 | 22.9 | 2.0 | 72.37 | 68.09 | 83.1 | 2.9 | **95.52** | **76.67** |
| Mean | 16.4 | 13.5 | 80.73 | 76.94 | 10.9 | 13.6 | 79.80 | 76.82 | **6.9** | **1.9** | 75.43 | 73.99 | 39.3 | 2.3 | **90.97** | **83.94** |

Iman–Davenport's test), and in the case of finding one, then, we can find out by using a *post-hoc* test to compare the control algorithm with the remaining algorithms (Holm's test). We use $\alpha = 0.05$ as the level of confidence in all cases. A wider description of these tests, together with software for their use, can also be found at: http://sci2s.ugr.es/sicidm/.

## V. EXPERIMENTAL RESULTS

In this section, we analyze the results obtained in the different experiments. This section is organized as follows.

1) In Section V-A, we show a statistical study obtained from the comparison with other three GFSs, including FH-GBML [66], 2SLAVE [64], and SGERD [67].
2) In Section V-B, we compare the performance of our approach with two other approaches to obtain a fuzzy associative classifier: the LAFAR algorithm [24] and the CFAR algorithm [27].
3) In Section V-C, we compare the performance of our approach with the C4.5 decision tree [39] and four classical approaches for associative classification: the CBA algorithm [12], the CBA2 algorithm [13], the CMAR algorithm [14], and the CPAR algorithm [15].
4) In Section V-D, we show an analysis of the performance of our approach, depending on the depth of the trees and the number of evaluations in the genetic process.
5) In Section V-E, we analyze the scalability of our proposal.

### A. Comparison With Other Genetic Fuzzy Systems

This section analyzes the performance of our model against three recognized GFSs. The results obtained by the analyzed methods are shown in Table IV, where we have the following.

1) $\#R$ stands for the average number of rules.
2) $\#C$ stands for the average number of conditions in the antecedent of the rules.
3) Tra stands for the average classification percentage obtained over the training data.
4) Tst stands for the average classification percentage obtained over the test data.

The best global result for each one is stressed in boldface in each case.

In order to compare the results, we have used nonparametric tests for multiple comparison to find the best approach (see Section IV-D), considering the average results obtained in test (Tst). First of all, we have used the Friedman and Iman–Davenport tests in order to find out whether significant differences exist among all the mean values. Table V shows the Friedman and Iman–Davenport statistics, and it relates them to the corresponding critical values for each distribution by using a level of significance, i.e., $\alpha = 0.05$. The $p$-value obtained is also reported for each test. Given that the statistics of Friedman and Iman–Davenport are clearly greater than their associated critical values, there are significant differences among the observed results with a level of significance $\alpha \leq 0.05$. Table VI

TABLE V
RESULTS OF THE FRIEDMAN AND IMAN–DAVENPORT TESTS ($\alpha = 0.05$)

| Friedman Test | | |
|---|---|---|
| Statistic ($\mathcal{X}_F^2$) | Critical Value | *p* value |
| **40.823** | 7.814 | <0.0001 |
| **ImanDavenport Test** | | |
| Statistic ($\mathcal{F}_F$) | Critical Value | *p* value |
| **27.452** | 2.7386 | <0.0001 |

TABLE VI
AVERAGE RANKINGS OF THE METHODS

| Method | Ranking |
|---|---|
| SGERD | 3.2884 |
| 2SLAVE | 2.9808 |
| FH-GBML | 2.5577 |
| FARC-HD | **1.1731** |

TABLE VII
HOLM'S TABLE FOR THE SELECTION METHODS WITH $\alpha = 0.05$

| *i* | Method | *z* | *p* | $\alpha/i$ | Hypothesis |
|---|---|---|---|---|---|
| 3 | SGERD | 5.01 | 3.46E-9 | 0.0166 | **Rejected** |
| 2 | 2SLAVE | 5.05 | 4.45E-7 | 0.0125 | **Rejected** |
| 1 | FH-GBML | 3.87 | 1.10E-4 | 0.05 | **Rejected** |

shows the rankings (which are computed by the use of a Friedman test) of the different methods that are considered in this study.

We now apply Holm's test to compare the best ranking method (FARC-HD) with the remaining methods. Table VII presents these results. In this table, the methods are ordered with respect to the *z*-value obtained. Holm's test rejects the hypothesis of equality with the rest of the methods ($p < \alpha/i$). Therefore, by the analysis of the statistical study that is shown in Tables VI and VII, we conclude that our model is a solid approach to deal with high-dimensional datasets, as it has shown itself to be the best accuracy method when compared with the remaining fuzzy GFSs that are applied in this study.

Finally, the results presented in Table IV show that our proposal obtains a higher average number of rules (39.2 rules on average) than all the GFSs (good approaches to obtain very compact models), showing a good trade-off closer to the accuracy with rules involving no more than three attributes in their antecedent and giving the advantage of easier understanding with respect to the 2SLAVE and FH-GBML.

### B. Comparison With Other Fuzzy Associative Classifiers

In this section, we compare the performance of our model with two other approaches to obtain a fuzzy associative classifier: the LAFAR algorithm [24] and the CFAR algorithm [27]. The results obtained by these methods are shown in Table VIII. (This kind of table was described in Section V-A.) Notice that we show less datasets; this is due to scalability problems in the LAFAR and CFAR algorithms, which cannot run in all datasets.

In order to compare the two algorithms, we use a Wilcoxon test, which is shown in Table IX. We can observe that the null hypothesis for the Wilcoxon test has been rejected (*p*-value<=

TABLE VIII
RESULTS OBTAINED BY THE ANALYZED METHODS

| Dataset | LAFAR | | | | CFAR | | | | FARC-HD | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | #R | #C | Tra | Tst | #R | #C | Tra | Tst | #R | #C | Tra | Tst |
| Iris | 17.4 | **1.0** | **98.74** | 92.67 | 9.1 | 1.5 | 91.33 | 90.67 | **4.0** | 1.1 | 98.59 | **96.00** |
| Phoneme | 16.9 | 1.1 | 78.49 | 77.87 | **2.0** | **1.0** | 70.65 | 70.65 | 17.8 | 2.2 | **83.52** | **82.14** |
| Monks | 126.4 | 3.9 | 94.64 | 93.36 | **12.1** | **1.8** | **100.00** | 99.55 | 14.2 | 2.0 | 99.92 | **99.77** |
| Appendicitis | 39.1 | **1.1** | **95.91** | 86.00 | 14.8 | 1.4 | 88.47 | **87.82** | **6.8** | 1.8 | 93.82 | 84.18 |
| Ecoli | 68.4 | **2.1** | 81.58 | 72.36 | **4.0** | 3.1 | 48.45 | 47.61 | 33.8 | 2.4 | **92.33** | **82.19** |
| Pima | 32.3 | **1.1** | 77.36 | 75.40 | **2.0** | 1.9 | 65.10 | 65.11 | 22.7 | 2.4 | **82.90** | 75.66 |
| Yeast | 40.2 | 2.0 | 54.53 | 50.90 | **3.0** | **1.0** | 36.44 | 36.44 | 35.2 | 2.6 | **63.81** | **58.50** |
| Glass | 36.0 | 2.5 | 68.59 | 52.17 | **3.4** | 2.1 | 44.19 | 44.66 | 22.7 | 2.5 | **81.10** | **70.24** |
| Page-blocks | | | | | **2.0** | **1.0** | 89.78 | 89.78 | 19.1 | 2.3 | **95.62** | **95.01** |
| Magic | | | | | **2.0** | 1.5 | 64.84 | 64.84 | 43.3 | 2.5 | **85.36** | **84.51** |
| Wine | | | | | 115.7 | 3.0 | 99.63 | 93.24 | **8.7** | 1.6 | **99.94** | **94.35** |
| Heart | | | | | 45.3 | 3.3 | 86.01 | 82.22 | **27.0** | 2.6 | **93.91** | **84.44** |
| Cleveland | | | | | **2.0** | 2.1 | 53.87 | 53.88 | 61.3 | 2.9 | **88.18** | **55.24** |
| Crx | | | | | 89.9 | 5.1 | 89.45 | **86.79** | 25.4 | 2.6 | **91.17** | 86.03 |
| Pen-based | | | | | **6.9** | 2.9 | 36.54 | 36.43 | 152.8 | 2.8 | **97.04** | **96.04** |
| German | | | | | **2.0** | 2.7 | 70.00 | 70.00 | 85.7 | 2.8 | **86.81** | **72.80** |
| Twonorm | | | | | 498.7 | 3.7 | 91.96 | 91.66 | **60.9** | 2.6 | **96.64** | **95.28** |
| Sonar | | | | | 37.3 | 3.0 | 83.01 | 72.48 | **18.0** | 2.3 | **98.77** | **80.19** |
| Mean | 47.1 | **1.8** | 81.23 | 75.09 | 47.3 | **2.3** | 72.76 | 71.32 | **36.6** | **2.3** | **90.52** | **82.92** |

TABLE IX
WILCOXON'S TEST ($\alpha = 0.05$)

| Comparison | $R^+$ | $R^-$ | Hypothesis | p-value |
|---|---|---|---|---|
| FARC-HD vs. CFAR | 161 | 10 | **Rejected** | 0.001 |
| FARC-HD vs. LAFAR | 34 | 2 | **Rejected** | 0.025 |
| LAFAR vs. CFAR | 32 | 4 | **Rejected** | 0.05 |

$\alpha$), and our proposal has achieved a higher ranking. We may conclude that our proposal also presents the best performance in this case.

On the other hand, the results presented in Table VIII show that our approach obtains an average number of rules lower than the LAFAR and CFAR algorithms. However, the CFAR algorithm obtains less rules than our approach in 11 of the 18 datasets.

### C. Comparison With Classical Approaches

This section analyzes the performance of our model against five classical approaches. The results obtained by the analyzed methods are shown in Table X.

In order to compare the results, we have applied the nonparametric tests described in Section V-A. Table XI shows that the statistics of Friedman and Iman–Davenport are clearly greater than their associated critical values, and there are significant differences among the observed results with a level of significance, i.e., $\alpha \leq 0.05$. Table XII shows the rankings (computed using a Friedman test) of the different methods considered in this study.

Table XIII shows that Holm's test rejects the hypothesis of equality with the rest of the methods ($p < \alpha/i$). Therefore, by the analysis of the statistical study shown in Tables XII and XIII, we conclude that our model is the best performing method when compared with the remaining classical approaches applied in this study. Finally, the results presented in Table X show that our proposal obtains a smaller average number of rules than the remaining approaches.

TABLE X
RESULTS OBTAINED BY THE ANALYZED METHODS

| | CBA | | | CBA2 | | | CMAR | | | CPAR | | | C45 | | | FARC-HD | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Data** | **#R** | **Tra** | **Tst** | **#R** | **Tra** | **Tst** | **#R** | **Tra** | **Tst** | **#R** | **Tra** | **Tst** | **#R** | **Tra** | **Tst** | **#R** | **Tra** | **Tst** |
| Iris | 4 | 96.7 | 93.3 | 4 | 96.7 | 93.3 | 41 | 96.4 | 94.0 | 34 | 96.4 | **96.0** | 5 | 98.0 | **96.0** | **4** | 98.6 | 96.0 |
| Phoneme | 81 | 81.4 | 80.5 | 116 | 82.2 | 81.2 | 358 | 79.3 | 78.6 | 1109 | 84.0 | 81.9 | 129 | **91.9** | **86.8** | 18 | 83.5 | 82.1 |
| Monks | 11 | **100.0** | **100.0** | 11 | **100.0** | **100.0** | 126 | **100.0** | **100.0** | 79 | **100.0** | **100.0** | 5 | **100.0** | **100.0** | 14 | 99.9 | 99.8 |
| Appendicitis | 6 | 91.5 | 89.6 | 6 | 91.5 | 89.6 | 42 | 90.8 | **89.7** | 25 | 89.5 | 87.8 | 3 | 91.0 | 83.3 | 7 | **93.8** | 84.2 |
| Ecoli | 29 | 87.3 | 78.0 | 35 | 89.2 | 77.1 | 178 | 82.5 | 77.7 | 158 | 82.6 | 76.2 | 20 | 91.7 | 79.5 | 34 | **92.3** | **82.2** |
| Pima | 42 | 79.6 | 72.7 | 44 | 79.9 | 72.5 | 237 | 79.4 | 75.1 | 166 | 79.1 | 74.5 | 18 | **83.2** | 74.0 | 23 | 82.9 | **75.7** |
| Yeast | 48 | 59.4 | 54.7 | 67 | 61.8 | 55.9 | 262 | 57.2 | 53.6 | 427 | 59.4 | 56.3 | 169 | **81.5** | 55.6 | 35 | 63.8 | **58.5** |
| Glass | 25 | 82.3 | 70.8 | 28 | 83.1 | **71.3** | 174 | 78.9 | 70.3 | 121 | 77.2 | 68.9 | 26 | **93.7** | 67.4 | 23 | 81.1 | 70.2 |
| Page-blocks | 172 | 94.6 | 94.0 | 256 | 98.2 | 95.9 | 1348 | 94.7 | 94.4 | 464 | 97.4 | 96.1 | 43 | **98.5** | **97.1** | 19 | 95.6 | 95.0 |
| Magic | 728 | 82.7 | 81.5 | 1015 | 84.8 | 83.7 | 4200 | 79.3 | 78.8 | 5422 | 87.9 | 84.9 | 321 | 90.9 | **85.1** | 43 | 85.4 | 84.5 |
| Wine | 9 | **99.9** | 93.8 | 9 | **99.9** | 93.8 | 73 | **99.9** | **96.7** | 44 | 99.7 | 95.6 | 5 | 98.9 | 93.3 | 9 | **99.9** | 94.3 |
| Heart | 40 | 93.8 | 83.0 | 40 | **94.2** | 81.5 | 306 | 91.5 | 82.2 | 93 | 90.1 | 80.7 | 17 | 91.7 | 78.5 | 27 | 93.9 | **84.4** |
| Cleveland | **30** | 64.3 | **56.9** | 47 | 70.0 | 54.9 | 230 | 54.1 | 53.9 | 107 | 61.8 | 54.9 | 40 | 83.1 | 54.5 | 61 | **88.2** | 55.2 |
| Vowel | 130 | 77.8 | 63.6 | 215 | 92.6 | 74.9 | 945 | 71.2 | 60.4 | 888 | 76.1 | 63.0 | 95 | **97.1** | **81.5** | 72 | 80.5 | 71.8 |
| Crx | 97 | **95.2** | 83.6 | 97 | 94.9 | 85.0 | 728 | 92.0 | 85.0 | 204 | 91.1 | **87.3** | 21 | 90.8 | 85.3 | 25 | 91.2 | 86.0 |
| Pen-based | 653 | 85.3 | 83.1 | 592 | 90.3 | 88.1 | 4328 | 72.0 | 71.7 | 2536 | 97.9 | 93.1 | 191 | **99.3** | **96.5** | 153 | 97.0 | 96.0 |
| German | 177 | **94.5** | **75.3** | 227 | 92.9 | 73.7 | 1984 | 84.4 | 71.9 | 666 | 88.1 | 73.3 | 83 | 84.9 | 72.5 | 86 | 86.8 | 72.8 |
| Twonorm | 708 | 97.7 | 91.6 | 724 | 97.0 | 91.7 | 2163 | 97.7 | **95.5** | 1646 | 99.0 | 89.4 | 286 | 98.7 | 84.5 | 61 | 96.6 | 95.3 |
| Ringnorm | 295 | 96.3 | **94.1** | 184 | 97.2 | 93.7 | 2788 | 87.2 | 83.7 | 1209 | **99.1** | 92.2 | 203 | 98.7 | 90.2 | 24 | 95.1 | 94.1 |
| Wdbc | 50 | **100.0** | 94.7 | 51 | 99.9 | 95.1 | 309 | 99.9 | 94.9 | 66 | 99.0 | 95.1 | 12 | 99.1 | 95.2 | 10 | 98.6 | **95.3** |
| SatImage | 722 | 88.5 | 85.2 | 587 | 87.6 | 83.6 | 7083 | 86.6 | 84.9 | 2032 | 97.3 | 85.8 | 287 | **97.7** | 85.8 | 76 | 88.7 | **87.3** |
| Texture | 377 | 88.3 | 84.5 | **50** | 53.5 | 52.5 | 5616 | 76.3 | 73.8 | 1492 | 98.5 | 90.7 | 152 | **99.0** | 92.6 | 55 | 93.7 | **92.9** |
| Spectfheart | 31 | 92.0 | 79.8 | 33 | 91.8 | 79.8 | 305 | 85.5 | 79.4 | 87 | 87.3 | 78.3 | 19 | **98.3** | 76.5 | 13 | 91.4 | 79.8 |
| Spambase | 278 | 95.0 | 93.2 | 302 | 95.2 | 93.0 | 2317 | 92.9 | 92.0 | 686 | 97.9 | **93.5** | 109 | 97.4 | 92.7 | 30 | 92.4 | 91.9 |
| Sonar | 27 | 92.3 | 75.4 | 39 | 97.7 | 77.9 | 234 | 98.3 | 78.8 | 92 | 95.8 | 75.0 | 14 | 97.7 | 70.5 | 18 | **98.8** | **80.2** |
| Movementlibras | 41 | 54.8 | 36.1 | **4** | 11.6 | 7.2 | 510 | 51.7 | 39.2 | 636 | 85.3 | 63.6 | 47 | 94.4 | 69.4 | 83 | **95.5** | **76.7** |
| Mean | 185 | 87.4 | 80.3 | 184 | 85.9 | 78.7 | 1419 | 83.8 | 79.1 | 788 | 89.1 | 82.1 | 89 | **94.1** | 82.5 | 39 | 91.0 | **83.9** |

TABLE XI
RESULTS OF THE FRIEDMAN AND IMAN–DAVENPORT TESTS ($\alpha = 0.05$)

| **Friedman Test** | | |
|---|---|---|
| **Statistic ($\mathcal{X}_F^2$)** | **Critical Value** | **$p$ value** |
| **15.84** | 11.0705 | $<0.01$ |
| **ImanDavenport Test** | | |
| **Statistic ($\mathcal{F}_F$)** | **Critical Value** | **$p$ value** |
| **3.4703** | 2.2885 | $<0.01$ |

TABLE XII
AVERAGE RANKINGS OF THE METHODS

| **Method** | **Ranking** |
|---|---|
| CBA | 3.807 |
| CMAR | 3.775 |
| CBA2 | 3.730 |
| C45 | 3.576 |
| CPAR | 3.346 |
| FARC-HD | **2.307** |

TABLE XIII
HOLM'S TABLE FOR THE SELECTION METHODS WITH $\alpha = 0.05$

| $i$ | **Method** | $z$ | $p$ | $\alpha/i$ | **Hypothesis** |
|---|---|---|---|---|---|
| 5 | CMAR | 3.71 | 2.1035E-4 | 0.01 | **Rejected** |
| 4 | CBA | 2.89 | 0.0038 | 0.0125 | **Rejected** |
| 3 | CBA2 | 2.74 | 0.0060 | 0.0166 | **Rejected** |
| 2 | C45 | 2.45 | 0.0144 | 0.025 | **Rejected** |
| 1 | CPAR | 2.01 | 0.0453 | 0.05 | **Rejected** |

## D. Analysis of the Influence of $Depth_{max}$ and the Number of Evaluations

In this section, several experiments have been carried out to analyze the performance of our approach depending on $Depth_{max}$ and the number of evaluations in the genetic selection and tuning process (using the experimental setting described in Section IV). In order to make this analysis easier to interpret, we have used four representative datasets in this experiments: Yeast, Vowel, Ringnorm, and Spectfheart (8, 13, 20, and 44 variables, respectively). Table XIV shows the results obtained with three different values for $Depth_{max}$ (2, 3, and 4), where $\#R1$ stands for the average number of rules obtained at the end of Stage 1, $\#R2$ stands for the average number of rules obtained at the end of Stage 2, $\#R3$ stands for the average number of rules obtained at the end of Stage 3, and "time" stands for the average runtime (in format hh : mm : ss).

By the analysis of the results presented in Table XIV, we can highlight the following facts.

1) Candidate rule prescreening allows the selection of a reduced number of interesting rules with the three values for $Depth_{max}$, decreasing the computational cost in the genetic selection and tuning process. Notice that the number of rules obtained in Stage 1 for the dataset Spectfheart is higher than 300 000 rules.

TABLE XIV
ANALYSIS OF THE PERFORMANCE DEPENDING ON $Depth_{max}$

| Data | $Depth_{max} = 2$ | | | | | | $Depth_{max} = 3$ | | | | | | $Depth_{max} = 4$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | #R1 | #R2 | #R3 | Tra | Tst | Time | #R1 | #R2 | #R3 | Tra | Tst | Time | #R1 | #R2 | #R3 | Tra | Tst | Time |
| Yeast | 96.9 | 38.9 | 27.9 | 62.2 | 57.7 | 0:00:51 | 373.2 | 56.3 | 35.2 | 63.8 | 58.5 | 0:01:00 | 845.5 | 68.1 | 43.7 | 64.6 | 57.6 | 0:02:03 |
| Vowel | 211.9 | 48.3 | 35.2 | 60.4 | 52.3 | 0:00:58 | 1482.3 | 97.5 | 72.3 | 80.4 | 71.8 | 0:02:16 | 6423 | 111.1 | 90.5 | 89.8 | 78.4 | 0:07:10 |
| Ringnorm | 1460.4 | 36.7 | 25.9 | 94.8 | 93.8 | 0:05:44 | 9927.7 | 41.4 | 24.0 | 95.1 | 94.1 | 0:07:28 | 35201.1 | 45.4 | 28.0 | 95.9 | 94.6 | 0:19:42 |
| Spectfheart | 2870.0 | 37.6 | 13.8 | 91.3 | 79.1 | 0:00:19 | 45508.1 | 34.7 | 12.9 | 91.4 | 79.8 | 0:01:59 | 389983 | 32.8 | 15.4 | 92.6 | 77.5 | 1:10:21 |

TABLE XV
AVERAGE RUNTIME OF THE ANALYZED METHODS (HH:MM:SS)

| Dataset | Var | Patt | 2SLAVE | FH-GBML | SGERD | CFAR | LAFAR | CBA | CBA2 | CMAR | CPAR | C4.5 | FARC-HD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Iris | 4 | 150 | 00:00:05 | 00:12:57 | **00:00:00** | **00:00:00** | 00:09:39 | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | 00:00:01 |
| Phoneme | 5 | 5404 | 00:03:01 | 01:39:48 | 00:00:01 | 00:00:05 | 01:08:27 | 00:00:01 | 00:00:01 | 00:00:01 | 00:00:05 | **00:00:00** | 00:01:44 |
| Monks | 6 | 432 | 00:00:07 | 00:10:57 | **00:00:00** | 00:00:01 | 07:39:18 | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | 00:00:09 |
| Appendicitis | 7 | 106 | 00:00:03 | 00:02:05 | **00:00:00** | 00:00:05 | 01:00:44 | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | 00:00:02 |
| Ecoli | 7 | 336 | 00:00:50 | 00:11:05 | **00:00:00** | 00:00:01 | 01:20:10 | **00:00:00** | **00:00:00** | **00:00:00** | 00:00:01 | **00:00:00** | 00:00:15 |
| Pima | 8 | 768 | 00:00:41 | 00:23:31 | **00:00:00** | 00:00:06 | 04:18:18 | **00:00:00** | **00:00:00** | **00:00:00** | 00:00:01 | **00:00:00** | 00:00:22 |
| Yeast | 8 | 1484 | 00:07:04 | 00:48:51 | **00:00:00** | 00:00:01 | 15:27:38 | **00:00:00** | **00:00:00** | **00:00:00** | 00:00:01 | 00:00:01 | 00:01:00 |
| Glass | 9 | 214 | 00:00:38 | 00:08:05 | **00:00:00** | 00:00:04 | 07:58:01 | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | 00:00:18 |
| Page-blocks | 10 | 5472 | 00:09:59 | 03:09:10 | **00:00:02** | 00:01:08 | - | 00:00:06 | 00:01:09 | 00:00:10 | 00:00:04 | 00:00:07 | 00:02:15 |
| Magic | 10 | 19020 | 01:03:25 | 13:04:02 | **00:00:21** | 00:16:30 | - | 00:00:28 | 00:00:50 | 00:01:18 | 00:01:20 | 00:00:52 | 00:17:52 |
| Wine | 13 | 178 | 00:00:12 | 00:10:16 | **00:00:00** | 00:00:45 | - | 00:04:45 | 00:43:04 | 00:15:58 | **00:00:00** | **00:00:00** | 00:00:09 |
| Heart | 13 | 270 | 00:00:15 | 00:12:51 | **00:00:00** | 00:00:12 | - | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | 00:00:14 |
| Cleveland | 13 | 303 | 00:01:29 | 00:14:36 | **00:00:00** | 00:00:09 | - | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | **00:00:00** | 00:00:37 |
| Vowel | 13 | 990 | 00:13:20 | 00:13:40 | **00:00:00** | 00:00:10 | - | 00:01:12 | 00:02:13 | 00:00:05 | 00:00:03 | **00:00:00** | 00:02:16 |
| Crx | 15 | 690 | 00:00:35 | 00:35:37 | **00:00:00** | 00:02:02 | - | 00:02:06 | 00:00:18 | 00:06:19 | 00:00:01 | **00:00:00** | 00:00:27 |
| Pen-based | 16 | 10992 | 02:32:58 | 10:29:28 | 00:00:13 | **00:00:07** | - | 00:00:29 | 00:00:52 | 00:01:24 | 00:00:36 | 00:00:27 | 00:45:08 |
| German | 20 | 1000 | 00:02:26 | 01:03:06 | **00:00:00** | 00:02:17 | - | 00:01:16 | 00:00:15 | 00:04:46 | 00:00:03 | **00:00:00** | 00:02:20 |
| Twonorm | 20 | 7400 | 00:39:37 | 09:29:46 | **00:00:07** | 05:03:33 | - | 00:00:49 | 00:00:39 | 00:00:13 | 00:00:14 | 00:00:10 | 00:15:46 |
| Ringnorm | 20 | 7400 | 00:08:04 | 10:05:13 | **00:00:07** | - | - | 00:02:12 | 00:01:14 | 00:13:33 | 00:00:16 | 00:00:31 | 00:07:28 |
| Wdbc | 30 | 569 | 00:00:30 | 00:50:01 | **00:00:00** | - | - | 00:00:09 | 00:00:05 | 01:01:00 | 00:00:01 | 00:00:01 | 00:00:51 |
| SatImage | 36 | 6435 | 01:57:23 | 04:07:42 | **00:00:13** | - | - | 00:00:43 | 00:00:24 | 00:04:50 | 00:00:54 | 00:00:18 | 00:35:08 |
| Texture | 40 | 5500 | 01:29:43 | 03:40:14 | **00:00:12** | - | - | 00:00:28 | 00:00:15 | 00:03:50 | 00:00:40 | 00:00:15 | 00:36:42 |
| Spectfheart | 44 | 267 | 00:00:31 | 00:14:30 | **00:00:00** | - | - | 00:00:09 | 00:00:05 | 00:02:39 | **00:00:00** | 00:00:10 | 00:01:59 |
| Spambase | 57 | 4597 | 00:42:58 | 13:01:47 | **00:00:10** | - | - | 00:00:15 | 00:00:12 | 00:24:55 | 00:00:13 | 00:00:27 | 00:14:36 |
| Sonar | 60 | 208 | 00:01:04 | 00:21:02 | 00:00:01 | 00:00:20 | - | 00:00:03 | 00:00:08 | 00:10:26 | 00:00:01 | **00:00:00** | 00:17:27 |
| Movementlibras | 90 | 360 | 00:10:45 | 05:17:58 | **00:00:01** | - | - | **00:00:01** | **00:00:01** | **00:00:01** | 00:00:06 | 00:00:27 | 01:20:10 |

2) When we use $Depth_{max} = 4$, we can see how the proposed approach does not obtain important improvement in training for three of the four datasets and only improves the results obtained in test for two of the four datasets. Moreover, the increase of the computational cost is high in all datasets, where $Depth_{max} = 3$ is a value with a good compromise between both properties.

On the other hand, Fig. 6 depicts the accuracy obtained over the training data along with different numbers of evaluations in the genetic process with $Depth_{max} = 4$. In this figure, we can highlight how this process obtains the best solution in less than 14 000 evaluations in all datasets because the initial RBs consist of a reduced number of rules.

Taking into account both studies, a good neutral choice ensuring the convergence may be to use 3 for $Depth_{max}$ and 15 000 for the number of evaluations in the genetic process, obtaining a good trade-off between accuracy and computational cost (good accuracy and not too great a computational cost).
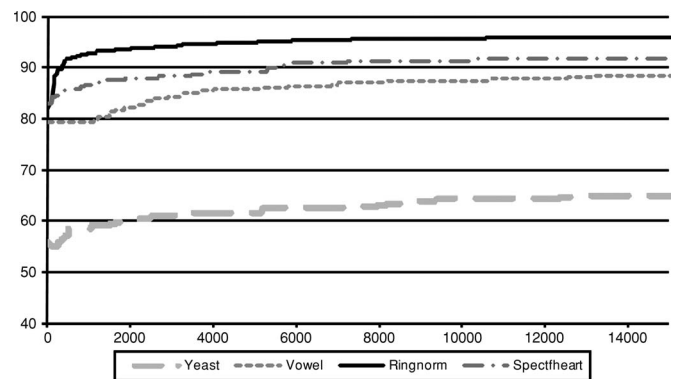


Fig. 6. Accuracy obtained over the training data with different numbers of evaluations in the genetic process with $Depth_{max} = 4$.

*E. Analysis of Scalability*

Table XV shows the average runtime of the analyzed methods in the previous sections on 26 real-world problems (with

a number of variables ranging from 4 to 90 and a number of patterns ranging from 150 to 19 020) and using the tenfold cross-validation model. The methods were implemented using Java, and all of the experiments were performed using a Pentium Core 2 Quad, 2.5-GHz CPU with 4 GB of memory and running Linux.

By the analysis of the results presented in Table XV, we can draw the following conclusions.

1) The SGERD algorithm presents a very low average run-time in all datasets, obtaining a good scalability when we increase the size of the problem. This method, however, should be the worst in Friedman's test when we compare the results obtained in the test data (see Table VI).

2) The 2SLAVE, FH-GBML, LAFAR, and CFAR algorithms expend a large amount of time when the number of attributes and patterns in the dataset is high. Notice that the CFAR and LAFAR cannot run in 7 and 18 of the 26 datasets, respectively.

3) The remainder of the methods obtain low computational costs in all datasets and present good results in accuracy. However, our proposal obtains the best ranking in Friedman's test when we compare the results that are obtained in the test partitions.

4) Notice that the CBA and CBA2 algorithms present similar runtimes to the CMAR and CPAR algorithms because they limit the total number of candidates rules that are generated in datasets with more than 15 variables since they cannot be completed within this limit.

5) The FARC-HD approach presents a good computational cost in all datasets, obtaining a good scalability and the best performance in accuracy.

## VI. CONCLUDING REMARKS

In this paper, we have proposed a new fuzzy associative classification method for high-dimensional datasets, named FARC-HD. Our aim was to obtain accurate and compact fuzzy associative classifiers with a low computational cost. To do this, we mine fuzzy association rules limiting the order of the associations in order to obtain a reduced set of candidate rules with less attributes in the antecedent. We have made use of a pattern weighting scheme in order to reduce the number of candidate rules, preselecting the rules with the best quality. A genetic rule selection and lateral tuning have been applied to select a small set of fuzzy association rules with a high classification accuracy.

Taking into account the results obtained, we can conclude that our model is a solid approach to deal with high-dimensional datasets, as it obtains the best accuracy in the experimental study. Moreover, the FARC-HD obtains models with a reduced number of rules (39.2 rules on average) and, particularly, with few attributes in the antecedent. Finally, the limit in the depth of the trees, along with candidate rule prescreening using the fuzzy measure wWRACC$''$, allows us to reduce the search space considerably. Thus, the genetic process for selection and tuning

does not introduce an excessive computational cost in to the whole process.

## REFERENCES

[1] H. Ishibuchi, T. Nakashima, and M. Nii, *Classification and Modeling with Linguistic Information Granules: Advanced Approaches to Linguistic Data Mining*. Berlin, Germany: Springer-Verlag, 2004.

[2] L. Kuncheva, *Fuzzy Classifier Design*. Berlin, Germany: Springer-Verlag, 2000.

[3] Y. Jin, W. Seelen, and B. Sendhoff, "Generating fc(3) fuzzy rule systems from data using evolution strategies," *IEEE Trans. Syst., Man Cybern. B, Cybern.*, vol. 29, no. 6, pp. 829–845, Dec. 1999.

[4] S. Ho, H. Chen, S. Ho, and T. Chen, "Design of accurate classifiers with a compact fuzzy-rule base using an evolutionary scatter partition of feature space," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 2, pp. 1031–1044, Apr. 2004.

[5] H. Wang, S. Kwong, Y. Jin, W. Wei, and K. F. Man, "Multi-objective hierarchical genetic algorithm for interpretable fuzzy rule-based knowledge extraction," *Fuzzy Sets Syst.*, vol. 149, pp. 149–186, 2005.

[6] Y. Zhang, X.-b. Wu, Z.-y. Xing, and W.-L. Hu, "On generating interpretable and precise fuzzy systems based on Pareto multi-objective cooperative co-evolutionary algorithm," *Appl. Soft Comput.*, vol. 11, pp. 1284–1294, 2011.

[7] C. Tsang, S. Kwong, and H. Wang, "Genetic-fuzzy rule mining approach and evaluation of feature selection techniques for anomaly intrusion detection," *Pattern Recognit.*, vol. 40, no. 9, pp. 2373–2391, 2007.

[8] T. Nakashima, G. Schaefer, and Y. Yokota, "A weighted fuzzy classifier and its application to image processing tasks," *Fuzzy Sets Syst.*, vol. 158, no. 3, pp. 284–294, 2007.

[9] W. Combs and J. Andrews, "Combinatorial rule explosion eliminated by a fuzzy rule configuration," *IEEE Trans. Fuzzy Syst.*, vol. 6, no. 1, pp. 1–11, Feb. 1998.

[10] Y. Jin, "Fuzzy modeling of high-dimensional systems: Complexity reduction and interpretability improvement," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 2, pp. 212–221, Apr. 2000.

[11] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd ed. San Fransisco, CA: Morgan Kaufmann, 2006.

[12] B. Liu, W. Hsu, and Y. Ma, "Integrating classification and association rule mining," in *Proc. Int. Conf. Knowl. Discov. Data Mining*, New York, 1998, pp. 80–86.

[13] B. Liu, Y. Ma, and C. Wong, "Classification using association rules: Weaknesses and enhancements," in *Data Mining for Scientific and Engineering Applications*, R. Grossman, C. Kamath, and V. Kumar, Eds. Norwell, MA: Kluwer, 2001, pp. 591–601.

[14] W. Li, J. Han, and J. Pei, "CMAR: Accurate and efficient classification based on multiple class-association rules," in *Proc. IEEE Int. Conf. Data Mining*, San Jose, CA, 2001, pp. 369–376.

[15] X. Yin and J. Han, "CPAR: Classification based on predictive association rules," in *Proc. 3rd SIAM Int. Conf. Data Mining*, San Francisco, CA, 2003, pp. 331–335.

[16] J. Li, G. Dong, K. Ramamohanarao, and L. Wong, "DeEPs: A new instance-based lazy discovery and classification system," *Mach. Learning*, vol. 54, no. 2, pp. 99–124, 2004.

[17] Y. Wang and A. Wong, "Boosting an associative classifier," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 7, pp. 988–992, Jul. 2006.

[18] F. Thabtah, "A review of associative classification mining," *Knowl. Eng. Rev.*, vol. 22, no. 1, pp. 37–65, 2007.

[19] R. Rak and L. K. M. Reformat, "A tree-projection-based algorithm for multi-label recurrent-item associative-classification rule generation," *Data Knowl. Eng.*, vol. 64, no. 1, pp. 171–197, 2008.

[20] C. Zhang and S. Zhang, *Association Rule Mining: Models and Algorithms Series* (Lecture Notes Computer Science Series 2307). Berlin, Germany: Springer-Verlag, 2002.

[21] V. Cherkasski and F. Mulier, *Learning from Data: Concepts, Theory, and Methods*. New York: Wiley-Interscience, 1998.

[22] P. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Boston, MA: Addison-Wesley/Longman, 2005.

[23] Y.-C. Hua and G.-H. Tzeng, "Elicitation of classification rules by fuzzy data mining," *Eng. Appl. Artif. Intell.*, vol. 16, no. 7–8, pp. 709–716, 2003.

[24] Y. Hu, R. Chen, and G. Tzeng, "Finding fuzzy classification rules using data mining techniques," *Pattern Recognit. Lett.*, vol. 24, no. 1–3, pp. 509–519, 2003.

[25] Y. Yi and E. Hullermeier, "Learning complexity-bounded rule-based classifiers by combining association analysis and genetic algorithms," in *Proc. 4th Conf. Eur. Soc. Fuzzy Logic Technol.*, Barcelona, Spain, 2005, pp. 47–52.

[26] Y.-C. Hua, "Determining membership functions and minimum fuzzy support in finding fuzzy association rules for classification problems," *Knowl.-Based Syst.*, vol. 19, no. 1, pp. 57–66, 2006.

[27] Z. Chen and G. Chen, "Building an associative classifier based on fuzzy association rules," *Int. J. Comput. Intell. Syst.*, vol. 1, no. 3, pp. 262–273, 2008.

[28] F. Pach, A. Gyenesei, and J. Abonyi, "Compact fuzzy association rule-based classifier," *Expert Syst. Appl.*, vol. 34, no. 4, pp. 2406–2416, 2008.

[29] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. New York: Addison-Wesley, 1989.

[30] J. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: Univ. Michigan Press, 1975.

[31] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. Int. Conf. Large Data Bases*, Santiago de Chile, Chile, Sep. 1994, pp. 487–499.

[32] K. Nozaki, H. Ishibuchi, and H. Tanaka, "Adaptive fuzzy rule-based classification systems," *IEEE Trans. Fuzzy Syst.*, vol. 4, no. 3, pp. 238–250, Aug. 1996.

[33] B. Kavsek and N. Lavrac, "APRIORI-SD: Adapting association rule learning to subgroup discovery," *Appl. Artif. Intell.*, vol. 20, no. 7, pp. 543–583, 2006.

[34] J. Casillas, O. Cordón, M. del Jesus, and F. Herrera, "Genetic tuning of fuzzy rule deep structures preserving interpretability for linguistic modeling," *IEEE Trans. Fuzzy Syst.*, vol. 13, no. 1, pp. 13–29, Feb. 2005.

[35] R. Alcalá, J. Alcalá-Fdez, and F. Herrera, "A proposal for the genetic lateral tuning of linguistic fuzzy systems and its interaction with rule selection," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 4, pp. 616–635, Aug. 2007.

[36] O. Cordón, F. Herrera, F. Hoffmann, and L. Magdalena, *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases*. (Advances in Fuzzy Systems—Applications and Theory). Singapore: World Scientific, 2001.

[37] O. Cordón, F. Gomide, F. Herrera, F. Hoffmann, and L. Magdalena, "Ten years of genetic fuzzy systems: Current framework and new trends," *Fuzzy Sets Syst.*, vol. 141, no. 1, pp. 5–31, 2004.

[38] F. Herrera, "Genetic fuzzy systems: Taxonomy, current research trends and prospects," *Evol. Intell.*, vol. 1, pp. 27–46, 2008.

[39] J. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kauffman, 1993.

[40] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learning Res.*, vol. 7, pp. 1–30, 2006.

[41] S. García and F. Herrera, "An extension on statistical comparisons of classifiers over multiple data sets for all pairwise comparisons," *J. Mach. Learning Res.*, vol. 9, pp. 2579–2596, 2008.

[42] S. García, A. Fernández, J. Luengo, and F. Herrera, "A study of statistical techniques and performance measures for genetics-based machine learning: Accuracy and interpretability," *Soft Comput.*, vol. 13, no. 10, pp. 959–977, 2009.

[43] S. García, D. Molina, M. Lozano, and F. Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: A case study on the CEC'2005 special session on real parameter optimization," *J. Heurist.*, vol. 15, pp. 617–644, 2009.

[44] H. Ishibuchi and T. Nakashima, "Effect of rule weights in fuzzy rule-based classification systems," *IEEE Trans. Fuzzy Syst.*, vol. 9, no. 4, pp. 506–515, Aug. 2001.

[45] H. Ishibuchi and T. Yamamoto, "Rule weight specification in fuzzy rule-based classification systems," *IEEE Trans. Fuzzy Syst.*, vol. 13, no. 4, pp. 428–435, Aug. 2005.

[46] O. Cordón, M. del Jesus, and F. Herrera, "A proposal on reasoning methods in fuzzy rule-based classification systems," *Int. J. Approximate Reason.*, vol. 20, no. 1, pp. 21–45, 1999.

[47] M. Delgado, N. Marín, D. Sánchez, and M. Vila, "Fuzzy association rules: General model and applications," *IEEE Trans. Fuzzy Syst.*, vol. 11, no. 2, pp. 214–225, Apr. 2003.

[48] D. Dubois, E. Hullermeier, and H. Prade, "A systematic approach to the assessment of fuzzy association rules," *Data Mining Knowl. Discov.*, vol. 13, no. 2, pp. 167–192, 2006.

[49] E. Hullermeier and Y. Yi, "In defense of fuzzy association analysis," *IEEE Trans. Syst., Man Cybern. B, Cybern.*, vol. 37, no. 4, pp. 1039–1043, Aug. 2007.

[50] T. Sudkamp, "Examples, counterexamples, and measuring fuzzy associations," *Fuzzy Sets Syst.*, vol. 149, no. 1, pp. 57–71, 2005.

[51] M. Kaya and R. Alhajj, "Utilizing genetic algorithms to optimize membership functions for fuzzy weighted association rules mining," *Appl. Intell.*, vol. 24, no. 1, pp. 7–15, 2006.

[52] T. Hong and Y. Lee, "An overview of mining fuzzy association rules," in *Studies in Fuzziness and Soft Computing*, vol. 220, H. Bustince, F. Herrera, and J. Montero, Eds. Berlin, Germany: Springer-Verlag, 2008, pp. 397–410.

[53] T. Hong, C. Chen, Y. Lee, and Y. Wu, "Genetic-fuzzy data mining with divide-and-conquer strategy," *IEEE Trans. Evol. Comput.*, vol. 12, no. 2, pp. 252–265, Apr. 2008.

[54] J. Alcalá-Fdez, R. Alcalá, M. Gacto, and F. Herrera, "Learning the membership function contexts for mining fuzzy association rules by using genetic algorithms," *Fuzzy Sets Syst.*, vol. 160, no. 7, pp. 905–921, 2009.

[55] F. Herrera and L. Martínez, "A 2-tuple fuzzy linguistic representation model for computing with words," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 6, pp. 746–752, Dec. 2000.

[56] L. Eshelman, "The CHC adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination," in *Foundations of Genetic Algorithms*, vol. 1, G. Rawlin, Ed. San Mateo, CA: Morgan Kaufmann, 1991, pp. 265–283.

[57] D. Whitley, S. Rana, J. Dzubera, and K. E. Mathias, "Evaluating evolutionary algorithms," *Artif. Intell.*, vol. 85, pp. 245–276, 1996.

[58] L. Eshelman, J. Schaffer, "Real-coded genetic algorithms and interval schemata," in *Foundations of Genetic Algorithms*, vol. 2, D. Whitley, Ed. San Mateo, CA: Morgan Kaufmann, 1993 pp. 187–202.

[59] M. Lozano, F. Herrera, N. Krasnogor, and D. Molina, "Real-coded memetic algorithms with crossover hill-climbing," *Evol. Comput.*, vol. 12, no. 3, pp. 273–302, 2004.

[60] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, S. García, L. Sánchez, and F. Herrera, "Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *J. Multiple-Valued Logic Soft Comput.*, vol. 17, nos. 2–3, pp. 255–287, 2011.

[61] J. Alcalá-Fdez, L. Sánchez, S. García, M. del Jesus, S. Ventura, J. Garrell, J. Otero, C. Romero, J. Bacardit, V. Rivas, J. Fernández, and F. Herrera, "KEEL: A software tool to assess evolutionary algorithms to data mining problems," *Soft Comput.*, vol. 13, no. 3, pp. 307–318, 2009.

[62] J. Quinlan, "Induction of decision trees," *Mach. Learning*, vol. 1, pp. 81–106, 1986.

[63] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: A frequent-pattern tree approach," *Data Mining Knowl. Discov.*, vol. 8, no. 1, pp. 53–87, 2004.

[64] A. González and R. Perez, "Selection of relevant features in a fuzzy genetic learning algorithm," *IEEE Trans. Syst., Man Cybern. B, Cybern.*, vol. 31, no. 3, pp. 417–425, Jun. 2001.

[65] A. González and R. Perez, "SLAVE: A genetic learning system based on an iterative approach," *IEEE Trans. Fuzzy Syst.*, vol. 7, no. 2, pp. 176–191, Apr. 1999.

[66] H. Ishibuchi, T. Yamamoto, and T. Nakashima, "Hybridization of fuzzy GBML approaches for pattern classification problems," *IEEE Trans. Syst., Man Cybern. B, Cybern.*, vol. 35, no. 2, pp. 359–365, Apr. 2005.

[67] E. Mansoori, M. Zolghadri, and S. Katebi, "Sgerd: A steady-state genetic algorithm for extracting fuzzy classification rules from data," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 4, pp. 1061–1071, Aug. 2008.

[68] U. Fayyad and K. Irani, "Multi-interval discretization of continuous-valued attributes for classification learning," in *Proc. 13th Int. Joint Conf. Uncertainly Artif. Intell.*, 1993, pp. 1022–1029.

[69] D. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*. London, U.K./Boca Raton, FL: Chapman & Hall/CRC, 2003.

[70] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics*, vol. 1, no. 6, pp. 80–83, 1945.

[71] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *J. Amer. Statist. Assoc.*, vol. 32, no. 200, pp. 675–701, 1937.

[72] R. Iman and J. Davenport, "Approximations of the critical region of the friedman statistic," *Commun. Statist. A, Theory Methods*, vol. 9, pp. 571–595, 1980.

[73] S. Holm, "A simple sequentially rejective multiple test procedure," *Scand. J. Statist.*, vol. 6, pp. 65–70, 1979.

**Jesús Alcalá-Fdez** received the M.Sc. and the Ph.D. degrees in computer science, both from the University of Granada, Granada, Spain, in 2002 and 2006, respectively.

He was with the Department of Computer Science, University of Jaén, Jaén, Spain, from 2005 to 2007. He is currently an Associate Professor with the Department of Computer Science and Artificial Intelligence, University of Granada, where he is a member of the Soft Computing and Intelligent Information Systems Research Group. He has published more than 50 papers in international journals, book chapters, and conferences. He was involved in several research projects supported by the Spanish Government and the Andalusian Government. His current research interests include fuzzy association rules, genetic fuzzy systems, and data mining software.

Dr. Alcalá-Fdez co-edited the IEEE TRANSACTIONS ON FUZZY SYSTEMS Special Issue on "Genetic Fuzzy Systems: What's Next" and the *Journal of Multiple-Valued Logic and Soft Computing* Special Issue on "Soft Computing Techniques in Data Mining." He currently serves as a member of the editorial board of the *International Journal on Advances in Information Sciences and Service Sciences*. He has been the President of the "Software Fuzzy Systems" Task Force of the Fuzzy Systems Technical Committee with the IEEE Computational Intelligence Society since March 2011. He was the Publicity-Chair for the Conference of Genetic and Evolutionary Fuzzy Systems in 2010 and 2011 and Special Sessions and Workshops-Chair at the Conference of Intelligent Systems Design and Applications in 2011.

**Rafael Alcalá** received the M.Sc. and the Ph.D. degrees in computer science, both from the University of Granada, Granada, Spain, in 1998 and 2003, respectively.

He was with Department of Computer Science, University of Jaén, Jaén, Spain, from 1998 to 2003. He is currently an Associate Professor with the Department of Computer Science and Artificial Intelligence, University of Granada, where he is a member of the Soft Computing and Intelligent Information Systems Research Group. He has published more than 75 papers in international journals, book chapters, and conferences. He was involved in several research projects supported by the Spanish Government and the European Union. His current research interests include multiobjective genetic algorithms and genetic fuzzy systems, particularly, the learning/tuning of fuzzy systems for modeling, and control with a good trade-off between accuracy and interpretability, as well as fuzzy association rules.

Dr. Alcalá co-edited the IEEE TRANSACTIONS ON FUZZY SYSTEMS Special Issue on "Genetic Fuzzy Systems: What's next," the *Evolutionary Intelligence* Special Issue on "Genetic Fuzzy Systems: New Advances," and the *Soft Computing* Special Issue on "Evolutionary Fuzzy Systems." He currently serves as a member of the editorial/reviewer board of the *International Journal of Computational Intelligence Research*, the *Journal of Advanced Research in Fuzzy and Uncertain Systems*, the *Journal of Universal Computer Science,* and *Applied Intelligence*. He is a member of the Fuzzy Systems Technical Committee of the IEEE Computational Intelligence Society and has been the President of the "Genetic Fuzzy Systems" Task Force since January 2009. He was the Program Co-Chair at the Conference of Genetic and Evolutionary Fuzzy Systems in 2010, the General Co-Chair at the Conference of Genetic and Evolutionary Fuzzy Systems in 2011, and the Area Co-Chair at the IEEE International Conference on Fuzzy Systems in 2011.

**Francisco Herrera** (M'10) received the M.Sc. and the Ph.D. degrees, both in mathematics, from the University of Granada, Granada, Spain, in 1988 and 1991, respectively.

He is currently a Professor with the Department of Computer Science and Artificial Intelligence, the University of Granada. He has published more than 200 papers in international journals. He is a coauthor of the book *Genetic Fuzzy Systems: Evolutionary Tuning and Learning of Fuzzy Knowledge Bases* (Singapore: World Scientific, 2001). His current research interests include computing with words and decision making, bibliometrics, data mining, data preparation, instance selection, fuzzy rule-based systems, genetic fuzzy systems, knowledge extraction based on evolutionary algorithms, and memetic and genetic algorithms.

Dr. Herrera has co-edited five international books and 21 Special Issues in international journals on different Soft Computing topics. He currently acts as the Editor-in-Chief of the international journal *Progress in Artificial Intelligence* (Springer) and serves as an Area Editor of the journal *Soft Computing* (area of evolutionary and bioinspired algorithms). He acts as an Associated Editor of the journals IEEE TRANSACTIONS ON FUZZY SYSTEMS, *Information Sciences*, *Advances in Fuzzy Systems*, and the *International Journal of Applied Metaheuristics Computing*. He also serves as a member of several journal editorial boards, including *Fuzzy Sets and Systems*, *Applied Intelligence*, *Knowledge and Information Systems*, *Information Fusion*, *Evolutionary Intelligence*, the *International Journal of Hybrid Intelligent Systems*, *Memetic Computation*, *Swarm,* and *Evolutionary Computation*. He has received the following honors and awards: the European Coordinating Committee for Artificial Intelligence Fellow in 2009, the 2010 Spanish National Award on Computer Science AR-ITMEL to the "Spanish Engineer on Computer Science," and the International Cajastur "Mamdani" Prize for Soft Computing (Fourth Edition, 2010).