International Journal of Approximate Reasoning 50 (2009) 561–577

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

journal homepage: www.elsevier.com/locate/ijar

Hierarchical fuzzy rule based classification systems with genetic rule selection for imbalanced data-sets ☆

Alberto Fernández [a,*], María José del Jesus [b], Francisco Herrera [a]

[a] Dept. of Computer Science and Artificial Intelligence, University of Granada, Periodista Daniel Saucedo Aranda s/n, 18071 Granada, Spain
[b] Dept. of Computer Science, University of Jaén, Spain

ARTICLE INFO

Article history:
Received 20 February 2008
Received in revised form 24 November 2008
Accepted 24 November 2008
Available online 6 December 2008

Keywords:
Classification
Fuzzy rule based classification systems
Imbalanced data-sets
Genetic fuzzy systems
Genetic rule selection
Hierarchical fuzzy partitions

ABSTRACT

In many real application areas, the data used are highly skewed and the number of instances for some classes are much higher than that of the other classes. Solving a classification task using such an imbalanced data-set is difficult due to the bias of the training towards the majority classes.

The aim of this paper is to improve the performance of fuzzy rule based classification systems on imbalanced domains, increasing the granularity of the fuzzy partitions on the boundary areas between the classes, in order to obtain a better separability. We propose the use of a hierarchical fuzzy rule based classification system, which is based on the refinement of a simple linguistic fuzzy model by means of the extension of the structure of the knowledge base in a hierarchical way and the use of a genetic rule selection process in order to get a compact and accurate model.

The good performance of this approach is shown through an extensive experimental study carried out over a large collection of imbalanced data-sets.

1. Introduction

Throughout the last years, the classification problem in the framework of imbalanced data-sets has been identified as an important problem in Data Mining [8,46]. This problem occurs when the number of instances of one class is much lower than the instances of the other classes. This phenomenon is growing in importance since it appears in most of the real domains of classification such as fraud detection [16], detection of oil spills from satellite images [31], prediction of pre-term births [21], or medical diagnosis [5].

When learning from imbalanced data-sets, the tendency is that the classifier might obtain a high predictive accuracy over the majority class, but might predict poorly over the minority class [43]. Furthermore, the minority class examples can be treated as noise and they can be completely ignored by the classifier. There are studies that show that most classification methods lose their classification ability when dealing with imbalanced data [30,33].

Our previous work on the topic [18] showed the good behaviour obtained by fuzzy rule based classification systems (FRBCSs) in the framework of imbalanced data-sets, by means of the application of a preprocessing step in order to balance the training data before the rule generation phase. We determined the robustness of this approach specially when increasing the imbalance degree.

☆ This work was supported in part by the Spanish Ministry of Education and Science (MEC) under Projects TIN-2005-08386-C05-01 and TIN-2005-08386-C05-03.
* Corresponding author. Tel.: +34 958 240598; fax: +34 958 243317.
E-mail addresses: alberto@decsai.ugr.es (A. Fernández), mjjesus@ujaen.es (M.J. del Jesus), herrera@decsai.ugr.es (F. Herrera).
URLs: http://sci2s.ugr.es (A. Fernández), http://wwwdi.ujaen.es (M.J. del Jesus), http://sci2s.ugr.es (F. Herrera).

0888-613X/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.ijar.2008.11.004

In this paper, we propose a hierarchical environment to improve the behaviour of linguistic FRBCSs. This approach preserves the original descriptive power and increases its accuracy by reinforcing those problem subspaces that are specially difficult. Therefore, we focus our efforts in enhancing the classification performance in the boundary areas of the problem, obtaining a good separability between the minority and majority classes.

We consider the modification of the knowledge base (KB) structure using the concept of "layers" that was introduced in [12], defined by the authors as hierarchical knowledge base (HKB). We propose a two-level learning method for obtaining a hierarchical fuzzy rule base classification system (HFRBCS) by means of two processes:

1. A linguistic rule generation (LRG) method is used to create the initial rule base (RB), from which we extract the hierarchical rule base (HRB).
2. A genetic algorithm (GA) is employed to select the best cooperative rules from the HRB.

This type of models are usually known as genetic fuzzy systems [23], which are an emerging tool during the last years with very good results from the optimization point of view of fuzzy models [1,10,36].

To obtain the initial linguistic fuzzy models, we will employ a simple inductive LRG-method, the Chi et al.'s method [9], that extends the well-known Wang and Mendel method [42] to classification problems. According to the decisions taken in our previous work [18], we will use triangular membership functions for the fuzzy partitions and rule weights in the consequent of the rules. We will also apply a re-sampling procedure to prepare the training data for the learning process, specifically using the "Synthetic Minority Over-sampling Technique" (SMOTE) [7]. In any case, we will also study the effect of preprocessing in the performance of HFRBCSs by contrasting the results obtained using the original data-sets against the ones obtained with the SMOTE algorithm.

We will analyze the behaviour of our HFRBCS proposal comparing its results with a linguistic FRBCS generated by a common approach [29], and a new one, the E-Algorithm [45], which is an extension of the previous method to generate an FRBCS adapted to imbalanced data-sets. We will also include the C4.5 decision tree [35] in our experimental study; thus, we will show that the HFRCBS is a very robust method in the framework of imbalanced data-sets when compared not only with other fuzzy systems, but also with a well-known machine learning algorithm. Furthermore, in this study we make use of some non-parametric tests [13] for statistical comparisons of the performance of these classifiers.

For the empirical analysis, we have considered 44 data-sets from UCI repository [2], making a division between two degrees of imbalance (low and high imbalance) according to the imbalance ratio (IR) [32], which is defined as the ratio of the number of instances of the majority class and the minority class. Multi-class data-sets are modified to obtain two-class non-balanced problems, defining the joint of one or more classes as positive and the joint of one or more classes as negative.

This paper is set up as follows. Section 2 introduces the imbalanced data-set problem, describing the preprocessing technique for imbalanced data-sets used in this work and discussing the evaluation metric used for this type of data. In Section 3, we describe our proposal and we present a methodology to automatically design an HFRBCS from a generic LRG-method in the framework of imbalanced data-sets. In Section 4 we include our experimental analysis where we first analyze the effect of preprocessing, and then we compare the performance of our model with the remaining FRBCSs methods and with C4.5 in order to validate our results in imbalanced data-sets with different IR. In Section 5 some concluding remarks are pointed out. Finally, we include two appendices with the description of the non-parametric tests used in our study and the detailed results for the experiments carried out in the experimental study, respectively.

2. Imbalanced data-sets in classification

In this section, we will first introduce the problem of imbalanced data-sets. Then, we will describe the preprocessing technique that we have applied in order to deal with the imbalanced data-sets: the SMOTE algorithm [7]. Finally, we will present the evaluation metrics for this kind of classification problem.

2.1. The problem of imbalanced data-sets

Learning from imbalanced data is an important topic that has recently appeared in the machine learning community. When treating with imbalanced data-sets, one or more classes might be represented by a large number of examples whereas the others are represented by only a few.

We focus on the binary-class imbalanced data-sets, where there is only one positive and one negative class. We consider the positive class as the one with the lowest number of examples and the negative class the one with the highest number of examples. Furthermore, in this work we use the IR [32], defined as the ratio of the number of instances of the majority class and the minority class, to organize the different data-sets according to their IR.

The problem of imbalanced data-sets is extremely significant because it is implicit in most real world applications, such as fraud detection [16], text classification [41], risk management [25] or medical applications [22].

In classification, this problem (also named the "class imbalance problem") will cause a bias on the training of classifiers and will result in the lower sensitivity of detecting the minority class examples. For this reason, a large number of approaches have been previously proposed to deal with the class imbalance problem. These approaches can be categorized into two groups: the internal approaches that create new algorithms or modify existing ones to take the class imbalance problem into consideration [3,45] and external approaches that preprocess the data in order to diminish the effect cause by their class imbalance [4,15].

The internal approaches have the disadvantage of being algorithm specific, whereas external approaches are independent of the classifier used and are, for this reason, more versatile. Furthermore, in our previous work on this topic [18] we analyzed the cooperation of some preprocessing methods with FRBCSs, showing a good behaviour for the over-sampling methods, specially in the case of the SMOTE methodology.

According to this, we will employ in this paper the SMOTE algorithm in order to deal with the problem of imbalanced data-sets. This method is detailed in the next subsection.

2.2. Preprocessing imbalanced data-sets. The SMOTE algorithm

As mentioned before, applying a preprocessing step in order to balance the class distribution is a positive solution to the imbalance data-set problem [4]. Specifically, in this work we have chosen an over-sampling method which is a reference in this area: the SMOTE algorithm [7].

In this approach the minority class is over-sampled by taking each minority class sample and introducing synthetic examples along the line segments joining any/all of the k minority class nearest neighbours. Depending upon the amount of over-sampling required, neighbours from the k-nearest neighbours are randomly chosen. This process is illustrated in Fig. 1, where x_i is the selected point, x_{i1} to x_{i4} are some selected nearest neighbours and r_1 to r_4 the synthetic data points created by the randomized interpolation. The implementation employed in this work uses only one nearest neighbour using the euclidean distance, and balance both classes to the 50% distribution.

Synthetic samples are generated in the following way: take the difference between the feature vector (sample) under consideration and its nearest neighbour. Multiply this difference by a random number between 0 and 1, and add it to the feature vector under consideration. This causes the selection of a random point along the line segment between two specific features. This approach effectively forces the decision region of the minority class to become more general. An example is detailed in Fig. 2.

In short, its main idea is to form new minority class examples by interpolating between several minority class examples that lie together. Thus, the overfitting problem is avoided and causes the decision boundaries for the minority class to spread further into the majority class space.

2.3. Evaluation in imbalanced domains

The measures of the quality of classification are built from a confusion matrix (shown in Table 1) which records correctly and incorrectly recognized examples for each class.

The most used empirical measure, accuracy (1), does not distinguish between the number of correct labels of different classes, which in the framework of imbalanced problems may lead to erroneous conclusions. For example a classifier that obtains an accuracy of 90% in a data-set with an IR value of 9, might not be accurate if it does not cover correctly any minority class instance.

$$Acc = \frac{TP + TN}{TP + FN + FP + TN}. \quad (1)$$

Because of this, instead of using accuracy, more correct metrics are considered. Two common measures, sensitivity and specificity (2,3), approximate the probability of the positive (negative) label being true. In other words, they assess the effectiveness of the algorithm on a single class.

$$\text{sensitivity} = \frac{TP}{TP + FN}, \quad (2)$$

$$\text{specificity} = \frac{TN}{FP + TN}. \quad (3)$$

The metric used in this work is the geometric mean of the true rates [3], which can be defined as

$$GM = \sqrt{\frac{TP}{TP + FN} \cdot \frac{TN}{FP + TN}}. \quad (4)$$

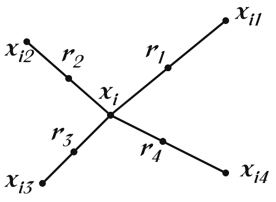


Fig. 1. An illustration on how to create the synthetic data points in the SMOTE algorithm.

```
Consider a sample (6,4) and let (4,3) be its nearest neighbour.
(6,4) is the sample for which k-nearest neighbours are being
identified (4,3) is one of its k-nearest neighbours.
Let: f1_1 = 6 f2_1 = 4, f2_1 - f1_1 = -2
f1_2 = 4 f2_2 = 3, f2_2 - f1_2 = -1
The new samples will be generated as
f1',f2') = (6,4) + rand(0-1) * (-2,-1)
rand(0-1) generates a random number between 0 and 1.
```

Fig. 2. Example of the SMOTE application.

This metric attempts to maximize the accuracy of each one of the two classes with a good balance. It is a performance metric that links both objectives.

3. Hierarchical fuzzy rule based classification system

In this section we will describe our algorithm proposal to obtain an HFRBCS, which is based on two processes:

1. HKB generation process: An HRB is created from a simple RB obtained by an LRG-method.
2. HRB genetic selection process: The best cooperative rules are selected by means of a GA.

In the following subsections we will first introduce the type of rules, rule weights and inference model used in this work. Next, we will describe each one of processes to obtain an HFRBCS, explaining in detail all their characteristics.

3.1. Fuzzy rule based classification systems

Any classification problem consists of m training patterns $x_p = (x_{p1}, \ldots, x_{pn})$, $p = 1, 2, \ldots, m$ from M classes where x_{pi} is the ith attribute value ($i = 1, 2, \ldots, n$) of the pth training pattern.

In this work we use fuzzy rules of the following form for our FRBCSs:

$$\text{Rule } R_j: \text{ If } x_1 \text{ is } A_{j1} \text{ and } \ldots \text{ and } x_n \text{ is } A_{jn} \text{ then Class} = C_j \text{ with } RW_j, \tag{5}$$

where R_j is the label of the jth rule, $x = (x_1, \ldots, x_n)$ is an n-dimensional pattern vector, A_{ji} is an antecedent fuzzy set (we use triangular membership functions), C_j is a class label, and RW_j is the rule weight.

In the specialized literature rule weights have been used in order to improve the performance of FRBCSs [27]. In this work, following the conclusions extracted in [18], we employ as heuristic method for the rule weight the penalized certainty factor [29]:

$$RW_j = \frac{\sum_{x_p \in ClassC_j} \mu_{A_j}(x_p)}{\sum_{p=1}^{m} \mu_{A_j}(x_p)} - \frac{\sum_{x_p \notin ClassC_j} \mu_{A_j}(x_p)}{\sum_{p=1}^{m} \mu_{A_j}(x_p)}. \tag{6}$$

We use the fuzzy reasoning method (FRM) of the winning rule (classical approach) [11] for classifying new patterns by the RB. The single winner rule R_w is determined for a new pattern $x_p = (x_{p1}, \ldots, x_{pn})$ as

$$\mu_w(x_p) \cdot RW_w = max\{\mu_j(x_p) \cdot RW_j\}; x_p \in X, \quad j = 1 \ldots L. \tag{7}$$

The new pattern x_p is classified as Class C_w, which is the consequent class of the winner rule R_w. If multiple fuzzy rules have the same maximum value but different consequent classes for the new pattern x_p in (7), the classification of x_p is rejected. The classification is also rejected if no fuzzy rule is compatible with the new pattern x_p.

3.2. Hierarchical systems of linguistic rules

This approach presents a more flexible KB structure that allows to improve the accuracy of the FRBCSs without losing their interpretability: the HKB, which is composed of a hierarchical data base (HDB) and an HRB.

Table 1
Confusion matrix for a two-class problem.

	Positive prediction	Negative prediction
Positive class	True positive (TP)	False negative (FN)
Negative class	False positive (FP)	True negative (TN)

The description of the HKB and the two-level learning method to generate an HFRBCS are introduced in the following two subsections.

3.2.1. Hierarchical knowledge base

The HKB [12] is composed of a set of layers, and each layer is defined by its components in the following way:

$$layer(t, n(t)) = DB(t, n(t)) + RB(t, n(t)), \tag{8}$$

with $n(t)$ being the number of linguistic terms in the fuzzy partitions of layer t,$DB(t,n(t))$ being the data base (DB) which contains the linguistic partitions with granularity level $n(t)$ of layer t (t-linguistic partitions), and $RB(t,n(t))$ being the RB formed by those linguistic rules whose linguistic variables take values in $DB(t,n(t))$ (t-linguistic rules). For the sake of simplicity in the descriptions, the following notation equivalences are established:

$$DB(t, n(t)) \equiv DB^t \text{ and } RB(t, n(t)) \equiv RB^t. \tag{9}$$

At this point, we should note that, in this work, we are using *linguistic partitions* with the same number of linguistic terms for all input variables, composed of symmetrical triangular-shaped and uniformly distributed membership functions (see Fig. 1). The number of linguistic terms in the t-linguistic partitions is defined in the following way:

$$n(t) = (n(1) - 1) \cdot 2^{t-1} + 1, \tag{10}$$

with $n(1)$ being the granularity of the initial fuzzy partitions.

Fig. 3 (left) graphically depicts the way in which a linguistic partition in DB^1 becomes a linguistic partition in DB^2. Each term of order k from DB^t, $S_k^{n(t)}$ ($S_k^{n(1)}$ in the figure), is mapped into the fuzzy set $S_{2k-1}^{2\cdot n(t)-1}$, preserving the former modal points, and a set of $n(t) - 1$ new terms is created, each one between $S_k^{n(t)}$ and $S_{k+1}^{n(t)}$ ($k = 1, \ldots, n(t) - 1$) (see Fig. 3 right).

The main purpose of developing an HRB is to divide the problem space in a more accurate way. To do so, those linguistic rules from $RB(t,n(t))$ – RB^t – that classify a subspace with bad performance are expanded into a set of more specific linguistic rules, which become their image in $RB(t + 1, 2 \cdot n(t) - 1)$ – RB^{t+1} – this set of rules classify the same subspace that the former one and replaces it. As a consequence of the previous definitions, we could now define the HKB as the union of every layer t:

$$HKB = \cup_t layer(t, n(t)). \tag{11}$$

In this paper, we will just consider a two-layer HKB which allows us to produce a refinement of simple FRBCS to increase their accuracy, preserving their structure and descriptive power, and reinforcing only the classification of those problem subspaces with more difficulties by a hierarchical treatment of the rules generated in these zones.

3.2.2. Two-level learning method for building HFRBCSs

In this subsection, we present the two-level learning method to generate two-layer HKBs [12]. To do so, we consider the existence of a set X of m training patterns $x_p = (x_{p1}, \ldots, x_{pn})$, $p = 1, 2, \ldots, m$ from M classes where x_{pi} is the ith attribute value ($i = 1, 2, \ldots, n$) of the pth training pattern.

We use an existing inductive LRG-method and a previously defined DB^1. Specifically, we consider as LRG-method the Chi et al. [9] approach, that will lead us to obtain simple linguistic fuzzy models, although any other technique could be used.

Two measures of error are used in the algorithm: a global measure, which is used to evaluate the complete RB, and a local measure, used to determine if an individual rule is expanded. Their expressions are defined below:

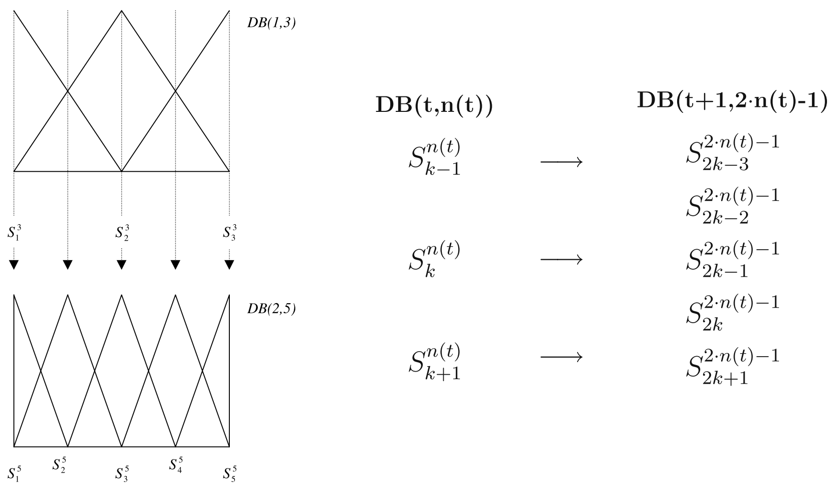


Fig. 3. Two-layers of linguistic partitions which compose the HDB and mapping between terms from successive DBs.

1. **Global measure**. We will employ the accuracy per class (sensitivity or specificity), computed as:

$$Acc_i(X_i, RB) = \frac{|\{x_p \in X_i / FRM(x_p, RB) = Class(x_p)\}|}{|X_i|}, \tag{12}$$

where $|\cdot|$ is the number of patterns, with X_i being the subset of examples of the ith class ($i \in 1 \ldots M$), $FRM(x_p,RB)$ is the output class computed following the fuzzy reasoning process using the current RB and $Class(x_p)$ is the class label for example x_p.

2. **Local measure**. The accuracy for a simple rule, $R_j^{n(1)}$, calculated over X, is showed as follows:

$$Acc(X, R_j^{n(1)}) = \frac{\left|X^+\left(R_j^{n(1)}\right)\right|}{\left|X\left(R_j^{n(1)}\right)\right|}, \tag{13}$$

$$X^+\left(R_j^{n(1)}\right) = \left\{x_p \in X / \mu_{R_j^{n(1)}}(x_p) > 0 \text{ and } Class(x_p) = Class\left(R_j^{n(1)}\right)\right\}, \tag{14}$$

$$X\left(R_j^{n(1)}\right) = \{x_p \in X / \mu_{R_j^{n(1)}}(x_p) > 0\}, \tag{15}$$

where $Class(\cdot)$ is a function that provides the class label for a pattern, or for a rule. We must note that $X^+\left(R_j^{n(1)}\right)$ and $X\left(R_j^{n(1)}\right)$ only include those examples that the rule actually classifies, because we are using as FRM the winning rule approach.

Now we will describe the HKB generation process (summarized in Table 2), which basically consists of the following steps:

Step 0: RB^1 *Generation*. Generate the rules from DB^1 by means of an existing LRG-method: $RB^1 = LRG - method(DB^1, X)$.

Step 1: RB^2 *Generation*. Generate RB^2 from RB^1, DB^1 and DB^2.

(a) *Calculate the global error of* RB^1 per class: $Acc_i(X_i, RB^1), i = 1, \ldots, M$.

(b) *Calculate the local error of each 1-linguistic rule:* $Acc(X, R_j^{n(1)})$.

(c) *Select the 1-linguistic rules with bad performance which will be expanded* (*the expansion factor* α *may be adapted in order to have more or less expanded rules*):

$$\begin{aligned} &\text{If } Acc\left(X, R_j^{n(1)}\right) \leqslant (1-\alpha)\cdot Acc_i(X_i, RB^1) \text{Then } R_j^{n(1)} \in RB^1_{bad} \\ &\text{Else } R_j^{n(1)} \in RB^1_{good}, \end{aligned} \tag{16}$$

where $Class(R_j^{n(1)}) = i$.

(d) *Create* DB^2.

(e) *For each bad performance 1-linguistic rule to be expanded*, $R_j^{n(1)} \in RB_{bad}$:

(i) *Select the 2-linguistic partitions terms* from DB^2 for each rule. For all linguistic terms considered in $R_j^{n(1)}$, i.e., $S_{jk}^{n(1)}$ defined in DB^1, select those terms $S_h^{2\cdot n(1)-1}$ in DB^2 that significantly intersect them. We consider that two linguistic terms have a "*significant intersection*" between each other, if the maximum cross level between their fuzzy sets in a linguistic partition overcomes a predefined threshold δ:

$$I\left(S_{jk}^{n(1)}\right) = \left\{S_h^{2\cdot n(1)-1} \in DB^2 / \max_{u \in U_k} \min\left\{\mu_{S_{jk}^{n(1)}}(u), \mu_{S_h^{2\cdot n(1)-1}}(u)\right\} \geqslant \delta\right\}, \tag{17}$$

where $\delta \in [0,1]$.

(ii) *Combine the previously selected s sets* $I\left(S_{jk}^{n(1)}\right)$ *by the following expression:*

$$I\left(R_j^{n(1)}\right) = I\left(S_{j1}^{n(1)}\right) \times \cdots \times I\left(S_{js}^{n(1)}\right). \tag{18}$$

(iii) *Extract 2-linguistic rules*, which are the expansion of the bad *1-linguistic rule* $R_j^{n(1)}$. This task is performed by the LRG-method, which takes $I\left(R_j^{n(1)}\right)$ and the set of examples $X\left(R_j^{n(1)}\right)$ as its parameters:

$$\text{CLR}(R_j^{n(1)}) = \text{LRG-method}\left(I\left(R_j^{n(1)}\right),\quad X\left(R_j^{n(1)}\right)\right) = \left\{R_{j_1}^{2\cdot n(1)-1}, \ldots, R_{j_L}^{2\cdot n(1)-1}\right\} \tag{19}$$

Table 2
Two-level learning method.

Hierarchical knowledge base generation process
Step 0. RB(1, n(1)) Generation process
Step 1. RB(2,2 $\cdot$ n(1)-1) Generation process
Step 2. Summarization process
Hierarchical rule base genetic selection process
Step 3. HRB genetic selection process

with $\text{CLR}\left(R_j^{n(1)}\right)$ being the image of the expanded linguistic rule $R_j^{n(1)}$, i.e., the candidates to be in the HRB from rule $R_j^{n(1)}$.

Step 2: Summarization. Obtain a Joined set of Candidate linguistic rules (JCLR), performing the union of the group of the new generated *2-linguistic rules* and the former good performance *1-linguistic rules:*

$$\text{JCLR} = \text{RB}^1_{\text{good}} \cup (\cup_j \text{CLR}(R_j^{n(1)})), \quad R_j^{n(1)} \in RB^1_{\text{bad}}.$$

Example. In the following, we show an example of the whole expansion process. Let us consider $n(1) = 3$ and the following linguistic partitions:

$$DB(1,3) = \{S^3, M^3, L^3\},$$
$$DB(2,5) = \{VS^5, S^5, M^5, L^5, VL^5\},$$

where *S* stands for Small, *M* for Medium, *L* for Large, and *V* for Very. Let us consider the following bad performance *1-linguistic rule* to be expanded (see Fig. 4):

$$R_i^3 : \text{IF } x_1 \textit{ is } S_{i1}^3 \text{ AND } x_2 \textit{ is } S_{i2}^3 \text{ THEN } Class = C \text{ with } RW_i,$$

where the linguistic terms are, $S_{i1}^3 = S^3, S_{i2}^3 = S^3$, and the resulting sets *I* with $\delta = 0.5$ are:

$$I(S_{i1}^3) = \{VS^5, S^5\}, \quad I(S_{i2}^3) = \{VS^5, S^5\},$$
$$I(R_i^3) = I(S_{i1}^3) \times I(S_{i2}^3).$$

Therefore, it is possible to obtain at most four *2-linguistic rules* generated by the LRG-method from the expanded R_i^3:

$$LRG(I(R_i^3), \quad X(R_i^3)) = \left\{R_{i1}^5, R_{i2}^5, R_{i3}^5, R_{i4}^5\right\}.$$

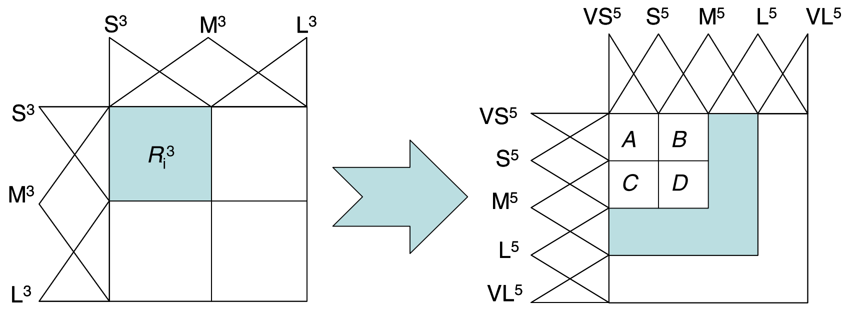
This example is graphically showed in Fig. 4. In the same way, other bad performance neighbour rules could be expanded simultaneously.

Step 3: HRB Selection. Simplify the set JCLR by removing the unnecessary rules from it and generating an HRB with good cooperation. In *JCLR* – where rules of different hierarchical layers coexist-, it may happen that a complete set of *2-linguistic rules* which replaces an expanded *1-linguistic rule* does not produce good results. However, a subset of this set of *2-linguistic rules* may work properly. A genetic process is considered to put this task into effect, which is explained on detail in the next subsection.

$$\text{HRB} = \text{Selection Process (JCLR)}.$$

After applying this algorithm, the HKB is obtained as:

$$\text{HKB} = \text{HDB} + \text{HRB}.$$



$$R_i^3 = \text{IF } x_1 \text{ is } S^3 \text{ AND } x_2 \text{ is } S^3 \text{ THEN } Class = C \text{ with } RW_i$$

$$R_{i1}^5 = \text{IF } x_1 \text{ is V}S^5 \text{ AND } x_2 \text{ is V}S^5 \text{ THEN } Class = C \text{ with } RW_{i1}$$
$$R_{i2}^5 = \text{IF } x_1 \text{ is V}S^5 \text{ AND } x_2 \text{ is } S^5 \text{ THEN } Class = C \text{ with } RW_{i2}$$
$$R_{i3}^5 = \text{IF } x_1 \text{ is } S^5 \text{ AND } x_2 \text{ is V}S^5 \text{ THEN } Class = C \text{ with } RW_{i3}$$
$$R_{i4}^5 = \text{IF } x_1 \text{ is } S^5 \text{ AND } x_2 \text{ is } S^5 \text{ THEN } Class = C \text{ with } RW_{i4}$$

Fig. 4. Example of the HRB generation process.

Remark 1. *About repeated 2-linguistic rules.* As a consequence of the previous DB^2 generation policy, which is based on selecting those terms in DB^2 which significantly intersect the ones of the bad rule, repeated 2-linguistic rules can be generated as a consequence of the expansion of adjacent bad 1-linguistic rules. If they are exactly the same we will eliminate one of the rules. On the other hand, if they have a different class in their consequent part, the rule with a higher rule weight remains in the RB whereas the other is removed.

3.3. Hierarchical rule base genetic rule selection process

In the previous section we have mentioned that an excessive number of rules may not produce a good performance and it makes difficult to understand the model behaviour. We may find different types of rules in a large fuzzy rule set: irrelevant rules, which do not contain significant information; redundant rules, whose actions are covered by other rules; erroneous rules, which are wrong defined and distort the performance of the FRBCS; and conflicting rules, which perturb the performance of the FRBCS when they coexist with others.

In this work, we consider the CHC genetic model [14] in order to make the rule selection process, since it has achieved good results for binary selection problems [6]. In the following, the main characteristics of this genetic approach are presented.

1. *Coding scheme and initial gene pool:* It is based on a binary coded GA where each gene indicates whether a rule is selected or not (alleles '1' or '0', respectively). Considering that N rules are contained in the preliminary/candidate rule set, the chromosome $C = (c_1, \ldots, c_N)$ represents a subset of rules composing the final HRB, such that:

   $$IF \quad c_i = 1 \quad THEN \quad (R_i \in HRB) \quad ELSE \quad (R_i \notin \mathrm{HRB}),$$

   with R_i being the corresponding ith rule in the candidate rule set and HRB being the final hierarchical rule base.The initial pool is obtained with an individual having all genes with value '1' and the remaining individuals generated at random in {0, 1}, so that the initial HRB is taking into account in the genetic selection process.
2. *Chromosome evaluation:* The fitness function must be in accordance with the framework of imbalanced data-sets. Thus, we will use, as presented in Section 2.3, the geometric mean of the true rates, defined in (4) as:

   $$GM = \sqrt{\frac{TP}{TP + FN} \cdot \frac{TN}{FP + TN}}$$
3. *Crossover operator:* The half uniform crossover scheme (HUX) is employed. In this approach, the two parents are combined to produce two new offspring. The individual bits in the string are compared between the two parents and exactly half of the non-matching bits are swapped. Thus the Hamming distance (the number of differing bits) is first calculated. This number is divided by two. The resulting number is how many of the bits that do not match between the two parents will be swapped.
4. *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 in {1,0}. The restart procedure is applied when a threshold value is reached, which means that all the individuals coexisting in the population are very similar.
5. *Evolutionary model:* The CHC genetic model makes use of a "Population-based Selection" approach. N parents and their corresponding offspring are combined to select the best N individuals to take part of the next population. The CHC approach makes use of an incest prevention mechanism and a restarting process to provoke diversity in the population, instead of the well-known mutation operator.

   This incest prevention mechanism will be considered in order to apply the HUX operator, i.e., two parents are crossed if their hamming distance divided by 2 is higher than a predetermined threshold, L. The threshold value is initialized as: $L = (\#Genes/4.0)$. Following the original CHC scheme, L is decremented by one when the population does not change in one generation. The algorithm restarts when L is below zero. We will stop the genetic process if more than 3 restarts are performed without including any new chromosome in the population.

4. Experimental study

In order to develop the study, we use a five fold cross validation approach, that is, five partitions for training and test sets, 80% for training and 20% for test, where the five test data-sets form the whole set. For each data-set we consider the average results of the five partitions.

Statistical analysis needs to be carried out in order to find significant differences among the results obtained by the studied methods [20]. We consider the use of non-parametric tests, according to the recommendations made in [13], where it is presented a set of simple, safe and robust non-parametric tests for statistical comparisons of classifiers. For pair wise comparison we will use the Wilcoxon Signed-Ranks Test [37,44], and for multiple comparison we will employ different approaches, including the Friedman test [19], the Iman and Davenport test [26] and the Holm method [24]. We will use in all cases $\alpha = 0.05$ as level of confidence. A wider description of these tests is presented in the Appendix A.

In this section we will first introduce the configuration of the two-level learning method, determining all the parameters used in this experimental study. Next we will study the effect of preprocessing in the performance of the HFRBCS by contrasting the results obtained using the original data-sets against the ones obtained with the SMOTE algorithm. Then, we will analyze the results of the HFRBCS when applied to imbalanced data-sets globally, and considering two different degrees of imbalance. This last part of the study is divided into two sections: on the one hand, we will make a comparative study between our model and other fuzzy learning methodologies, including Chi et al.'s [9] and Ishibuchi et al.'s [29] rule learning algorithms, and a new approach proposed by Xu et al. for imbalanced data-sets, called E-Algorithm [45]. On the other hand, we will compare the performance of the HFRBCSs against the well-known C4.5 algorithm [35], that has been widely used for this kind of problems [4,15,32,38–40].

4.1. Experimental setup: parameters and data-sets

In our former studies [17,18] we selected as a good FRBCS model the use of the product T-norm as conjunction operator, together with the Penalized Certainty Factor [29] approach for the rule weight and FRM of the winning rule. This configuration will be employed for all the FRBCSs used in this work, including Chi et al.'s method, Ishibuchi et al.'s approach and E-Algorithm.

After several trials, we selected the following values for the parameters in the learning method for building HFRBCSs:

- Rule generation:
  - δ, $n(t+1)$-*linguistic partition* terms selector: 0.1.
  - α, used to decide the expansion of the rule: 0.2.

- GA Selection:
  - Number of evaluations: 10,000.
  - Population length: 61.

In the SMOTE preprocessing we consider only the 1-nearest neighbour to generate the synthetic samples, and we balance the training data to the 50% class distribution.

For Ishibuchi et al.'s rule generation method and E-Algorithm, only rules with three or less antecedent attributes are generated. Furthermore we have restricted the number of fuzzy rules in the RB to 30 for each class, using as selection measure the product of support and confidence. This configuration is the one indicated by the authors in [29,45].

In this paper we use the IR to distinguish between two-classes of imbalanced data-sets: data-sets with a *low imbalance* when the instances of the positive class are between 10 and 40% of the total instances (IR between 1.5 and 9) and data-sets with a *high imbalance* where there are no more than 10% of positive instances in the whole data-set compared to the negative ones (IR higher than 9). Specifically, we have considered 44 data-sets from UCI repository [2] with different IR. Table 3 summarizes the data employed in this study and shows, for each data-set, the number of examples (#Ex.), number of attributes (#Atts.), class name of each class (minority and majority), class attribute distribution and IR. This table is ordered by the IR, from low to highly imbalanced data-sets.

4.2. Analysis of the significance of the preprocessing approach

Our first aim is to show that preprocessing is a necessity in the framework of imbalanced data-sets. As mentioned in Section 2.2, the objective of the preprocessing step is to prepare the data for the experiments, removing the imbalance among classes by changing the original class distribution. In this manner, we can have all the data-sets prepared and stored in advance and thus, there is no need to adapt the algorithm itself to perform well with this type of data.

Table 4 shows the mean results for the Chi et al.'s method and the HFRBCS without preprocessing and with the SMOTE technique [7] in all the imbalanced data-sets. The difference in performance achieved in each case, is very clear only by observing this table. We also show statistically the goodness of preprocessing using a Wilcoxon test (Table 5), in which the p-value is 0 in all cases.

4.3. Analysis of the hierarchical fuzzy rule based classification system on imbalanced data-sets

In this part of the study we will focus on determining whether our HFRBCS is robust in the framework of imbalanced data-sets and if it improves the performance of other FRBCSs approaches and the well-known C4.5 algorithm. According to the conclusions of the previous section, the SMOTE preprocessing is applied for all approaches apart from the E-Algorithm, which is an algorithm proposed for imbalanced data-sets that uses cost values for instances.

Following this idea, Table 6 shows the results for the test partitions for each FRBCS method with its associated standard deviation. Specifically, by columns we include the Chi et al.'s method with 3 and 5 labels (Chi-3 and Chi-5), the Ishibuchi et al.'s method (Ishibuchi05), the E-Algorithm and the HFRBCS. Additionally, we include the results for the C4.5 decision tree. This table is divided by the IR, on the one hand data-sets with low imbalance and, on the other hand, data-sets with high imbalance. The best global result for test is stressed in **boldface** in each case. In Appendix B the reader can examine the whole training and test results.

Table 3
Summary description for imbalanced data-sets.

Data-set	#Ex.	#Atts.	Class (min., maj.)	% Class (min.; maj.)	IR
Data-sets with low imbalance (IR 1.5–9)					
Glass1	214	9	(build-win-non_float-proc; remainder)	(35.51, 64.49)	1.82
Ecoli0vs1	220	7	(im; cp)	(35.00, 65.00)	1.86
Wisconsin	683	9	(malignant; benign)	(35.00, 65.00)	1.86
Pima	768	8	(tested-positive; tested-negative)	(34.84, 66.16)	1.90
Iris0	150	4	(Iris-Setosa; remainder)	(33.33, 66.67)	2.00
Glass0	214	9	(build-win-float-proc; remainder)	(32.71, 67.29)	2.06
Yeast1	1484	8	(nuc; remainder)	(28.91, 71.09)	2.46
Vehicle1	846	18	(Saab; remainder)	(28.37, 71.63)	2.52
Vehicle2	846	18	(Bus; remainder)	(28.37, 71.63)	2.52
Vehicle3	846	18	(Opel; remainder)	(28.37, 71.63)	2.52
Haberman	306	3	(Die; Survive)	(27.42, 73.58)	2.68
Glass0123vs456	214	9	(non-window glass; remainder)	(23.83, 76.17)	3.19
Vehicle0	846	18	(Van; remainder)	(23.64, 76.36)	3.23
Ecoli1	336	7	(im; remainder)	(22.92, 77.08)	3.36
New-thyroid2	215	5	(hypo; remainder)	(16.89, 83.11)	4.92
New-thyroid1	215	5	(hyper; remainder)	(16.28, 83.72)	5.14
Ecoli2	336	7	(pp; remainder)	(15.48, 84.52)	5.46
Segment0	2308	19	(brickface; remainder)	(14.26, 85.74)	6.01
Glass6	214	9	(headlamps; remainder)	(13.55, 86.45)	6.38
Yeast3	1484	8	(me3; remainder)	(10.98, 89.02)	8.11
Ecoli3	336	7	(imU; remainder)	(10.88, 89.12)	8.19
Page-blocks0	5472	10	(remainder; text)	(10.23, 89.77)	8.77
Data-sets with high imbalance (IR higher than 9)					
Yeast2vs4	514	8	(cyt; me2)	(9.92, 90.08)	9.08
Yeast05679vs4	528	8	(me2; mit,me3,exc,vac,erl)	(9.66, 90.34)	9.35
Vowel0	988	13	(hid; remainder)	(9.01, 90.99)	10.10
Glass016vs2	192	9	(Ve-win-float-proc; build-win-float-proc, build-win-non_float-proc,headlamps)	(8.89, 91.11)	10.29
Glass2	214	9	(Ve-win-float-proc; remainder)	(8.78, 91.22)	10.39
Ecoli4	336	7	(om; remainder)	(6.74, 93.26)	13.84
Yeast1vs7	459	8	(nuc; vac)	(6.72, 93.28)	13.87
Shuttle0vs4	1829	9	(Rad Flow; Bypass)	(6.72, 93.28)	13.87
Glass4	214	9	(containers; remainder)	(6.07, 93.93)	15.47
Page-blocks13vs2	472	10	(graphic; horiz.line,picture)	(5.93, 94.07)	15.85
Abalone9vs18	731	8	(18; 9)	(5.65, 94.25)	16.68
Glass016vs5	184	9	(tableware; build-win-float-proc, build-win-non_float-proc,headlamps)	(4.89, 95.11)	19.44
Shuttle2vs4	129	9	(Fpv Open; Bypass)	(4.65, 95.35)	20.5
Yeast1458vs7	693	8	(vac; nuc,me2,me3,pox)	(4.33, 95.67)	22.10
Glass5	214	9	(tableware; remainder)	(4.20, 95.80)	22.81
Yeast2vs8	482	8	(pox; cyt)	(4.15, 95.85)	23.10
Yeast4	1484	8	(me2; remainder)	(3.43, 96.57)	28.41
Yeast1289vs7	947	8	(vac; nuc,cyt,pox,erl)	(3.17, 96.83)	30.56
Yeast5	1484	8	(me1; remainder)	(2.96, 97.04)	32.78
Ecoli0137vs26	281	7	(pp,imL; cp,im,imU,imS)	(2.49, 97.51)	39.15
Yeast6	1484	8	(exc; remainder)	(2.49, 97.51)	39.15
Abalone19	4174	8	(19; remainder)	(0.77, 99.23)	128.87

This study is divided into two parts. First, we will analyze the results globally for all imbalanced data-sets and then, we will study the two imbalance scenarios defined in this paper. Furthermore, our aim is to test the HFRBCS against the FRBCSs approaches and C4.5 separately.

4.3.1. Global analysis of the hierarchical fuzzy rule based classification system

First of all, we will study the performance of the HFRBCS with the remaining FRBCSs approaches. In order to compare the results, we will use a multiple comparison test to find the best approach in this case, considering the results in the test par-

Table 4
Average results for FRBCS in imbalanced data-sets with and without preprocessing.

Algorithm	No preprocessing		SMOTE preprocessing	
	GM_{Tr}	GM_{Tst}	GM_{Tr}	GM_{Tst}
Chi-3	50.64 ± 3.59	42.83 ± 9.47	84.57 ± 1.86	79.65 ± 7.71
Chi-5	73.70 ± 2.99	57.60 ± 11.33	90.17 ± 1.01	77.97 ± 8.77
HFRBCS	82.80 ± 2.30	66.02 ± 11.49	93.82 ± 1.05	81.57 ± 9.10

Table 5
Wilcoxon Test to compare the use of the SMOTE preprocessing against original data-sets. R^+ corresponds to no preprocessing and R^- to SMOTE.

Comparison	R^+	R^-	Hypothesis ($\alpha = 0.1$)	p-Value
Chi-3 vs. Chi-3 + SMOTE	21.5	977.5	Rej. for Chi3 + SMOTE	0.000
Chi-5 vs. Chi-5 + SMOTE	38.5	960.5	Rej. for Chi5 + SMOTE	0.000
HFRBCS vs. HFRBCS + SMOTE	36.5	953.5	Rej. for HFRBCS + SMOTE	0.000

Table 6
Detailed results table for FRBCSs in imbalanced data-sets. Only test results are shown.

Data-Set	Chi-3	Chi-5	Ishibuchi05	E-Algorithm	HFRBCS	C4.5
Data-sets with low imbalance						
Glass1	64.90 ± 6.91	64.91 ± 6.87	59.29 ± 10.33	0.00 ± 0.00	73.66 ± 4.66	75.11 ± 3.74
Ecoli0vs1	92.27 ± 5.93	95.56 ± 5.15	96.70 ± 2.40	95.25 ± 4.75	93.63 ± 6.45	97.95 ± 2.20
Wisconsin	88.91 ± 2.13	43.58 ± 5.86	95.78 ± 1.38	96.01 ± 1.55	88.24 ± 1.63	95.44 ± 2.01
Pima	66.80 ± 5.93	66.78 ± 2.28	71.10 ± 4.45	55.01 ± 4.64	68.72 ± 5.26	71.26 ± 4.05
Iris0	100.0 ± 0.00	98.97 ± 2.29	100.0 ± 0.00	100.0 ± 0.00	100.0 ± 0.00	98.97 ± 2.29
Glass0	64.06 ± 3.51	63.69 ± 1.80	69.39 ± 7.70	0.00 ± 0.00	76.57 ± 8.05	78.14 ± 2.21
Yeast1	67.69 ± 1.91	69.66 ± 1.52	51.41 ± 12.18	0.00 ± 0.00	71.71 ± 2.39	70.86 ± 2.95
Vehicle1	70.92 ± 4.34	71.88 ± 1.25	64.89 ± 4.37	3.09 ± 6.90	71.76 ± 2.64	69.28 ± 3.41
Vehicle2	85.54 ± 3.36	87.19 ± 3.04	67.82 ± 4.95	43.83 ± 13.17	90.61 ± 2.17	94.85 ± 1.68
Vehicle3	69.22 ± 4.89	63.13 ± 1.95	63.12 ± 4.06	0.00 ± 0.00	66.80 ± 3.34	74.34 ± 1.08
Haberman	58.91 ± 6.03	60.40 ± 2.40	62.65 ± 2.84	4.94 ± 11.06	57.08 ± 4.09	61.32 ± 3.85
Glass0123vs456	85.83 ± 3.04	85.94 ± 1.66	88.56 ± 5.18	82.09 ± 6.96	88.37 ± 3.97	90.13 ± 3.17
Vehicle0	86.41 ± 3.06	84.93 ± 1.61	75.94 ± 1.42	39.07 ± 16.49	88.92 ± 1.96	91.10 ± 2.70
Ecoli1	85.28 ± 9.77	86.05 ± 8.57	85.71 ± 2.86	77.81 ± 7.90	84.18 ± 12.69	76.10 ± 9.58
New-Thyroid2	89.81 ± 10.77	96.34 ± 6.65	94.21 ± 4.23	88.57 ± 3.82	99.72 ± 0.63	96.51 ± 4.87
New-Thyroid1	87.44 ± 8.11	95.38 ± 8.80	89.02 ± 13.52	88.52 ± 8.79	98.58 ± 2.48	97.98 ± 3.79
Ecoli2	88.01 ± 5.45	87.64 ± 4.96	87.00 ± 4.43	70.35 ± 15.36	87.62 ± 8.24	91.60 ± 4.86
Segment0	94.99 ± 0.45	95.88 ± 1.21	42.47 ± 2.79	95.33 ± 1.14	97.51 ± 1.11	99.26 ± 0.61
Glass6	83.87 ± 9.82	78.13 ± 7.78	86.27 ± 8.19	90.23 ± 3.77	86.95 ± 10.84	83.00 ± 9.05
Yeast3	90.13 ± 4.09	89.33 ± 3.30	77.06 ± 17.73	81.99 ± 2.28	90.41 ± 2.34	88.50 ± 3.66
Ecoli3	87.58 ± 4.08	91.61 ± 4.95	85.39 ± 3.70	78.54 ± 8.68	90.81 ± 4.43	88.77 ± 7.65
Page-blocks0	79.91 ± 4.29	87.25 ± 1.94	32.16 ± 9.61	64.51 ± 2.79	91.40 ± 0.67	94.84 ± 1.52
Mean	81.29 ± 4.90	80.19 ± 3.90	74.81 ± 5.83	57.05 ± 5.46	84.69 ± 4.09	**85.70 ± 3.68**
Data-sets with high imbalance						
Yeast2vs4	86.80 ± 5.53	86.39 ± 7.35	70.85 ± 23.45	80.92 ± 9.09	89.32 ± 4.18	85.09 ± 10.15
Yeast05679vs4	78.91 ± 5.99	75.99 ± 6.39	79.49 ± 9.54	59.99 ± 16.44	73.18 ± 7.47	74.88 ± 10.88
Vowel0	98.37 ± 0.61	97.87 ± 1.84	89.03 ± 6.63	89.63 ± 6.09	98.82 ± 1.62	94.74 ± 5.22
Glass016vs2	40.84 ± 7.62	56.17 ± 5.16	41.18 ± 15.40	0.00 ± 0.00	58.37 ± 20.04	48.91 ± 29.44
Glass2	47.67 ± 10.16	49.24 ± 8.19	43.55 ± 15.70	9.87 ± 22.07	54.84 ± 20.57	33.86 ± 32.29
Ecoli4	91.27 ± 7.43	92.11 ± 8.35	86.92 ± 8.65	92.43 ± 8.24	93.02 ± 8.17	81.28 ± 11.67
Yeast1vs7	80.05 ± 6.43	63.02 ± 12.62	53.15 ± 10.35	27.55 ± 26.06	70.74 ± 12.40	67.73 ± 2.28
Shuttle0vs4	99.12 ± 1.15	98.71 ± 1.18	99.16 ± 1.15	98.40 ± 1.26	99.12 ± 1.15	99.97 ± 0.07
Glass4	84.96 ± 13.80	81.75 ± 11.24	78.27 ± 17.70	83.38 ± 19.89	70.39 ± 40.49	83.71 ± 10.78
Page-Blocks13vs4	91.92 ± 4.76	92.93 ± 9.48	94.53 ± 4.88	94.12 ± 10.33	98.64 ± 0.65	99.55 ± 0.47
Abalone9-18	63.93 ± 11.00	66.47 ± 10.67	65.78 ± 9.23	32.29 ± 20.61	67.56 ± 14.01	53.19 ± 8.25
Glass016vs5	71.48 ± 40.17	75.59 ± 42.27	88.77 ± 2.48	65.14 ± 39.41	77.96 ± 43.61	72.08 ± 42.33
Shuttle2vs4	89.99 ± 8.61	78.34 ± 43.87	99.17 ± 1.13	100.0 ± 0.00	97.49 ± 2.71	99.15 ± 1.90
Yeast1458vs7	62.40 ± 4.55	58.76 ± 8.57	40.80 ± 16.58	0.00 ± 0.00	62.49 ± 6.26	41.19 ± 6.06
Glass5	81.56 ± 12.65	64.33 ± 38.40	89.96 ± 2.43	50.61 ± 47.17	68.73 ± 39.56	86.70 ± 15.44
Yeast2vs8	72.75 ± 14.99	78.76 ± 8.60	72.83 ± 14.97	72.83 ± 14.97	72.47 ± 15.10	78.23 ± 13.05
Yeast4	82.99 ± 3.10	83.07 ± 2.58	71.36 ± 23.29	32.16 ± 20.59	82.64 ± 2.29	65.00 ± 8.94
Yeast1289vs7	76.12 ± 7.24	69.26 ± 4.57	48.55 ± 16.86	50.00 ± 13.62	69.37 ± 4.37	64.13 ± 9.00
Yeast5	93.41 ± 5.35	93.64 ± 2.70	94.94 ± 0.38	88.17 ± 7.04	94.20 ± 2.59	92.04 ± 4.99
Ecoli0137vs26	71.04 ± 41.38	49.57 ± 46.41	71.31 ± 41.65	73.65 ± 43.09	71.48 ± 41.80	71.21 ± 41.31
Yeast6	87.50 ± 10.55	87.73 ± 9.32	88.42 ± 6.06	51.72 ± 13.76	84.92 ± 12.88	80.38 ± 15.47
Abalone19	62.96 ± 8.27	66.71 ± 10.21	66.09 ± 9.40	0.00 ± 0.00	70.19 ± 8.56	15.58 ± 21.36
Mean	78.00 ± 10.51	75.75 ± 13.63	74.28 ± 11.72	56.95 ± 15.44	**78.45 ± 14.11**	72.21 ± 13.70
All data-sets						
Mean	79.65 ± 7.71	77.97 ± 8.77	74.55 ± 8.78	57.00 ± 10.45	**81.57 ± 9.10**	78.95 ± 8.69

titions (GM_{Tst}). In Table 7, the results of applying Friedman and Iman-Davenport tests are shown in order to see if there are differences in the results. We employ the χ^2-distribution with 4 degrees of freedom and the F-distribution with 4 and 172 degrees of freedom for $N_{ds} = 44$. We emphasize in boldface the highest value between the two values that are being compared, and as the smallest in both cases corresponds to the value given by the statistic, it informs us of the rejection of the null hypothesis of equality of means, telling us of the existence of significant differences among the observed results in all data-sets. Table 8 shows the rankings (computed using a Friedman test) of the 5 algorithms considered.

Now, we apply a Holm test to compare the best ranking method (HFRBCS) with the remaining fuzzy methods. The result of this test is shown in Table 9, in which the algorithms are ordered with respect to the z value obtained. Thus, by using the normal distribution, we can obtain the corresponding p-value associated with each comparison and this can be compared with the associated α/i in the same row of the table to show whether the associated hypothesis of equal behaviour is rejected in favour of the best ranking algorithm or not.

Therefore, analyzing the results presented in Table 6 and the statistical study shown in Tables 8 and 9 we conclude that our model is a solid FRBCS approach to deal with imbalanced data-sets, as it has shown to be the best performing algorithm when comparing with the remaining fuzzy rule learning methods applied in this study.

Finally, we use a Wilcoxon test for the comparison with the C4.5 algorithm, which is shown in Table 10. We can observe that our proposal achieves a higher ranking, but this is not enough to reject the null hypothesis. We may conclude that both approaches have a similar performance when treating all imbalanced data-sets as a whole, without taking into account the IR.

4.3.2. Analysis of the hierarchical fuzzy rule based classification system according to the imbalance ratio

In the final part of our study, we will analyze the behaviour of our hierarchical approach in each imbalanced scenario. Table 11 shows, by columns, the geometric mean in training and test of the different algorithms considered, for the two types

Table 7
Results of the Friedman and Iman-Davenport tests for comparing performance of the FRBCS in all imbalanced data-sets.

Method	Test value	Distribution value	p-Value
Friedman	**37.29091**	9.4877	1.56929E−7
Iman-Davenport	**11.56023**	2.4242	2.45881E−8

Table 8
Rankings obtained through a Friedman test for FRBCSs in all imbalance data-sets.

Algorithm	Ranking
HFRBCS	2.09091
Chi-5	2.77273
Chi-3	3.0
Ishibuchi05	3.02273
E-Algorithm	4.11364

Table 9
Holm test table for FRBCSs in all imbalanced data-sets. HFRBCS is the control method.

i	Algorithm	z	p	α/i	Hypothesis
4	E-Algorithm	6.00038	1.96858E−9	0.0125	Rejected for HFRBCS
3	Ishibuchi05	2.76422	0.00576	0.01667	Rejected for HFRBCS
2	Chi-3	2.69680	0.00700	0.025	Rejected for HFRBCS
1	Chi-5	2.02260	0.04311	0.05	Rejected for HFRBCS

Table 10
Wilcoxon test to compare the HFRBCS against C4.5 in all imbalanced data-sets. R^+ corresponds to HFRBCS and R^- to C4.5.

Comparison	R^+	R^-	Hypothesis ($\alpha = 0.05$)	p-Value
HFRBCS vs. C4.5	589	401	Accepted	0.273

Table 11
Results table for FRBCSs for the different degrees of imbalance.

Algorithm	Low imbalance		High imbalance		All data-sets	
	GM_{Tr}	GM_{Tst}	GM_{Tr}	GM_{Tst}	GM_{Tr}	GM_{Tst}
Chi-3	85.50 ± 1.28	81.29 ± 4.90	83.64 ± 2.43	78.00 ± 10. 51	84.57 ± 1.86	79.65 ± 7.71
Chi-5	91.31 ± 0.69	80.19 ± 3.90	89.04 ± 1.32	75.75 ± 13. 63	90.17 ± 1.01	77.97 ± 8.77
Ishibuchi05	75.45 ± 3.04	74.81 ± 5.83	76.90 ± 6.35	74. 28 ± 11.72	76.17 ± 4.70	74.55 ± 8.78
E-Algorithm	58.33 ± 4.09	57.05 ± 5.46	65.72 ± 5.06	56. 95 ± 15.44	62.02 ± 4.57	57.00 ± 10.45
HFRBCS	94.30 ± 0.80	84.69 ± 4.09	93.35 ± 1.30	**78.45 ± 14.11**	93.82 ± 1.05	**81.57 ± 9.10**
C4.5	94.95 ± 0.87	**85.70 ± 3.68**	95.81 ± 1.77	72.21 ± 13.70	95.38 ± 1.32	78. 95 ± 8.69

of data-sets, that is, low and high imbalance (IR lower than 9 and higher than 9, respectively). The last column corresponds to the global results. Reader can refer to Table 6, presented in the previous part of this study, where we show the detailed results in each data-set.

The main conclusion extracted from this table is that our HFRBCS is very robust in both imbalanced scenarios considered, as it obtains very competitive results independently of the IR. Next, we will analyze the results in each case, for data-sets with low and high imbalance. We will employ multiple comparison tests for the statistical study, using for this purpose Friedman, Iman-Davenport and Holm tests. As we did in the previous section, we will compare the HFRBCS with the FRBCSs and with the C4.5 decision tree separately, using a Wilcoxon test for the study with C4.5.

- **Data-sets with low imbalance:** This study is shown through Tables 12–15. First, we check for statistical differences using Friedman and Iman-Davenport tests, following the same scheme as in the previous section. Since the smallest value corresponds in both cases to the one given by the statistic, we conclude that there are differences among the algorithms. Thus, Table 13 shows the ranking for the algorithms and Table 14 contains a Holm test, which shows that the HFRBCS is better in performance than the remaining FRBCS unless the Chi et al.'s method with 5 labels.
  Now, we will compare the performance achieved by our proposal with C4.5 in low imbalanced data-sets by means of a Wilcoxon test, which is shown in Table 15. Furthermore, we compare the HFRBCS with the Chi et al.'s approach with 5 labels in order to check if we find differences between both algorithms.
  The main conclusion after this study is that the HFRBCS is better than the rest of the FRBCS methods. It outperforms the base Chi LRG-method, the Ishibuchi et al.'s approach and the E-Algorithm. When compared with C4.5, there are no statistical differences in this imbalance scenario.
- **Data-sets with high imbalance:** This part of the study is very important, since it includes the data-sets with a higher degree of imbalance. In this manner, we can analyze how the imbalance actually affects the different methods employed in this study. For this purpose, we use the Friedman and Iman-Daverport tests in order to find statistical differences, as shown in Table 16. Next, Table 17 shows the ranking for the FRBCS algorithms, in which our HFRBCS proposal is the first one. Finally, we perform a Holm test, which is shown in Table 18, where we can only conclude that the HFRBCS is better than the E-Algorithm in data-sets with high imbalance.
  A Wilcoxon test (Table 19) will help us to make a pairwise comparison between our proposal and the remaining algorithms, including C4.5 in this case. Now, we detect differences between the HFRBCS and the Chi et al.'s method with 5

Table 12
Results of the Friedman and Iman-Davenport tests for comparing performance of the FRBCS in data-sets with low imbalance.

Method	Test value	Distribution value	p-Value
Friedman	**23.29091**	9.4877	1.10759E−4
Iman-Davenport	**7.55858**	2.4803	2.98974E−5

Table 13
Rankings obtained through a Friedman test for FRBCSs in data-sets with low imbalance.

Algorithm	Ranking
HFRBCS	1.97727
Chi-5	2.63636
Chi-3	3.06818
Ishibuchi05	3.11364
E-Algorithm	4.20454

Table 14
Holm test table for FRBCSs in data-sets with low imbalance. HFRBCS is the control method.

i	Algorithm	z	p	α/i	Hypothesis
4	E-Algorithm	4.67197	2.98329E−6	0.0125	Rejected for HFRBCS
3	Ishibuchi05	2.38366	0.01714	0.01667	Rejected for HFRBCS
2	Chi-3	2.28831	0.02212	0.025	Rejected for HFRBCS
1	Chi-5	1.38252	0.16681	0.05	Accepted

Table 15
Wilcoxon test to compare the HFRBCS against Chi-5 and C4.5 in data-set with low imbalance. R^+ corresponds to HFRBCS and R^- to Chi-5 and C4.5 in each case.

Comparison	R^+	R^-	Hypothesis ($\alpha = 0.05$)	p-Value
HFRBCS vs. Chi-5	219	34	Rejected for HFRBCS	0.003
HFRBCS vs. C4.5	84	169	Accepted	0.168

Table 16
Results of the Friedman and Iman-Davenport tests for comparing performance of the FRBCS in data-sets with high imbalance.

Method	Test value	Distribution value	*p*-Value
Friedman	**14.92727**	9.4877	0.00485
Iman-Davenport	**4.28987**	2.4803	0.00330

Table 17
Rankings obtained through a Friedman test for FRBCSs in data-sets with high imbalance.

Algorithm	Ranking
HFRBCS	2.20454
Chi-5	2.90909
Chi-3	2.93182
Ishibuchi05	2.93182
E-Algorithm	4.02273

Table 18
Holm test table for FRBCSs in data-sets with high imbalance. HFRBCS is the control method.

i	Algorithm	*z*	*p*	α/i	Hypothesis
4	E-Algorithm	3.81385	1.36818E−4	0.0125	Rejected for HFRBCS
3	Ishibuchi05	1.52554	0.12712	0.01667	Accepted
2	Chi-3	1.52554	0.12712	0.025	Accepted
1	Chi-5	1.47787	0.13944	0.05	Accepted

Table 19
Wilcoxon test to compare the HFRBCS against the remaining FRBCS approaches and C4.5 in data-set with high imbalance. R^+ corresponds to HFRBCS and R^- to the remaining algorithms in each case.

Comparison	R^+	R^-	Hypothesis ($\alpha = 0.05$)	*p*-Value
HFRBCS vs. Chi-3	148.5	104.5	Accepted	0.498
HFRBCS vs. Chi-5	191	62	Rejected for HFRBCS	0.036
HFRBCS vs. Ishibuchi05	175	78	Accepted	0.115
HFRBCS vs. C4.5	192	61	Rejected for HFRBCS	0.033

labels per variable, but it remains statistically similar to the Ishibuchi et al.'s algorithm and the Chi et al.'s method with 3 labels. Nevertheless, watching the results for the comparison with C4.5 we see that the null hypothesis is rejected in favour of our HFRBCS proposal.

According to these results, we must emphasize the good behaviour achieved in highly imbalanced data-sets by the all fuzzy models studied here, particularly for our proposal. Furthermore, we can determine that it is very competitive, since it outperforms C4.5 algorithm in this kind of data-sets, with a *p*-value of 0.033.

In brief, we have improved the behaviour of the base FRBCS by a simple and effective methodology, that is, applying a higher granularity in the areas where the RB has a bad performance in order to obtain a better coverage of that area of the space of solutions. As future work we consider the inclusion of a multi-objective GA for rule selection with the aim of getting a trade-off between interpretability and accuracy [28,34].

5. Concluding remarks

In this paper, we have proposed an HFRBCS approach for classification with imbalanced data-sets. Our aim was to employ a hierarchical model to obtain a good balance among different granularity levels. A fine granularity is applied in the boundary areas, and a thick granularity may be applied in the rest of the classification space providing a good generalization. Thus, this approach enhances the classification performance in the overlapping areas between the minority and majority classes.

Furthermore, we have made use of the SMOTE algorithm in order to balance the training data before the rule learning generation phase. This preprocessing step enables the obtention of better fuzzy rules than using the original data-sets and therefore, we improve the global performance of the fuzzy model.

In the experimental study, we have shown statistically that our proposal performs better than well-known FRBCSs approaches and that clearly outperforms the C4.5 decision tree, generally for all data-sets and particularly in data-sets with high imbalance.

Appendix A. On the use of non-parametric tests based on rankings

In this paper, we have made use of statistical techniques for the analysis of GBML methods, since they are a necessity in order to provide a correct empirical study [13,20]. Specifically, we have employed non-parametric tests, due to the fact that the initial conditions that guarantee the reliability of the parametric tests may not be satisfied, making the statistical analysis to lose credibility [13].

In this appendix, we describe the procedures for performing pairwise an multiple comparisons. Specifically, we have employed the Wilcoxon signed-rank test as non-parametric statistical procedure for performing pairwise comparisons between two algorithms. For multiple comparison we have used the Friedman and Iman-Davenport tests to detect statistical differences and the Holm post-hoc test in order to find what algorithms partners' average results are dissimilar. Next, we will describe both approaches.

A.1. Pairwise comparisons: Wilcoxon signed-ranks test

This is the analogous of the paired *t*-test in non-parametrical statistical procedures; therefore, it is a pair wise test that aims to detect significant differences between the behaviour of two algorithms.

Let d_i be the difference between the performance scores of the two-classifiers on *i*th out of N_{ds} data-sets. The differences are ranked according to their absolute values; average ranks are assigned in case of ties. Let R^+ be the sum of ranks for the data-sets on which the second algorithm outperformed the first, and R^- the sum of ranks for the opposite. Ranks of $d_i = 0$ are split evenly among the sums; if there is an odd number of them, one is ignored:

$$R^+ = \sum_{d_i>0} rank(d_i) + \frac{1}{2}\sum_{d_i=0} rank(d_i), \tag{20}$$

$$R^- = \sum_{d_i<0} rank(d_i) + \frac{1}{2}\sum_{d_i=0} rank(d_i). \tag{21}$$

Let T be the smallest of the sums, $T = min(R^+, R^-)$. If T is less than or equal to the value of the distribution of Wilcoxon for N_{ds} degrees of freedom (Table B.12 in [47]), the null hypothesis of equality of means is rejected.

A.2. Multiple comparisons: Friedman test and Holm post-hoc test

In order to perform a multiple comparison, it is necessary to check whether all the results obtained by the algorithms present any inequality. In the case of finding it, then we can know, by using a post-hoc test, what algorithms partners' average results are dissimilar. Next, we describe the non-parametric tests used.

- The first one is the Friedman test [37], which is a non-parametric equivalent of the test of repeated-measures ANOVA. It computes the ranking of the observed results for algorithm (r_j for the algorithm j with k algorithms) for each data-set, assigning to the best of them the ranking 1, and to the worst the ranking k. Under the null hypothesis, formed from supposing the results of the algorithms are equivalents and, therefore, their rankings are also similar, Friedman's statistic

$$\chi^2_F = \frac{12N_{ds}}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right], \tag{22}$$

is distributed according to χ^2_F with $k-1$ degrees of freedom, being $R_j = \frac{1}{N_{ds}}\sum_i r_i^j$, and N_{ds} the number of data-sets. The critical values for Friedman's statistic coincide with the established in the χ^2 distribution when $N_{ds} > 10$ and $k > 5$. In a contrary case, the exact values can be seen in [37,47].

- The second one of them is the Iman and Davenport test [26], which is a non-parametric test, derived from the Friedman test, less conservative than the Friedman statistic:

$$F_F = \frac{(N_{ds}-1)\chi^2_F}{N_{ds}(K-1)-\chi^2_F}$$

which is distributed according to the F-distribution with $k-1$ and $(k-1)(N_{ds}-1)$ degrees of freedom. Statistical tables for critical values can be found at [37,47].

- Holm test [24]: it is a multiple comparison procedure that can work with a control algorithm and compares it with the remaining methods. The test statistics for comparing the *i*th and *j*th method using this procedure is:

$$z = (R_i - R_j)\Big/\sqrt{\frac{k(k+1)}{6N_{ds}}}.$$

The z value is used to find the corresponding probability from the table of normal distribution, which is then compared with an appropriate level of confidence α. A Holm test is a step-up procedure that sequentially tests the hypotheses ordered by their significance. We will denote the ordered p-values by $p_1, p_2, \ldots$, so that $p_1 \leqslant p_2 \leqslant \cdots \leqslant p_{k-1}$. The Holm test compares each p_i with $\alpha/(k-i)$, starting from the most significant p value. If p_1 is below $\alpha/(k-1)$, the corresponding hypothesis is rejected and we allow to compare p_2 with $\alpha/(k-2)$. If the second hypothesis is rejected, the test proceeds with the third, and so on. As soon as a certain null hypothesis cannot be rejected, all the remain hypotheses are retained as well.

Appendix B. Detailed results for the experiments

See Table 20.

Table 20
Detailed results table for FRBCSs and C4.5 in imbalanced data-sets.

Data-set	Chi-3		Chi-5		Ishibuchi05		E-Algorithm		HFRBCS		C4.5	
	GM_{Tr}	GM_{Tst}	GM_{Tr}	GM_{Tst}	GM_{Tr}	GM_{Tst}	GM_{Tr}	GM_{Tst}	GM_{Tr}	GM_{Tst}	GM_{Tr}	GM_{Tst}
Data-sets with low imbalance												
Glass1	75.37	64.90	77.30	64.91	65.33	59.29	10.24	0.00	87.76	73.66	89.74	75.11
Ecoli0vs1	95.49	92.27	98.19	95.56	97.00	96.70	95.16	95.25	98.26	93.63	99.26	97.95
Wisconsin	98.07	88.91	99.72	43.58	96.17	95.78	96.04	96.01	99.92	88.24	98.31	95.44
Pima	72.31	66.80	85.53	66.78	71.31	71.10	55.86	55.01	90.97	68.72	83.88	71.26
Iris0	100.00	100.0	100.0	98.97	100.0	100.0	100.0	100.0	100.0	100.0	100.0	98.97
Glass0	66.57	64.06	74.44	63.69	72.22	69.39	0.00	0.00	86.94	76.57	94.23	78.14
Yeast1	68.33	67.69	72.75	69.66	51.83	51.41	0.00	0.00	78.22	71.71	80.34	70.86
Vehicle1	76.47	70.92	91.18	71.88	64.83	64.89	5.93	3.09	93.78	71.76	95.50	69.28
Vehicle2	88.10	85.54	96.36	87.19	66.28	67.82	46.24	43.83	98.82	90.61	98.95	94.85
Vehicle3	75.52	69.22	90.22	63.13	63.21	63.12	0.00	0.00	93.65	66.80	94.88	74.34
Haberman	66.21	58.91	70.86	60.40	64.36	62.65	8.47	4.94	76.53	57.08	74.00	61.32
Glass0123vs456	94.05	85.83	98.48	85.94	85.68	88.56	82.08	82.09	99.22	88.37	99.07	90.13
Vehicle0	88.23	86.41	96.26	84.93	76.54	75.94	44.68	39.07	98.28	88.92	98.97	91.10
Ecoli1	87.92	85.28	93.78	86.05	85.45	85.71	75.34	77.81	96.02	84.18	96.28	76.10
New-Thyroid2	94.70	89.81	99.58	96.34	94.34	94.21	88.94	88.57	99.79	99.72	99.57	96.51
New-Thyroid1	92.32	87.44	99.58	95.38	90.97	89.02	88.92	88.52	99.30	98.58	99.21	97.98
Ecoli2	89.66	88.01	92.90	87.64	87.23	87.00	71.98	70.35	94.93	87.62	95.11	91.60
Segment0	95.45	94.99	98.19	95.88	42.61	42.47	95.64	95.33	99.32	97.51	99.85	99.26
Glass6	95.04	83.87	98.06	78.13	86.42	86.27	90.84	90.23	98.61	86.95	99.59	83.00
Yeast3	91.37	90.13	92.01	89.33	79.97	77.06	82.09	81.99	95.22	90.41	95.64	88.50
Ecoli3	89.24	87.58	94.75	91.61	85.78	85.39	80.21	78.54	96.34	90.81	98.14	88.77
Page-Blocks0	80.60	79.91	88.64	87.25	32.41	32.16	64.65	64.51	92.72	91.40	98.46	94.84
Mean	85.50	81.29	91.31	80.19	75.45	74.81	58.33	57.05	94.30	84.69	94.95	**85.70**
Standard deviation	1.28	4.90	0.69	3.90	3.04	5.83	4.09	5.46	0.80	4.09	0.87	3.68
Data-sets with high imbalance												
Abalone9-18	69.80	63.93	71.07	66.47	66.42	65.78	39.67	32.29	83.96	67.56	95.20	53.19
Abalone19	70.39	62.96	75.99	66.71	66.93	66.09	0.00	0.00	83.43	70.19	84.31	15.58
Ecoli4	94.04	91.27	98.12	92.11	89.21	86.92	92.80	92.43	98.69	93.02	97.67	81.28
Glass2	58.00	47.67	71.39	49.24	45.25	43.55	27.03	9.87	82.99	54.84	95.68	33.86
Yeast4	83.44	82.99	87.94	83.07	75.80	71.36	38.31	32.16	90.01	82.64	90.76	65.00
Vowel0	98.56	98.37	99.64	97.87	89.99	89.03	89.84	89.63	99.99	98.82	99.67	94.74
Yeast2vs8	75.66	72.75	82.35	78.76	74.01	72.83	74.01	72.83	83.34	72.47	90.93	78.23
Glass4	95.15	84.96	98.87	81.75	87.03	78.27	84.82	83.38	99.81	70.39	98.42	83.71
Glass5	94.15	81.56	98.77	64.33	89.88	89.96	80.60	50.61	97.64	68.73	99.76	86.70
Yeast5	94.67	93.41	95.40	93.64	94.93	94.94	88.66	88.17	97.82	94.20	97.75	92.04
Yeast6	88.43	87.50	89.57	87.73	88.48	88.42	53.82	51.72	93.41	84.92	92.15	80.38
Ecoli0137vs26	93.85	71.04	96.79	49.57	85.21	71.31	83.99	73.65	98.67	71.48	96.70	71.21
Shuttle0vs4	100.0	99.12	100.0	98.71	99.18	99.16	98.42	98.40	100.0	99.12	99.99	99.97
Yeast1vs7	81.67	80.05	83.99	63.02	62.17	53.15	57.25	27.55	91.63	70.74	93.42	67.73
Shuttle2vs4	94.77	89.99	100.0	78.34	92.89	99.17	100.0	100.0	99.90	97.49	99.90	99.15
Glass016vs2	50.29	40.84	73.06	56.17	44.02	41.18	37.77	0.00	87.26	58.37	97.10	48.91
Glass016vs5	89.98	71.48	98.42	75.59	88.72	88.77	81.82	65.14	99.71	77.96	99.21	72.08
Page-Blocks13vs4	93.59	91.92	98.70	92.93	96.88	94.53	94.53	94.12	99.89	98.64	99.75	99.55
Yeast05679vs4	82.63	78.91	87.85	75.99	80.20	79.49	63.28	59.99	92.90	73.18	95.20	74.88
Yeast1289vs7	73.80	76.12	79.92	69.26	52.23	48.55	51.41	50.00	86.99	69.37	94.63	64.13
Yeast1458vs7	67.90	62.40	80.60	58.76	46.47	40.80	23.77	0.00	90.37	62.49	91.46	41.19
Yeast2vs4	89.36	86.80	90.40	86.39	75.82	70.85	83.98	80.92	95.27	89.32	98.13	85.09
Mean	83.64	78.00	89.04	75.75	76.90	74.28	65.72	56.95	93.35	**78.45**	95.81	72.21
Standard deviation	2.43	10.51	1.32	13.63	6.35	11.72	5.06	15.44	1.30	14.11	1.77	13.70
All data-sets												
Mean	84.57	79.65	90.17	77.97	76.17	74.55	62.02	57.00	93.82	**81.57**	95.38	78.95
Standard deviation	1.86	7.71	1.01	8.77	4.70	8.78	4.57	10.45	1.05	9.10	1.32	8.69

References

[1] R. Alcalá, J. Alcalá-Fdez, F. Herrera, J. Otero, Genetic learning of accurate and compact fuzzy rule based systems based on the 2-tuples linguistic representation, International Journal of Approximate Reasoning 44 (2007) 4564.
[2] A. Asuncion, D. Newman, 2007. UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences. URL: <http://www.ics.uci.edu/~mlearn/MLRepository.html>.
[3] R. Barandela, J.S. Sánchez, V. García, E. Rangel, Strategies for learning in class imbalance problems, Pattern Recognition 36 (3) (2003) 849–851.
[4] G.E.A.P.A. Batista, R.C. Prati, M.C. Monard, A study of the behaviour of several methods for balancing machine learning training data, SIGKDD Explorations 6 (1) (2004) 20–29.
[5] P. Campadelli, E. Casiraghi, G. Valentini, Support vector machines for candidate nodules classification, Letters on Neurocomputing 68 (2005) 281–288.
[6] J.R. Cano, F. Herrera, M. Lozano, Using evolutionary algorithms as instance selection for data reduction in kdd: an experimental study, IEEE Transactions on Evolutionary Computation 7 (6) (2003) 561–575.
[7] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, Smote: synthetic minority over-sampling technique, Journal of Artificial Intelligent Research 16 (2002) 321–357.
[8] N.V. Chawla, N. Japkowicz, A. Kolcz, Editorial: special issue on learning from imbalanced data-sets, SIGKDD Explorations 6 (1) (2004) 1–6.
[9] Z. Chi, H. Yan, T. Pham, Fuzzy algorithms with applications to image processing and pattern recognition, World Scientific, 1996.
[10] J.-N. Choi, S.-K. Oh, W. Pedrycz, Structural and parametric design of fuzzy inference systems using hierarchical fair competition-based parallel genetic algorithms and information granulation, International Journal of Approximate Reasoning 49 (3) (2008) 631–648.
[11] O. Cordón, M.J. del Jesus, F. Herrera, A proposal on reasoning methods in fuzzy rule-based classification systems, International Journal of Approximate Reasoning 20 (1) (1999) 21–45.
[12] O. Cordón, F. Herrera, I. Zwir, Linguistic modeling by hierarchical systems of linguistic rules, IEEE Transactions on Fuzzy Systems 10 (1) (2002) 2–20.
[13] J. Demšar, Statistical comparisons of classifiers over multiple data-sets, Journal of Machine Learning Research 7 (2006) 1–30.
[14] L.J. Eshelman, 1991. Foundations of Genetic Algorithms. Morgan Kaufman, Ch. The CHC Adaptive Search Algorithm: How to have Safe Search when Engaging in Nontraditional Genetic Recombination, pp. 265–283.
[15] A. Estabrooks, T. Jo, N. Japkowicz, A multiple resampling method for learning from imbalanced data-sets, Computational Intelligence 20 (1) (2004) 18–36.
[16] T. Fawcett, F.J. Provost, Adaptive fraud detection, Data Mining and Knowledge Discovery 1 (3) (1997) 291–316.
[17] A. Fernández, S. García, M.J. del Jesus, F. Herrera, An analysis of the rule weights and fuzzy reasoning methods for linguistic rule based classification systems applied to problems with highly imbalanced data-sets, in: International Workshop on Fuzzy Logic and Applications (WILF07), Lecture Notes on Computer Science, vol. 4578, Springer-Verlag, 2007, pp. 170–179.
[18] A. Fernández, S. García, M.J. del Jesus, F. Herrera, A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets, Fuzzy Sets and Systems 159 (18) (2008) 2378–2398.
[19] M. Friedman, The use of ranks to avoid the assumption of normality implicit in the analysis of variance, Journal of the American Statistical Association 32 (1937) 675–701.
[20] S. García, D. Molina, M. Lozano, 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. Journal of Heuristics, in press, doi: 10.1007/s10732-008-9080-4.
[21] J.W. Grzymala-Busse, L.K. Goodwin, X. Zhang, Increasing sensitivity of preterm birth by changing rule strengths, Pattern Recognition Letters 24 (6) (2003) 903–910.
[22] J.W. Grzymala-Busse, J. Stefanowski, S. Wilk, A comparison of two approaches to data mining from imbalanced data, Journal of Intelligent Manufacturing 16 (6) (2005) 565–573.
[23] F. Herrera, Genetic fuzzy systems: taxonomy, current research trends and prospects, Evolutionary Intelligence 1 (2008) 27–46.
[24] S. Holm, A simple sequentially rejective multiple test procedure, Scandinavian Journal of Statistics 6 (1979) 65–70.
[25] Y.M. Huang, C.M. Hung, H.C. Jiau, Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem, Nonlinear Analysis: Real World Applications 7 (4) (2006) 720–747.
[26] R.L. Iman, J.M. Davenport, Approximations of the critical region of the friedman statistic, Communications in Statistics, Part A – Theory Methods 9 (1980) 571–595.
[27] H. Ishibuchi, T. Nakashima, Effect of rule weights in fuzzy rule-based classification systems, IEEE Transactions on Fuzzy Systems 9 (4) (2001) 506–515.
[28] H. Ishibuchi, Y. Nojima, Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning, International Journal of Approximate Reasoning 44 (2007) 4–31.
[29] H. Ishibuchi, T. Yamamoto, Rule weight specification in fuzzy rule-based classification systems, IEEE Transactions on Fuzzy Systems 13 (2005) 428–435.
[30] N. Japkowicz, S. Stephen, The class imbalance problem: a systematic study, Intelligent Data Analysis 6 (5) (2002) 429–450.
[31] M. Kubat, R.C. Holte, S. Matwin, Machine learning for the detection of oil spills in satellite radar images, Machine Learning 30 (2–3) (1998) 195–215.
[32] A. Orriols-Puig, E. Bernadó-Mansilla, Evolutionary rule-based systems for imbalanced data-sets, Soft Computing 13 (3) (2009) 213–225.
[33] C. Phua, D. Alahakoon, V. Lee, Minority report in fraud detection: classification of skewed data, SIGKDD Explorations Newsletter 6 (1) (2004) 50–59.
[34] P. Pulkkinen, H. Koivisto, Fuzzy classifier identification using decision tree and multiobjective evolutionary algorithms, International Journal of Approximate Reasoning 48 (2) (2008) 526–543.
[35] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA, 1993.
[36] L. Sánchez, M.R. Suárez, J.R. Villar, I. Couso, Mutual information-based feature selection and partition design in fuzzy rule-based classifiers from vague data, International Journal of Approximate Reasoning 49 (3) (2008) 607–622.
[37] D. Sheskin, Handbook of parametric and nonparametric statistical procedures, second ed., Chapman and Hall/CRC, 2006.
[38] C.-T. Su, L.-S. Chen, Y. Yih, Knowledge acquisition through information granulation for imbalanced data, Expert Systems with Applications 31 (2006) 531–541.
[39] C.-T. Su, Y.-H. Hsiao, An evaluation of the robustness of MTS for imbalanced data, IEEE Transactions on Knowledge and Data Engineering 19 (10) (2007) 1321–1332.
[40] Y. Sun, M.S. Kamel, A.K. Wong, Y. Wang, Cost-sensitive boosting for classification of imbalanced data, Pattern Recognition 40 (2007) 3358–3378.
[41] S. Tan, Neighbor-weighted k-nearest neighbor for unbalanced text corpus, Expert Systems with Applications 28 (4) (2005) 667–671.
[42] L.X. Wang, J.M. Mendel, Generating fuzzy rules by learning from examples, IEEE Transactions on Systems, Man, and Cybernetics 25 (2) (1992) 353–361.
[43] G.M. Weiss, Mining with rarity: a unifying framework, SIGKDD Explorations 6 (1) (2004) 7–19.
[44] F. Wilcoxon, Individual comparisons by ranking methods, Biometrics 1 (1945) 80–83.
[45] L. Xu, M.Y. Chow, L.S. Taylor, Power distribution fault cause identification with imbalanced data using the data mining-based fuzzy classification e-algorithm, IEEE Transactions on Power Systems 22 (1) (2007) 164–171.
[46] Q. Yang, X. Wu, 10 challenging problems in data mining research, International Journal of Information Technology and Decision Making 5 (4) (2006) 597–604.
[47] J.H. Zar, Biostatistical Analysis, Prentice Hall, Upper Saddle River, NJ, 1999.