Noname manuscript No.
(will be inserted by the editor)

Multi-objective Evolutionary Design of Granular Rule-based Classifiers

**Michela Antonelli, Pietro Ducange,
Beatrice Lazzerini, Francesco Marcelloni**

Received: date / Accepted: date

Abstract In the last years, rule-based systems have been widely employed in several different application domains. The performance of these systems is strongly affected by the process of information granulation, which defines in terms of specific information granules such as sets, fuzzy sets and rough sets, the labels used in the rules. Generally, information granules are either provided by an expert, when possible, or extracted from the available data. In the framework of rule-based classifiers, we investigate the importance of determining an effective information granulation from data, preserving the comprehensibility of the granules. We show how the accuracies of rule-based classifiers can be increased by learning number and parameters of the granules, which partition the involved variables. To perform this analysis, we exploit a multi-objective evolutionary approach to the classifier generation we have recently proposed. We discuss different levels of information granulation optimization employing both the learning of the number of granules per variable and the tuning of each granule during the evolutionary process. We show and discuss the results obtained on several classification benchmark datasets by using fuzzy sets and intervals as types of information granules.

M. Antonelli
Translational Imaging Group, Centre for Medical Image Computing (CMIC), University College London
Wolfson House, Stephenson Way London, NW1 2HE, UK
E-mail: m.antonelli@ucl.ac.uk

P. Ducange
Faculty of Engineering, eCampus University,
Via Isimbardi, 10, 22060 Novedrate, Italy
E-mail: pietro.ducange@uniecampus.it

B. Lazzerini, F. Marcelloni
Department of Information Engineering, University of Pisa,
Largo Lucio Lazzarino 1, 56122 Pisa, Italy
E-mail: {beatrice.lazzerini, francesco.marcelloni}@unipi.it

Keywords Granular Rule-based Classifiers · Multi-objective Evolutionary Optimization · Fuzzy Sets · Intervals

1 Introduction

In all activities involving knowledge representation, reasoning and decision-making, people typically express themselves by resorting to some generic and conceptually meaningful entities, which are called information granules [1, 2]. The meaning and purpose of information granulation depend on the specific application domain. For example, granulation may simply refer to variable quantization. On the other hand, granules may correspond to data clusters, or to modules of a software design, etc. Granules may, in turn, consist of finer granules based, e.g., on similarity and functionality criteria.

Various formalisms and processing platforms for information granulation exist, including sets (in particular, intervals [3]), rough sets [4], fuzzy sets [5], shadowed sets [6], etc. The choice of a suitable formalism is basically problem-dependent. Further, granules can be directly specified by a human expert or derived automatically from data.

The term Granular Computing (GC) is often used to refer to a common conceptual and algorithmic platform for granular information processing. GC can be seen as a general framework embracing all methodologies and techniques that make use of information granules in problem solving [7, 8].

In this paper, we will consider granular rule-based classifiers (GRBCs), i.e., rule-based systems aimed at performing classification and consisting of rules whose antecedent part includes information granules. In particular, we are interested in granules that form a partition of the universe of the variables involved in the rules. To this aim, we take into account two formalisms for representing information granules: sets (in particular, numeric intervals [3]) and fuzzy sets [1]. While a set (interval) realizes an information granule by allowing each element of the universe of discourse either to belong or not to belong to that granule, fuzzy sets generalize this notion by allowing any number in the real unit interval to represent the membership degree of an element to the information granule.

A granular rule-based system basically includes a rule base (RB), a database (DB) containing the definition of the granules used in the RB, and an inference engine. RB and DB comprise the knowledge base of the rule-based system. Of course, the input-output mapping performed by the granular rule-based system relies on the specific formal frameworks in which the various types of information granules are defined and processed.

The rules can be generated either by encoding an expert's knowledge or automatically from data, typically exploiting a set of training samples consisting of input-target pairs. Once you choose the granulation type, the automatic generation of rules should be guided by a suitable trade-off between accuracy and rule interpretability so as to avoid, e.g., a too high number of rules and hardly comprehensible partitions of the involved variables. To this aim, multi-

objective evolutionary algorithms can be profitably exploited. In particular, when fuzzy sets are used as information granules, the systems resulting from multi-objective evolutionary optimization are typically referred to as multi-objective evolutionary fuzzy systems (MOEFSs) in the literature [9, 10].

Several papers, mainly related to MOEFSs, adopt multi-objective evolutionary algorithms for rule selection [11] (e.g., from an initial RB heuristically generated) or rule learning [12], DB tuning [13], rule learning/selection together with DB learning (in particular, partition granularity and membership function parameters)[14, 15, 16].

In this paper, within the framework of GRBCs, we will show how granulation tuning (i.e., tuning of the partitions) and granulation learning (i.e., learning of the most suitable number of information granules), used either separately or jointly, can sensibly influence performance.

More precisely, we will start from uniform partitions of the involved variables. Each of these partitions consists of equally sized information granules, whose number is chosen based on heuristic considerations. In particular, in case of intervals and fuzzy sets, we adopt, respectively, partitions consisting of consecutive, non-intersecting intervals, and overlapping triangular fuzzy sets.

In this context we will exploit rule learning from data. We will use the performance obtained by the developed rule-based systems on classification benchmark datasets as a quantity of reference against which we will assess the improved results achieved by equipping the rule-based systems with granulation tuning and/or granulation learning.

To this aim, on the one hand, starting from fixed uniform initial partitions of the variables, we will perform, besides rule learning, partition tuning. The meaning of partition tuning depends, of course, on the specific granulation tool adopted. E.g., in the case of intervals, partition tuning consists in suitably moving the endpoints, whereas, in the case of fuzzy sets, partition tuning concerns the determination of the position of fuzzy sets by identifying the values of the parameters of the corresponding membership functions.

On the other hand, when dealing with uniform partitions, we may be interested in determining the suitable number of elements of each partition: thus, we will learn, besides the rules, the number of elements of the partitions (without performing any tuning). Finally, we will perform concurrently rule learning, learning of the number of elements making the partitions, and tuning of the partitions.

We will show that the introduction of tuning and learning of information granulation helps to improve performance in all the considered cases.

From an operation point of view we will approach the generation of the GRBCs, including the number and the position of the granules, from data through a multi-objective evolutionary process, considering accuracy and interpretability as the objectives to be optimized. At the end of the optimization process, the decision maker will just have to choose the system representing the best trade-off between the considered objectives for the particular application.

Finally, we present and discuss the results obtained by applying the generated GRBCs to twenty-four well-known classification benchmark datasets.

In particular, we show how granulation tuning and learning affect the accuracy and the interpretability of the solutions generated by the multi-objective evolutionary optimization process.

The paper is organized as follows. Section 2 introduces the GRBCs and discusses their interpretability. Section 3 describes the main features of multi-objective evolutionary granular rule-based classifiers, which are simply called multi-objective granular classifiers (MOGCs) from now on, such as chromosome coding, mating operators, objective functions and multi-objective evolutionary algorithm. In Section 4, we discuss experimental results obtained by applying MOGCs to classification problems. Finally, Section 5 draws conclusions.

2 Rule-Based Classifiers

Let $X = \{X_1, \ldots, X_F\}$ be the set of input variables and X_{F+1} be the output variable. Let U_f, with $f = 1, ..., F$, be the universe of the f^{th} input variable X_f. Let $P_f = \{A_{f,1}, \ldots, A_{f,T_f}\}$ be a partition of variable X_f consisting of T_f information granules. In classification problems, the output variable X_{F+1} is a categorical variable assuming values in the set Γ of K possible classes $\Gamma = \{C_1, \ldots, C_K\}$. Let $\{(\mathbf{x}_1, x_{F+1,1}), \ldots, (\mathbf{x}_N, x_{F+1,N})\}$ be a training set composed of N input-output pairs, with $\mathbf{x}_t = [x_{t,1} \ldots, x_{t,F}] \in \Re^F$, $t = 1, \ldots, N$ and $x_{F+1,t} \in \Gamma$.

With the aim of determining the class of a given input vector, we adopt an RB composed of M rules expressed as:

$$\begin{aligned} R_m : \textbf{IF } X_1 \textbf{ is } A_{1,j_{m,1}} \textbf{ AND} \ldots \textbf{AND } X_f \textbf{ is } A_{f,j_{m,f}} \textbf{ AND} \ldots \\ \ldots \textbf{AND } X_F \textbf{ is } A_{F,j_{m,F}} \textbf{ THEN } X_{F+1} \textbf{ is } C_{j_m} with\ RW_m \end{aligned} \quad (1)$$

where C_{j_m} is the class label associated with the m^{th} rule, and RW_m is the rule weight, i.e., a certainty degree of the classification in the class C_{j_m} for a pattern belonging to the subspace delimited by the antecedent of the rule R_m.

In this paper, we consider only sets (intervals) and fuzzy sets as information granule types. A set A defined on a universe of discourse U is typically described by a characteristic function $A(x) : U \rightarrow \{0, 1\}$: the value 1 (respectively, 0) means that the element belongs (does not belong) to the information granule represented by the set. The characteristic function is also used to define the three fundamental set operations, namely, union, intersection and complement. A particular type of sets are intervals, for which both set-theoretic and algebraic operations are defined [17, 3].

The classical notion of set (or crisp set) can be extended by introducing fuzzy sets. A fuzzy set A defined on a universe of discourse U is characterized by a membership function $A(x) : U \rightarrow [0, 1]$ which associates with each element $\hat{x}$ of U a number $A(\hat{x})$ in the interval $[0, 1]$: $A(\hat{x})$ represents the membership degree of $\hat{x}$ in A [18]. The support and the core of A are the crisp subsets of A with, respectively, nonzero membership degrees and membership degrees equal to 1.

Though different types of membership functions, such as Gaussian, triangular and trapezoidal, can be used for characterizing fuzzy sets, for the sake of simplicity, we will consider triangular fuzzy sets, which are identified by the tuples (a, b, c), where a and c correspond to the left and right extremes of the support, and b to the core. Formally, a triangular membership function can be defined as follows:

$$A(x) = \begin{cases} \frac{a-x}{b-a} & a \leq x \leq b \\ \frac{c-x}{c-b} & b < x \leq c \\ 0 & otherwise. \end{cases} \tag{2}$$

A variable whose values are linguistic terms is called linguistic variable [18]. A linguistic variable L is characterized by a term set $T(L)$, with each term labelling an information granule defined on universe U. The set of terms $P = \{A_1, \ldots, A_{|T(L)|}\}$, where $|\cdot|$ is the cardinality, constitutes a partition of the universe U.

Usually, a purposely-defined granule $A_{f,0}$ $(f = 1, \ldots, F)$ is considered for all the F input variables. This granule, which represents the "don't care" condition, is defined by a characteristic/membership function equal to 1 on the overall universe. The term $A_{f,0}$ allows generating rules that contain only a subset of the input variables [19].

Given an input pattern $\hat{\mathbf{x}} \in \Re^F$, the strength of activation (*matching degree* of the rule with the input) of the rule R_m is computed as:

$$w_m(\hat{\mathbf{x}}) = \prod_{f=1}^{F} A_{f,j_{m,f}}(\hat{x}_f), \tag{3}$$

where $A_{f,j_{m,f}}(X_f)$ is the characteristic/membership function associated with the granule $A_{f,j_{m,f}}$. For the sake of simplicity, in the formula, we have only considered the product as t-norm for implementing the logical conjunction.

The estimated output class $\hat{C}$ is obtained by first calculating the *association degree* $h_m(\hat{\mathbf{x}})$ with class C_{j_m} for each rule R_m, and then by applying a reasoning method so as to take into account all the rules that constitute the classifier.

The *association degree* $h_m(\hat{\mathbf{x}})$ is computed as:

$$h_m(\hat{\mathbf{x}}) = w_m(\hat{\mathbf{x}}) \cdot RW_m. \tag{4}$$

In this paper, we adopt as rule weight the *certainty factor* CF_m defined as [20, 19]:

$$CF_m = \frac{\sum_{\mathbf{x}_t \in C_{j_m}} w_m(\mathbf{x_t})}{\sum_{t=1}^{N} w_m(\mathbf{x_t})}. \tag{5}$$

The *reasoning method* uses the information from the RB to determine the class label for a given input pattern. We adopt the *maximum matching* as reasoning method: an input pattern is classified into the class corresponding to the rule with the maximum association degree calculated for the pattern. In case of tie, the pattern is classified into the class associated with the most specific rule.

3 Multi-objective Evolutionary Granular Rule-based Systems

The adequate granulation of the input variables affects the accuracy of the rule-based system, but also its interpretability. Accuracy is typically expressed in terms of classification rate. As regards interpretability, it is quite difficult to find a universally accepted index for interpretability assessment since it is a rather subjective and application-dependent concept. Thus, researchers have focused their attention on some factors, which influence interpretability, and on some constraints that have to be satisfied for these factors (see, e.g., [21, 22]). Various semantic and syntactic interpretability issues regarding both the RB and the DB have been taken into account mainly in the framework of fuzzy rule-based systems (see, e.g. [23, 24, 25]).

Recently, the most relevant measures and strategies exploited to design interpretable fuzzy rule-based systems have been reviewed in [26]. Here, a taxonomy of the interpretability measures has been proposed by considering two different dimensions, namely semantics and complexity, at RB and DB levels. In particular, complexity at RB level is expressed in terms of number of rules, total rule length (TRL) and average rule length, while complexity at DB level is determined by the number of attributes and the number of granules. On the other hand, the semantic dimension, when considered at the RB level, concerns aspects like consistency of rules, number of rules fired at the same time, and transparency of the structure, while, at DB level, concepts like coverage of the universes, normalization of the functions characterizing the granules, distinguishability and order of granules are taken into account. In this paper, we have used just a measure of RB complexity, namely TRL.

Accuracy and interpretability are objectives in competition with each other: an increase in the former corresponds typically to a decrease in the latter. The best trade-off between the two objectives generally depends on the application context and cannot be fixed a-priori. Thus, the generation of GRBCs from data taking both the objectives into consideration is a typical multi-objective optimization problem, which can be tackled by using multi-objective evolutionary algorithms (MOEAs). The output of the MOEA is a set of rule-based systems with different trade-offs between accuracy and interpretability: the user can decide for the best solution on the basis of the specific application context.

During the evolutionary process we focus on learning data and rule bases. We generalize to generic information granules the approaches we proposed in [15, 16] for fuzzy sets and regression problems. As regards data base learn-

ing, we aim to learn both the number and the parameters of the information granules. According to psychologists, to preserve interpretability, the number of granules, expressed as linguistic terms, per variable should be 7 ± 2 due to a limit of human information processing capability [27]. Thus, we can fix an upper bound T_{max} for the number of granules. T_{max} is a user-defined parameter which, for specific application domains, might be lower, but should never be higher than 9 for preserving interpretability.

For each variable X_f, we define initial partitions with the maximum possible number T_{max} of granules. These partitions can be provided by an expert, when possible, or can be generated uniformly. These partitions are denoted as *virtual partitions* in the following [15, 16]. During the evolutionary process, rule generation and granule parameter tuning are performed on these virtual partitions. The actual granularity is used only in the computation of the objectives. In practice, we generate RBs, denoted as *virtual RBs*, and tune granule parameters by using virtual partitions, but assess their quality using each time different "lens" depending on the actual number of granules used to partition the single variables. Thus, we do not worry about the actual number of granules in applying crossover and mutation operators. Obviously, to compute the fitness we have to transform the virtual GRBC into the actual GRBC and this process requires to define appropriate mapping strategies, both for the RB and for the granule parameters.

3.1 RB mapping strategy

To map the virtual RB defined on variables partitioned with T_{max} granules into a concrete RB defined on variables partitioned with T_f granules, we adopt the following simple mapping strategy proposed in [15, 16]. Let X_f **is** $\widehat{A}_{f,h}$, $h \in [1, T_{max}]$, be a generic proposition defined in a rule of the virtual RB. Then, the proposition will be mapped to X_f **is** $\widetilde{A}_{f,s}$, with $s \in [1, T_f]$, where $\widetilde{A}_{f,s}$ is the granule most similar to $\widehat{A}_{f,h}$ among the T_f granules $\widehat{A}_{f,h}$ defined on X_f. The definition of similarity depends on the specific type of granules considered in the rule-based system.

In the case of fuzzy sets, we can trivially consider as similarity measure the distance between the centers of the cores of the two fuzzy sets. If there are two fuzzy sets in the partition with centers of the cores at the same distance from the center of the core of $\widehat{A}_{f,h}$, we choose randomly one of the two fuzzy sets. In the case of intervals, similarly to the fuzzy case, we consider as similarity measure the distance between the centers of the two intervals. Since during the evolutionary process endpoints are constrained to vary within a pre-fixed range, this measure, although quite coarse, can be considered adequate.

Note that different rules of the virtual RB can be mapped to equal rules in the concrete RB. This occurs because distinct granules defined on the partitions used in the virtual RB can be mapped to the same granule defined on the partitions used in the concrete RB. In the case of equal rules, only one of these rules is considered in the concrete RB. The original different rules are,

however, maintained in the virtual RB. Indeed, when the virtual RB will be interpreted by using different "lens", all these rules can again be meaningful and contribute to increase the accuracy of the granular rule-based system. Thus, the concept of virtual RB allows us to explore the search space and concurrently exploit the optimal solutions achieved during the evolutionary process.

3.2 Granule parameter mapping strategy

As regards the granule parameter tuning, we approach the problem by using a piecewise linear transformation [28, 29, 15]. We start from an initial partition of the input variables and tune the parameters of the granules, which compose the partition, by applying this transformation. Let $\widetilde{P}_f = \left\{\widetilde{A}_{f,1}, \ldots, \widetilde{A}_{f,T_f}\right\}$ and $P_f = \left\{A_{f,1}, \ldots, A_{f,T_f}\right\}$ be the initial and the transformed partitions, respectively. In the following, we assume that the universes $\widetilde{U}_f$ and U_f of the two partitions are identical. Further, we consider each variable normalized in $[0, 1]$.

Let $t(x_f) : U_f \rightarrow \widetilde{U}_f$ be the piecewise linear transformation. We have that $A_{f,j}(x_f) = \widetilde{A}_{f,j}\left(t\left(x_f\right)\right) = \widetilde{A}_{f,j}\left(\widetilde{x}_f\right)$, where $\widetilde{A}_{f,j}$ and $A_{f,j}$ are two generic granules from the initial and transformed partitions, respectively. As observed in [29], the transformation must be non-decreasing. We define the piecewise linear transformation by considering one representative for each granule. In the case of fuzzy sets, we assume that the representative coincides with the center of the core. In the case of intervals, it corresponds to the center of the interval. The representatives determine the change of slopes of the piecewise linear transformation $t(x_f)$ for each variable X_f. Let $\widetilde{b}_{f,1}, \ldots, \widetilde{b}_{f,T_f}$ and $b_{f,1}, \ldots, b_{f,T_f}$ be the representatives of $\widetilde{A}_{f,1}, \ldots, \widetilde{A}_{f,T_f}$ and $A_{f,1}, \ldots, A_{f,T_f}$, respectively. Transformation $t(x_f)$ is defined as:

$$t(x_f) = \frac{\widetilde{b}_{f,j} - \widetilde{b}_{f,j-1}}{b_{f,j} - b_{f,j-1}} \cdot \left(x_f - b_{f,j-1}\right) + \widetilde{b}_{f,j-1} \tag{6}$$

with $b_{f,j-1} \leq x_f \leq b_{f,j}$.

Once defined transformation $t(x_f)$, all the parameters which define the granules are transformed using $t(x_f)$. As an example, we consider triangular fuzzy sets as granules. Further, we assume that the initial partition is a uniform partition (see Fig. 1). Thus, $b_{f,1}$ and b_{f,T_f} coincide with the extremes of the universe U_f of X_f. It follows that $t(x_f)$ depends on $T_f - 2$ parameters, that is, $t\left((x_f; b_{f,2}, \ldots, b_{f,T_f-1})\right)$ [15]. Once fixed $b_{f,2}, \ldots, b_{f,T_f-1}$, the partition P_f can be obtained simply by transforming the three points $(\widetilde{a}_{f,j}, \widetilde{b}_{f,j}, \widetilde{c}_{f,j})$, which describe the generic fuzzy set $\widetilde{A}_{f,j}$, into $(a_{f,j}, b_{f,j}, c_{f,j})$ applying $t^{-1}(\widetilde{x}_f)$. In those regions where $t(x_f)$ has a high value of the derivative (high slope of the lines), the fuzzy sets are narrower; otherwise, the fuzzy sets $A_{f,j}$ are wider. We define the piecewise linear transformation on the maximum granularity T_{max}.

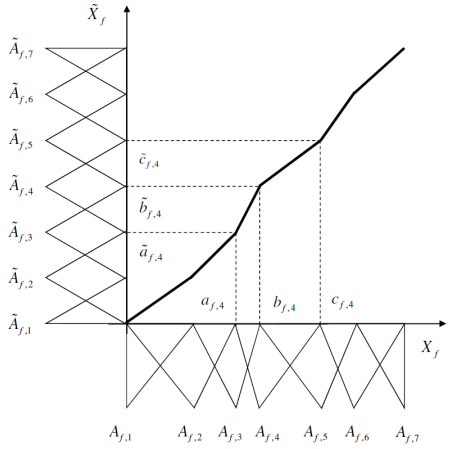

Fig. 1 An example of piecewise linear transformation

In the case of intervals, we adopt a similar strategy. Each interval $A_{f,j}$ is defined by its endpoints $[a_{f,j}, c_{f,j}]$. For similarity to the fuzzy case, we assume that $b_{f,1}$ and b_{f,T_f} coincide with the extremes of the universe U_f of X_f. The partition P_f can be obtained by simply transforming the two endpoints $(\widetilde{a}_{f,j}, \widetilde{c}_{f,j})$, which describe the generic interval $\widetilde{A}_{f,j}$, into $(a_{f,j}, c_{f,j})$ applying $t^{-1}(\widetilde{x}_f)$.

When we reduce the granularity, in order to maintain the original shape of the granules, we apply $t^{-1}(\widetilde{x}_f)$ for $j = 2, \ldots, T_f - 1$, where $T_f \geq 3$ is the actual granularity, only to the parameters which describe the granule. In the case of fuzzy sets, we apply the transformation only to the three points $(\widetilde{a}_{f,j}, \widetilde{b}_{f,j}, \widetilde{c}_{f,j})$, which describe the generic fuzzy set $\widetilde{A}_{f,j}$. Fig. 2 shows an example of this transformation for granularity $T_f = 5$ by using the piecewise linear transformation in Fig. 1, defined with granularity $T_{max} = 7$. In the case of intervals, we apply the transformation only to the two endpoints $(\widetilde{a}_{f,j}, \widetilde{c}_{f,j})$, which describe the generic interval $\widetilde{A}_{f,j}$.

3.3 Chromosome coding

As shown in Fig. 3, each solution is codified by a chromosome C composed of three parts (C_R, C_G, C_T), which define the rule base, the number of granules, and the positions of the representatives of the granules in the transformed space, respectively.

In particular, C_R contains, for each rule R_m, the index $j_{m,f}$ of the antecedent, for each input variable X_f, and the consequent class C_{j_m}. Thus, C_R is composed by $M \cdot (F + 1)$ natural numbers where M is the number of rules currently present in the virtual RB. The RB (defined as concrete RB) used to compute the fitness is obtained by means of the RB mapping strategy using

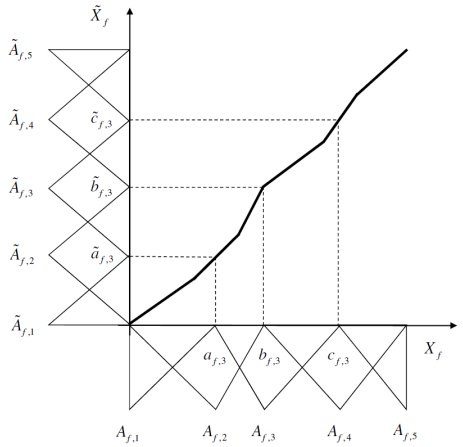

Fig. 2 An example of piecewise linear transformation with granularity $T_f = 5$ different from $T_{max} = 7$

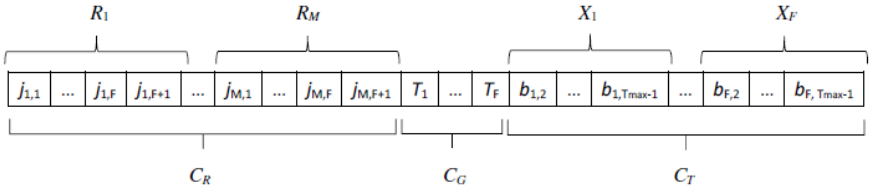

Fig. 3 Chromosome coding

the actual granularities fixed by C_G. We assume that at most M_{max} rules can be contained in the RB.

C_G is a vector containing F natural numbers: the f^{th} element of the vector contains the number $T_f \in [2, T_{max}]$ of granules, which partition variable X_f. T_{max} is fixed by the user and is the same for all the variables.

C_T is a vector containing F vectors of $T_{max}-2$ real numbers: the f^{th} vector contains $[b_{f,2}, \ldots, b_{f,T_{max}-1}]$ points, which define where the granule representatives are moved and consequently the piecewise linear transformation.

To preclude that the piecewise linear transformation can become decreasing, we force $b_{f,j}$ to vary in $\left[\widetilde{b}_{f,j} - \frac{\widetilde{b}_{f,j}-\widetilde{b}_{f,j-1}}{2}, \widetilde{b}_{f,j} + \frac{\widetilde{b}_{f,j}-\widetilde{b}_{f,j-1}}{2}\right], \forall j \in [2, T_{max}-1]$.

3.4 Mating Operators

In order to generate the offspring populations, we exploit both crossover and mutation. We apply separately the one-point crossover to C_R and C_G and the BLX-α-crossover, with $\alpha = 0.5$, to C_T. Let s_1 and s_2 be two selected parent chromosomes. The common gene for C_G is extracted randomly in $[1, F]$. The common gene for C_R is selected by extracting randomly a number in

$[1, \rho_{min} - 1]$, where ρ_{min} is the minimum number of rules in s_1 and s_2. The crossover point is always chosen between two rules and not within a rule. When we apply the one-point crossover to C_R, we can generate GRBC with one or more pairs of equal rules. In this case, we simply eliminate one of the rules from each pair. This allows us to reduce the total number of rules.

As regards mutation, we apply two mutation operators for C_R. The first operator adds γ rules to the virtual RB, where γ is randomly chosen in $[1, \gamma_{max}]$. The upper bound γ_{max} is fixed by the user. If $\gamma + M > M_{max}$, then $\gamma = M_{max} - M$. For each rule R_m added to the chromosome, we generate a random number $v \in [1, F]$, which indicates the number of input variables used in the antecedent of the rule. Then, we generate v natural random numbers between 1 and F to determine the input variables which compose the antecedent part of the rule. Finally, for each selected input variable f, we generate a random natural number $j_{m,f}$ between 0 and T_{max}, which determines the granule $A_{f,j_{m,f}}$ to be used in the antecedent of rule R_m in the virtual RB. To select the consequent, a random number between 1 and the number K of classes is generated.

The second mutation operator randomly changes δ propositions of the virtual RB. The number δ is randomly generated in $[1, \delta_{max}]$. The upper bound δ_{max} is fixed by the user. For each element to be modified, a number is randomly generated in $[0, T_{max}]$.

The mutation applied to C_G randomly chooses a gene $f \in [1, F]$ and changes the value of this gene by randomly adding or subtracting 1. If the new value is lower than 2 or larger than T_{max}, then the mutation is not applied.

The mutation applied to C_T first chooses randomly a variable X_f, then extracts a random value $j \in [2, T_{max} - 1]$ and changes the value of $b_{f,j}$ to a random value in the allowed interval. We experimentally verified that these mating operators ensure a good balancing between exploration and exploitation, thus allowing the MOEA described in the next subsection to create good approximations of the Pareto fronts.

3.5 Multi-objective evolutionary algorithm

As MOEA we use the (2+2)M-PAES that has been successfully employed in our previous works [30, 31, 32]. Each chromosome is associated with a bi-dimensional objective vector. The first element of the vector measures the complexity of the granular rule-based system as TRL, that is the number of propositions used in the antecedents of the rules contained in the concrete RB (the number of rules may be different between the virtual and concrete RBs). The second element assesses the accuracy in terms of classification rate.

(2+2)M-PAES, which is a modified version of the well-known (2+2)PAES introduced in [33], is a steady state multi-objective evolutionary algorithm which uses two current solutions s_1 and s_2 and stores the non dominated solutions in an archive. Unlike the classical (2+2)PAES, which maintains the

current solutions until they are not replaced by solutions with particular characteristics, we randomly extract, at each iteration, the current solutions. If the archive contains a unique solution, s_1 and s_2 correspond to this unique solution.

At the beginning, the archive is initialized as an empty structure and two initial current solutions s_1 and s_2 are randomly generated. At each iteration, the application of crossover and mutation operators produces two new candidate solutions, o_1 and o_2, from the current solutions s_1 and s_2. These candidate solutions are added to the archive only if they are dominated by no solution contained in the archive; possible solutions in the archive dominated by the candidate solutions are removed. Typically, the size of the archive is fixed at the beginning of the execution of the (2+2)M-PAES. In this case, when the archive is full and a new solution o_i, where $i = 1, 2$, has to be added to the archive, if it dominates no solution in the archive, then we insert o_i into the archive and remove the solution (possibly o_i itself) that belongs to the region with the highest crowding degree. The crowding degree is calculated by using an adaptive grid defined on the objective space. If the region contains more than one solution, then the solution to be removed is randomly chosen. (2+2)M-PAES terminates after a given number Z of iterations. The candidate solution acceptance strategy generates an archive which contains only non-dominated solutions. On (2+2)M-PAES termination, the archive includes the set of solutions which are an approximation of the Pareto front.

4 Experimental results

In this Section, we aim to show how learning number and/or parameters of information granules affects the accuracy and interpretability of GRBCs. We will discuss the two types of granules investigated in this paper separately. The analysis has been carried out by executing (2+2)M-PAES in four different modalities. First, we have executed (2+2)M-PAES for learning only the rules by using a uniform partition with $T_f = 5$, $f = 1..F$, granules for each input variable. This value has proved to be the most effective in our previous works [15, 16]. We have denoted this modality as PAES-R. In practice, we have used only the C_R part of chromosome C during the evolutionary process. Second, we have executed (2+2)M-PAES for learning the rules and the parameters of the granules, using an initial uniform partition with $T_f = 5$, $f = 1..F$, granules for each input variable. We have denoted this modality as PAES-RT. In practice, we have used the C_R and C_T parts of chromosome C during the evolutionary process. Third, we have executed (2+2)M-PAES for learning the rules and the number of granules, using a uniform partition with $T_{max} = 7$, $f = 1..F$, granules for each input variable. This value has been suggested in [27]. We have denoted this modality as PAES-RG. In practice, we have used the C_R and C_G parts of chromosome C during the evolutionary process. Fourth, we have executed (2+2)M-PAES for learning the rules, the number of granules and the parameters of the granules, using virtual uniform partitions with $T_{max} = 7$,

$f = 1..F$, granules for each input variable. We have denoted this modality as PAES-RGT. In this last case, we have used the overall chromosome C during the evolutionary process. We decided to adopt these different modalities for evaluating the impact on accuracy and interpretability of each different level of granulation.

We aim to compare the different solutions on the Pareto fronts in the four different modalities, to evaluate how the different levels of granulation can affect the accuracy and the interpretability of the GRBCs. We executed the four modalities of (2+2)M-PAES on twenty-four classification datasets extracted from the KEEL repository[1]. As shown in Table 1, the datasets are characterized by different numbers of input variables (from 3 to 19), input/output instances (from 80 to 19020) and classes (from 2 to 8). For the datasets CLE, HEP, MAM, and WIS, we removed the instances with missing values. The number of instances in the table refers to the datasets after the removing process. For each dataset, we performed a ten-fold cross-validation and executed three trials for each fold with different seeds for the random function generator (30 trials in total). All the results presented in this section are obtained by using the same folds for all the algorithms. Table 2 shows the parameters of (2+2)M-PAES used in the experiments.

Since several solutions can lie on the Pareto front approximations, typically only some representative solutions are considered in the comparison. In our previous papers [35, 31] and also in [11], for each fold and each trial, the Pareto front approximations of each algorithm are computed and the solutions are sorted in each approximation according to decreasing accuracies on the training set. Then, for each approximation, we select the first (the most accurate), the median and the last (the least accurate) solutions. We denote these solutions as FIRST, MEDIAN and LAST, respectively. Finally, for the three solutions, we compute the average values over all the folds and trials of the accuracy on both the training and the test sets, and of the TRL. On the one side, the three solutions allow us to graphically show the average trend of the Pareto front approximations obtained in the executions performed on the different folds. On the other side, we can analyze how these solutions are able to generalize when applied to the test set.

In the following, we discuss the results obtained by applying the four modalities of the (2+2)M-PAES execution on fuzzy sets and on intervals.

4.1 Fuzzy Sets

Figures 4 and 5 show the FIRST, MEDIAN and LAST solutions obtained by the execution of PAES-R, PAES-RT, PAES-RG and PAES-RGT. In the figures, the x and y axes indicate the complexity calculated as TRL and the accuracy expressed in terms of classification rate. We can realize how the three solutions allow us to visualize the trend of the average Pareto front

[1] available at http://sci2s.ugr.es/keel/datasets.php)[34]

Table 1 Datasets used in the experiments

Dataset	# Instances	# Variables	# Classes
Appendicitis (APP)	106	7	2
Australian (AUS)	690	14	2
Bands (BAN)	365	19	2
Bupa (BUP)	345	6	2
Cleveland (CLE)	297	13	5
Ecoli (ECO)	336	7	8
Glass(GLA)	214	9	6
Haberman (HAB)	306	3	2
Hayes-roth (HAY)	160	3	3
Heart (HEA)	270	13	2
Hepatitis (HEP)	80	19	2
Iris (IRI)	150	4	3
Magic (MAG)	19020	10	2
Mammographic (MAM)	830	5	2
Monk-2 (MON)	432	6	2
Newthyroid (NEW)	215	5	3
Page-blocks (PAG)	5472	10	5
Phoneme (PHO)	5404	5	2
Pima (PIM)	768	8	2
Saheart (SAH)	462	9	2
Tae (TAE)	151	5	3
Vehicle (VEH)	846	18	4
Wine (WIN)	178	13	3
Wisconsin (WIS)	683	9	2

Table 2 Values of the parameters used in the experiments for (2+2)M-PAES

AS	(2+2)M-PAES archive size	64
M_{max}	Maximum number of rules in an RB	50
P_{CC_R}	Probability of applying the crossover operator to C_R	0.6
P_{CC_G}	Probability of applying the crossover operator to C_G	0.5
P_{CC_T}	Probability of applying the crossover operator to C_T	0.5
$P^1_{MC_R}$	Probability of applying the first mutation operator to C_R	0.55
γ_{max}	Upper bound of the added rules in the first mutation operator for C_R	2
$P^2_{MC_R}$	Probability of applying the second mutation operator to C_R	0.45
δ_{max}	Upper bound of the changed propositions in the second mutation operator for C_R	2
P_{MC_G}	Probability of applying the mutation operator to C_G	0.2
P_{MC_T}	Probability of applying the mutation operator to C_T	0.9
Z	Number of iterations of (2+2)M-PAES	50000

approximations. Further, by comparing the accuracies of the three solutions on the training and test sets, we can verify whether these solutions, especially the FIRST solution, suffer from overtraining. Indeed, the FIRST solution is in general the most prone to overtraining since it achieves the highest accuracy on the training set. We can observe from the plots that there exists some difference for all the three solutions between the classification rates obtained on the training set and the ones achieved on the test set. Thus, we can conclude that the decrease of performance between training and test sets does not occur only for the FIRST solution. In general, we observe that the fronts generated by PAES-R and PAES-RT are small and concentrated in an area of low TRL values. The fronts generated by PAES-RG and PAES-RGT are on average wider than the ones generated by PAES-R and PAES-RT and concentrated

in an area with higher TRL values. This different distribution of the fronts is mainly due to the value of T_{max} used for learning the number of granules during the evolutionary process. Indeed, we adopt $T_{max} = 7$ for each input variable for PAES-RG and PAES-RGT, while we adopt $T_f = 5$ for each input variable for PAES-R and PAES-RT. A higher possible number of fuzzy sets in the partitions induces a higher number of rules. Indeed, since the rules can adopt more precise fuzzy sets at least for the most difficult attributes, they tend to be more specialized. Thus, a higher number of rules is needed to cover the dataset instances. On the other hand, this allows us to achieve in general higher accuracies. Anyway, this specialization occurs only for a limited number of attributes, which adopt a high value of granularity. The other attributes are characterized by a low value of granularity, thus highlighting the good characteristics of our granulation learning approach.

For the sake of fairness, we have executed the four algorithms using the same number of iterations. On the other hand, we have to consider that the granularity learning increases the search space and therefore would need a higher number of iterations. This is true, in particular, for PAES-RGT which has to cope with the highest search space.

Table 3 shows the numerical values for the FIRST solutions obtained by PAES-R, PAES-RT, PAES-RG and PAES-RGT. For the sake of brevity, we do not show the values of the MEDIAN and LAST solutions.

To statistically validate the results, we apply a non-parametric statistical test for multiple comparisons by using all the datasets. First, we generate a distribution consisting of the mean values of the accuracies of the three solutions calculated on the test set. Then, we apply the Friedman test in order to compute a ranking among the distributions [36], and the Iman and Davenport test [37] to evaluate whether there exists a statistical difference among the distributions. If the Iman and Davenport p-value is lower than the level of significance α (in the experiments $\alpha = 0.05$), we can reject the null hypothesis and affirm that there exist statistical differences among the multiple distributions associated with each approach. Otherwise, no statistical difference exists. If there exists a statistical difference, we apply a post-hoc procedure, namely the Holm test [38]. This test allows detecting effective statistical differences between the control approach, i.e. the one with the lowest Friedman rank, and the remaining approaches.

Table 4 shows the results of the statistical tests for the fuzzy set granules on the classification rate computed on the test set. We observe that for the FIRST solution the null hypothesis is rejected. The Holm post-hoc procedure, executed using PAES-RGT as control algorithm, states that only PAES-RG is statistically equivalent to PAES-RGT. We can conclude that the learning of the number of granules allows generating solutions with higher accuracy. From Table 3 we can realize, however, that these solutions are obtained at the expense of a higher complexity. For the MEDIAN and LAST solutions, the null hypothesis is not rejected, thus stating that these solutions are statistically equivalent in terms of accuracy. However, the solutions generated by PAES-R and PAES-RT are typically characterized by a lower complexity, as we can

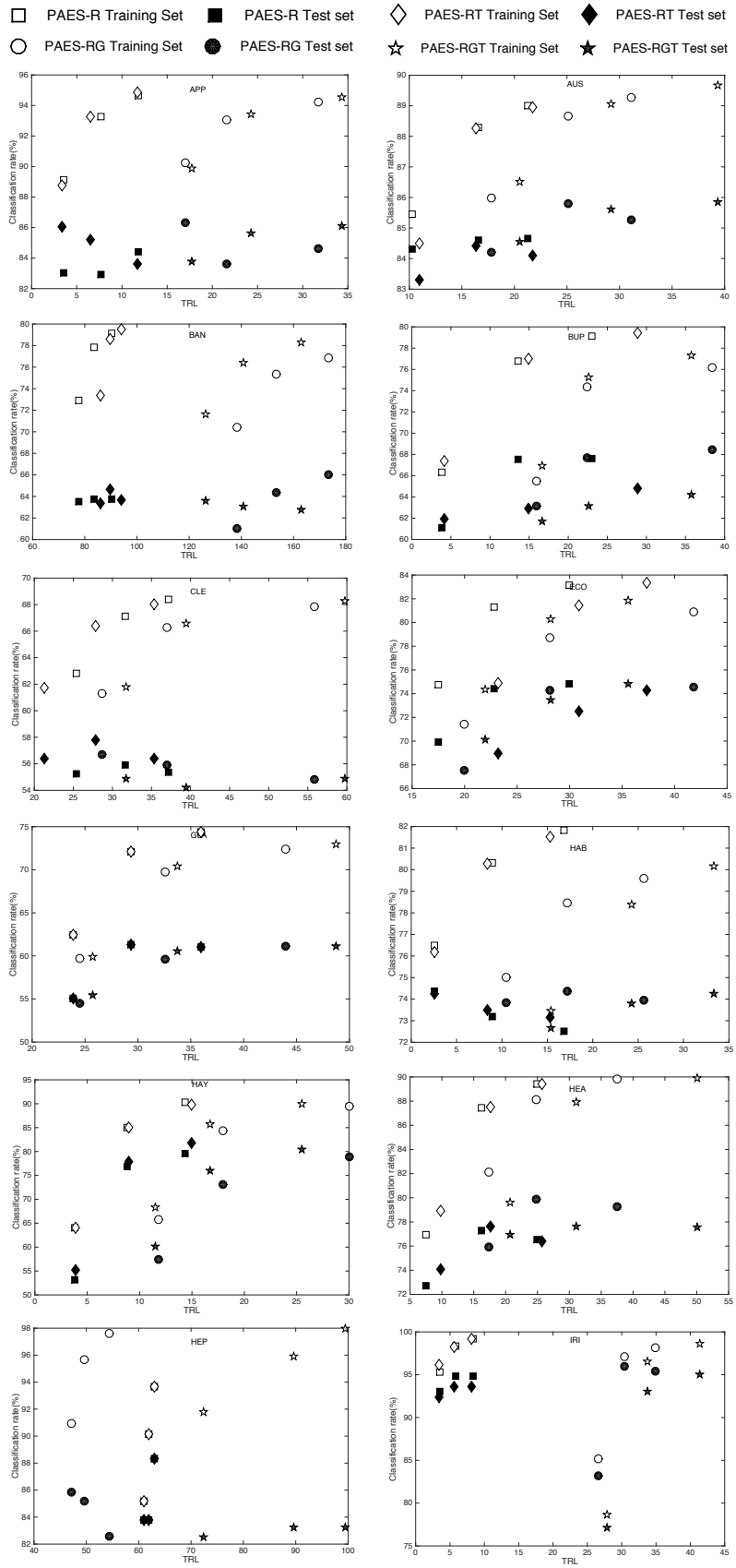

Fig. 4 Pareto front approximations visualized by using the three representative points for PAES-R, PAES-RT, PAES-RG and PAES-RGT, employing fuzzy sets as information granules (first group of datasets)

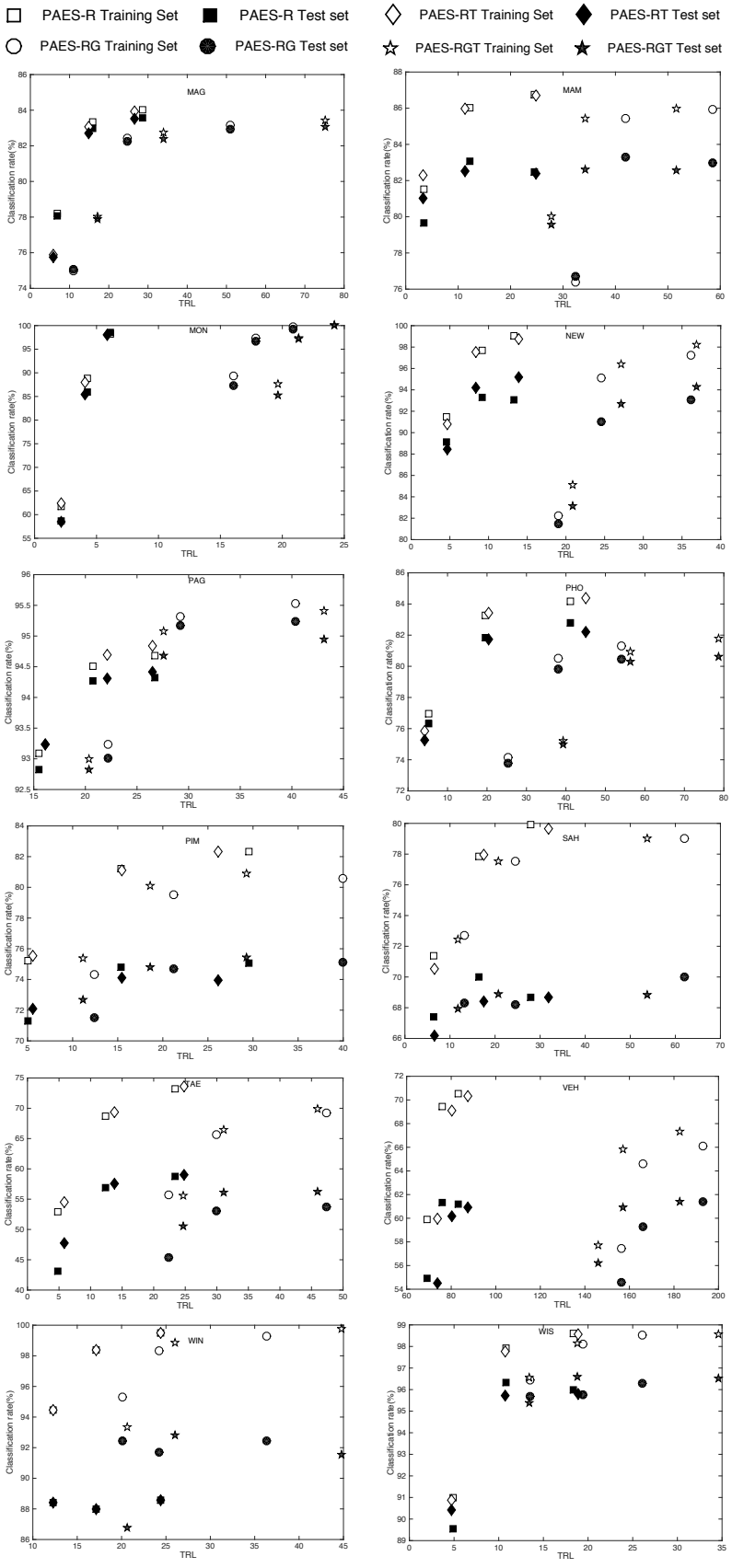

Fig. 5 Pareto front approximations visualized by using the three representative points for PAES-R, PAES-RT, PAES-RG and PAES-RGT, employing fuzzy sets as information granules (second group of datasets)

derive from Figures 4 and 5. Concluding, we can observe that the different levels of granulation actually affect the performance of the classifiers. When we learn the number of granules for each attribute and we learn the membership parameters together with the rules, we obtain the best performance in terms of accuracy, although with a higher complexity.

Table 3 Average accuracy on the training (AccTr) and test (AccTs) sets, TRL and number of rules (R) for the FIRST solutions generated by PAES-R, PAES-RT, PAES-RG and PAES-RGT, employing fuzzy sets as information granules

	PAES-R				PAES-RT				PAES-RG				PAES-RGT			
	AccTr	AccTs	TRL	R	AccTr	AccTs	TRL	R	AccTr	AccTs	TRL	R	AccTr	AccTs	TRL	R
APP	94.64	84.44	11.8	9.9	94.85	83.64	11.8	9.9	94.24	84.64	31.8	17.5	94.54	86.09	34.3	18.7
AUS	88.99	84.66	21.3	15.6	88.96	84.11	21.8	17.1	89.26	85.27	31.1	20.7	89.66	85.85	39.4	23.6
BAN	79.14	63.77	90.2	41.3	79.47	63.65	93.9	41.6	76.83	66	173.3	47.8	78.26	62.77	162.9	46.9
BUP	79.1	67.65	23.1	13.6	79.45	64.84	28.9	15.1	76.21	68.48	38.4	18	77.27	64.24	35.7	17.7
CLE	68.41	55.38	37.2	28.8	68.04	56.37	35.4	27.2	67.88	54.82	55.8	32.8	68.25	54.89	59.7	34.4
ECO	83.14	74.83	29.9	22.8	83.34	74.26	37.3	25	80.9	74.55	41.8	26.8	81.84	74.85	35.6	25.3
GLA	74.36	61.06	36	25.1	74.36	61.06	36	25.1	72.4	61.15	44	28.3	72.97	61.11	48.8	31.3
HAB	81.84	72.52	16.8	10.9	81.53	73.15	15.3	10.1	79.58	73.93	25.6	13.9	80.15	74.26	33.4	17
HAY	90.25	79.58	14.4	11	89.88	81.88	15	11.2	89.44	78.96	30	18.2	89.95	80.42	25.5	15.9
HEA	89.44	76.54	24.9	18.3	89.4	76.42	25.7	18.2	89.85	79.26	37.5	22.3	89.89	77.53	50.1	26.3
HEP	93.68	88.31	63	39.5	93.68	88.31	63	39.5	97.6	82.59	54.3	31.7	97.95	83.25	99.4	38.5
IRI	99.16	94.85	8.4	6.6	99.16	93.64	8.1	6.5	98.2	95.36	34.9	17.3	98.65	95.07	41.3	19.9
MAG	84.01	83.59	28.7	16	83.93	83.52	26.5	14.4	83.16	82.93	50.9	21.2	83.43	83.07	75.2	29
MAM	86.76	82.47	24.6	14.3	86.73	82.37	24.9	14	85.94	82.96	58.5	24.3	86	82.59	51.6	20.9
MON	98.15	98.51	6.1	5.7	98.08	98.12	5.9	5.6	99.67	99.24	20.9	13.5	100	100	24.2	13.7
NEW	99.02	93.1	13.2	9.1	98.72	95.22	13.9	9.8	97.24	93.06	36.2	18.3	98.19	94.26	36.8	18.5
PAG	94.68	94.32	26.8	19.7	94.84	94.41	26.5	19.8	95.52	95.24	40.4	24.4	95.42	94.95	43.1	26.6
PHO	84.18	82.8	41.2	18.4	84.36	82.23	45	19.5	81.3	80.46	54	22.4	81.78	80.61	78.7	28.4
PIM	82.34	75.05	29.5	17.5	82.3	73.96	26.2	16.7	80.59	75.14	39.9	19.1	80.9	75.44	29.3	16.8
SAH	79.91	68.69	28	17.7	79.64	68.7	31.9	18.6	79.01	69.99	62.1	26.8	79.02	68.83	53.8	24.5
TAE	73.17	58.81	23.4	15.9	73.66	59.03	24.8	15.7	69.17	53.75	47.4	23.2	69.88	56.18	46	22.5
VEH	70.53	61.2	83	40.5	70.35	60.91	87.6	41.5	66.13	61.38	193.1	49.6	67.36	61.39	182.6	48
WIN	99.51	88.55	24.4	19.5	99.51	88.55	24.4	19.5	99.27	92.44	36.4	25.1	99.76	91.53	44.7	27.1
WIS	98.58	95.98	18.4	15.2	98.56	95.78	18.9	15	98.54	96.31	26.1	17.2	98.58	96.51	34.6	19.9

Table 4 Results of the non-parametric statistical tests on the accuracy computed on the test set for the FIRST, MEDIAN and LAST solutions generated by PAES-R, PAES-RT, PAES-RG and PAES-RGT, employing fuzzy sets as information granules

FIRST					
	Algorithm	*Friedman rank*	*Iman and Davenport p-value*		*Hypothesis*
	PAES-R	2.979			
	PAES-RT	2.792			
	PAES-RG	2.250	0.0042		Rejected
	PAES-RGT	2.041			
		Holm post-hoc procedure			
i	*Algorithm*	*z-value*	*p-value*	*alpha/i*	*Hypothesis*
3	PAES-R	2.515576	0.011884	0.016	Rejected
2	PAES-RT	1.844756	0.065073	0.025	Rejected
1	PAES-RG	0.559017	0.57615	0.05	Not Rejected
MEDIAN					
	Algorithm	*Friedman rank*	*Iman and Davenport p-value*		*Hypothesis*
	PAES-R	2.833			
	PAES-RT	2.416			
	PAES-RG	2.333	0.540		Not Rejected
	PAES-RGT	2.416			
LAST					
	Algorithm	*Friedman rank*	*Iman and Davenport p-value*		*Hypothesis*
	PAES-R	2.6042			
	PAES-RT	2.5625			
	PAES-RG	2.5	0.897		Not Rejected
	PAES-RGT	2.3333			

4.2 Intervals

Figures 6 and 7 show the FIRST, MEDIAN and LAST solutions obtained by the execution of PAES-R, PAES-RT, PAES-RG and PAES-RGT using intervals as granules.

As we have already observed with the fuzzy sets, the fronts generated by PAES-R and PAES-RT are small and concentrated in an area of low TRL values. The fronts generated by PAES-RG and PAES-RGT are on average wider than the ones generated by PAES-R and PAES-RT, and concentrated in an area with higher TRL values. This different distribution of the fronts can be explained using the same motivations advanced for the case of fuzzy sets.

Table 5 shows the numerical values for the FIRST solutions obtained by PAES-R, PAES-RT, PAES-RG and PAES-RGT. For the sake of brevity, we do not show the values of the MEDIAN and LAST solutions.

To statistically validate the results, we apply a non-parametric statistical test for multiple comparisons by using all the datasets. Table 6 shows the results of the statistical tests for the interval granules on the classification rate computed on the test set. We observe that for the FIRST solution the null hypothesis is rejected. The Holm post-hoc procedure, executed using PAES-RGT as control algorithm, states that PAES-RG is statistically equivalent to PAES-RGT. We can conclude that the learning of the number of granules allows generating solutions with higher accuracy. Also for the MEDIAN and LAST solutions, we have the same situation: The Holm post-hoc procedure, executed using PAES-RGT as control algorithm, states that PAES-RG is statistically equivalent to PAES-RGT. The solutions generated by PAES-R and PAES-RT are however characterized by a lower complexity, as we can derive from Figures 6 and 7. Concluding, we can again observe that by learning the number of granules for each attribute and the interval parameters together with the rules, we obtain the best performance in terms of accuracy, although with a higher complexity.

5 Conclusions

In this paper we have dealt with the problem of developing accurate and easily comprehensible granular rule-based classifiers (GRBCs). The classification rules are learnt from data and describe the involved variables in terms of information granules, i.e., abstract entities that represent essential aspects of knowledge and system modeling. The considered GRBCs have been generated through a multi-objective evolutionary process, considering accuracy and interpretability as the objectives to be optimized. Information granules have been formalized in terms of sets (in particular, intervals) and fuzzy sets. With reference to well-known classification benchmark datasets, we have shown how granulation tuning (i.e., partition adaptation) and/or granulation learning (i.e., learning of the number of partition components) can effectively influ-

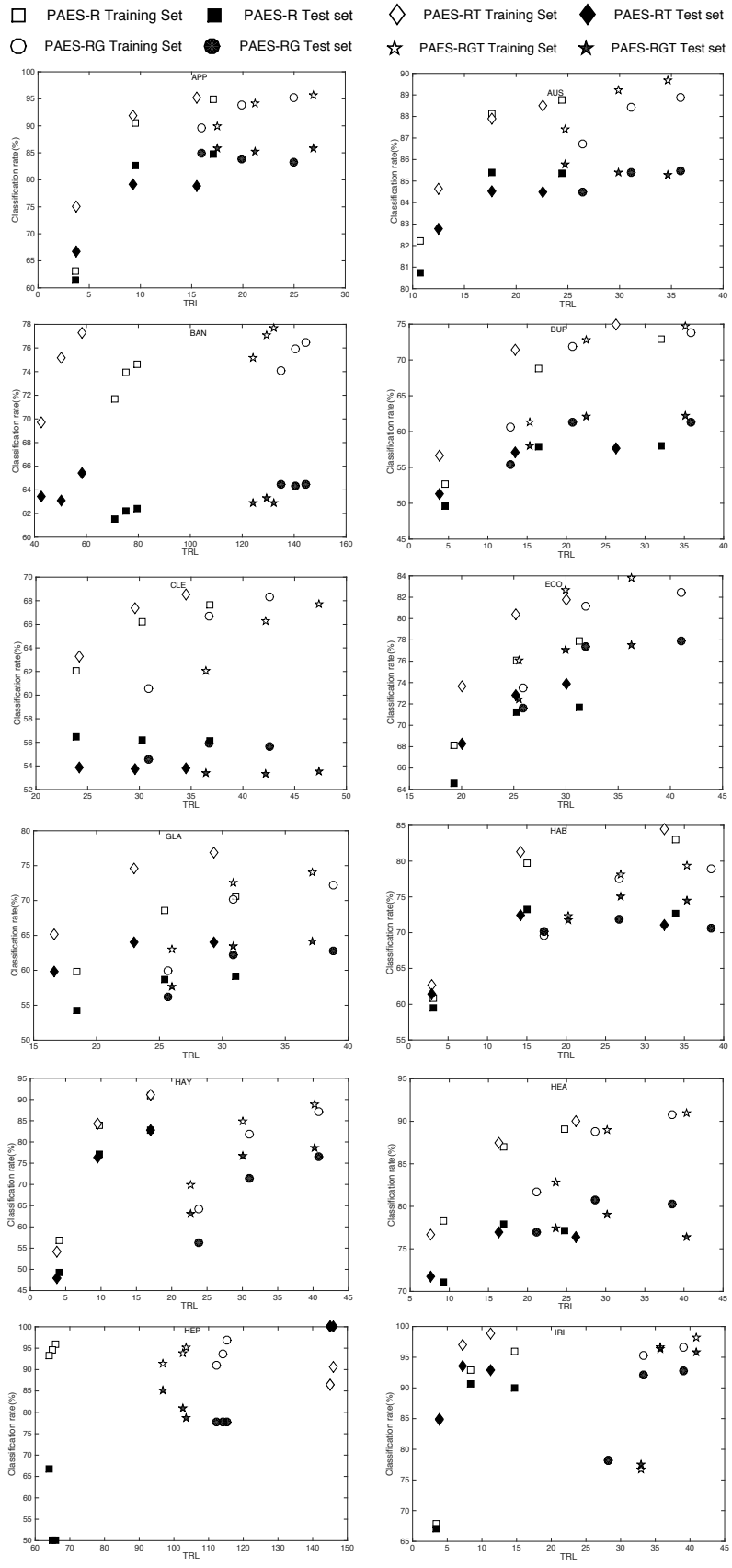

Fig. 6 Pareto front approximations visualized by using the three representative points for PAES-R, PAES-RT, PAES-RG and PAES-RGT, employing intervals as information granules (first group of datasets)

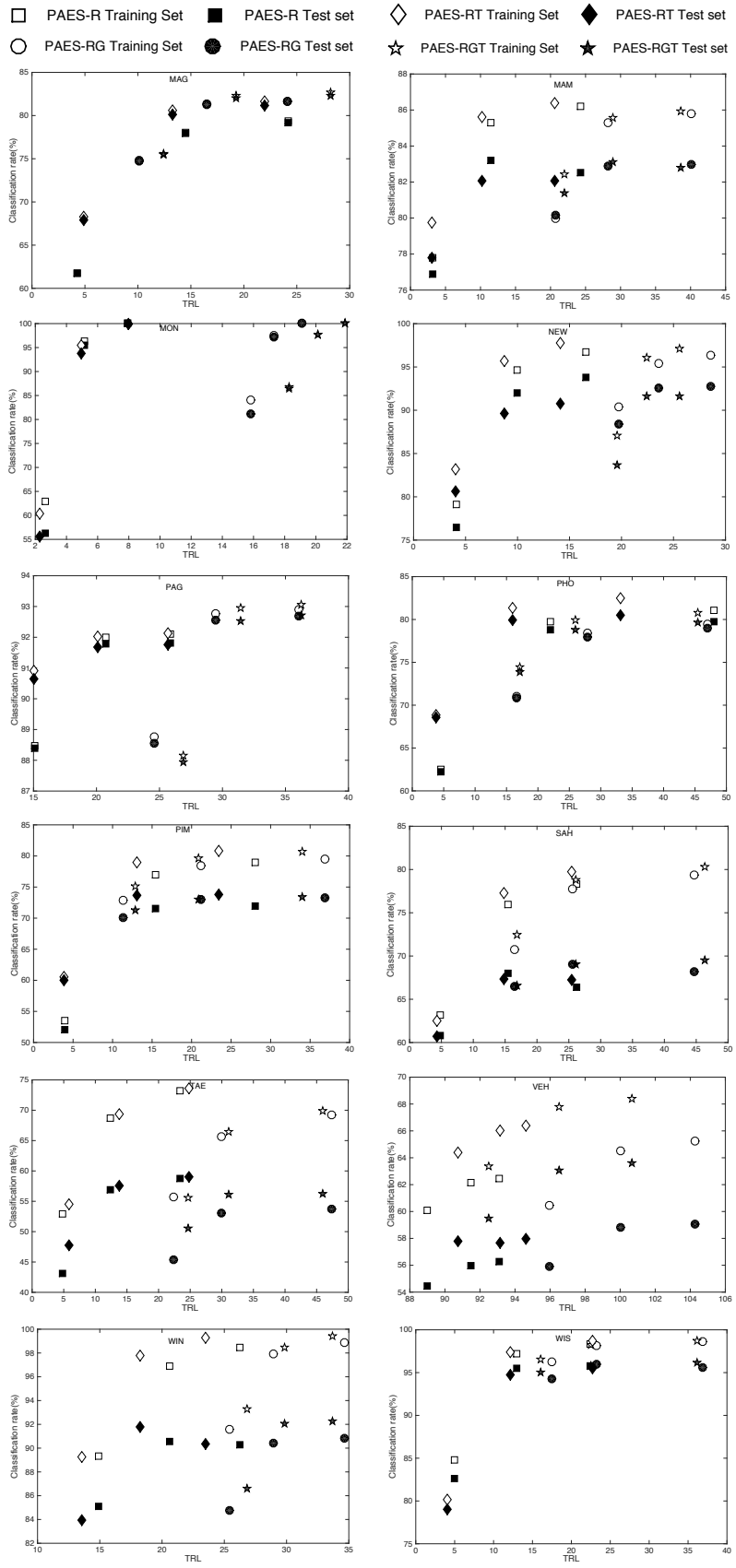

Fig. 7 Pareto front approximations visualized by using the three representative points for PAES-R, PAES-RT, PAES-RG and PAES-RGT, employing intervals as information granules (second group of datasets)

Table 5 Average accuracy on the training (AccTr) and test (AccTs) sets, TRL and number of rules (R) for the FIRST solutions generated by PAES-R, PAES-RT, PAES-RG and PAES-RGT, employing intervals as information granules

	PAES-R				PAES-RT				PAES-RG				PAES-RGT			
	AccTr	AccTs	TRL	R	AccTr	AccTs	TRL	R	AccTr	AccTs	TRL	R	AccTr	AccTs	TRL	R
APP	94.97	84.70	17.1	14.3	95.28	78.85	15.5	13.9	95.21	83.18	24.9	15.5	95.74	85.77	26.9	17.3
AUS	88.78	85.36	24.4	18.3	88.49	84.49	22.6	16.9	88.87	85.46	35.9	25.9	89.68	85.50	34.6	25.1
BAN	74.63	62.41	79.6	40.4	77.31	65.40	58.3	36.5	76.49	64.43	144.4	47.6	77.67	62.88	132.3	46.6
BUP	72.88	58.06	32.0	19.9	74.92	57.64	26.3	17.4	73.79	61.35	35.9	19.8	74.66	62.20	35.1	19.2
CLE	67.65	56.10	36.8	28.9	68.56	53.80	34.5	28.6	68.30	55.67	42.6	31.4	67.70	53.85	47.4	34.4
ECO	77.88	71.71	31.2	27.0	81.77	73.87	30.0	26.4	82.47	77.89	41.1	30.1	83.81	77.50	36.2	27.8
GLA	70.64	59.20	31.0	25.8	76.92	64.03	29.3	24.9	72.26	62.74	38.8	28.9	74.02	64.15	37.1	28.6
HAB	83.05	72.64	33.9	20.0	84.43	71.08	32.4	20.0	78.90	70.67	38.4	19.6	79.43	74.51	35.3	18.0
HAY	90.88	82.71	17.0	13.1	91.16	82.71	17.1	13.2	87.15	76.46	40.8	22.0	88.77	78.54	40.2	21.3
HEA	89.12	77.16	24.7	18.1	90.00	76.42	26.1	19.7	90.82	80.25	38.5	24.4	90.96	76.42	40.4	25.5
HEP	95.95	50.00	66.0	42.0	90.54	100.00	146.0	50.0	96.85	77.78	115.3	44.7	95.19	78.79	103.5	41.5
IRI	95.95	90.00	14.7	13.2	98.81	92.89	11.3	9.6	96.61	92.82	39.0	19.9	98.21	95.83	41.0	20.0
MAG	79.33	79.16	24.2	17.0	81.67	81.20	21.9	16.0	81.67	81.63	24.1	15.8	82.68	82.34	28.2	18.2
MAM	86.19	82.51	24.3	15.8	86.39	82.06	20.6	13.9	85.81	82.96	40.1	19.2	85.94	82.80	38.6	18.2
MON	100.00	100.00	7.9	6.9	99.91	99.85	8.0	7.0	100.00	100.00	19.1	11.9	100.00	100.00	21.9	13.1
NEW	96.76	93.82	16.6	13.5	97.74	90.76	14.1	11.7	96.35	92.72	28.6	18.0	97.11	91.62	25.6	15.8
PAG	92.10	91.82	25.9	21.3	92.14	91.75	25.7	21.7	92.89	92.68	36.0	27.3	93.06	92.70	36.2	28.6
PHO	81.12	79.74	48.0	24.9	82.48	80.47	33.2	19.4	79.42	78.95	47.0	21.9	80.75	79.64	45.4	20.6
PIM	78.94	71.92	28.1	18.5	80.78	73.75	23.4	17.4	79.47	73.28	36.9	20.5	80.68	73.41	34.0	19.8
SAH	78.30	66.43	26.3	17.7	79.73	67.24	25.5	18.6	79.35	68.18	44.7	23.9	80.29	69.55	46.3	23.5
TAE	74.03	52.06	30.5	19.5	74.10	53.56	26.1	18.4	70.45	53.93	51.3	23.7	71.48	56.38	51.0	23.9
VEH	62.45	56.30	93.1	45.1	66.42	57.94	94.6	46.1	65.24	59.07	104.3	44.0	68.38	63.62	100.7	43.2
WIN	98.44	90.25	26.2	23.4	99.30	90.33	23.5	21.6	98.90	90.83	34.7	27.2	99.45	92.27	33.7	26.4
WIS	98.39	95.81	22.4	19.5	98.68	95.48	22.8	19.7	98.62	95.62	36.9	21.2	98.70	96.13	36.1	21.7

Table 6 Results of the non-parametric statistical tests on the accuracy computed on the test set for the FIRST, MEDIAN and LAST solutions generated by PAES-R, PAES-RT, PAES-RG and PAES-RGT, employing intervals as information granules

FIRST

Algorithm	*Friedman rank*	*Iman and Davenport p-value*	*Hypothesis*
PAES-R	3.2917		
PAES-RT	2.875		
PAES-RG	2.0833	0.000016	Rejected
PAES-RGT	1.75		

Holm post-hoc procedure

i	*Algorithm*	*z-value*	*p-value*	*alpha/i*	*Hypothesis*
3	PAES-R	4.136726	0.000035	0.016	Rejected
2	PAES-RT	3.018692	0.002539	0.025	Rejected
1	PAES-RG	0.894427	0.371093	0.05	Not Rejected

MEDIAN

Algorithm	*Friedman rank*	*Iman and Davenport p-value*	*Hypothesis*
PAES-R	2.9167		
PAES-RT	3.00		
PAES-RG	2.1667	0.005587	Rejected
PAES-RGT	1.9167		

Holm post-hoc procedure

i	*Algorithm*	*z-value*	*p-value*	*alpha/i*	*Hypothesis*
3	PAES-T	2.906888	0.00365	0.016	Rejected
2	PAES-RT	2.683282	0.00729	0.025	Rejected
1	PAES-RG	0.67082	0.502335	0.05	Not Rejected

LAST

Algorithm	*Friedman rank*	*Iman and Davenport p-value*	*Hypothesis*
PAES-R	3.5417		
PAES-RT	2.8333		
PAES-RG	2.0417	0.000000005	Rejected
PAES-RGT	1.5833		

Holm post-hoc procedure

i	*Algorithm*	*z-value*	*p-value*	*alpha/i*	*Hypothesis*
3	PAES-R	5.25476	0	0.016	Rejected
2	PAES-RT	3.354102	0.000796	0.025	Rejected
1	PAES-RG	1.229837	0.218758	0.05	Not Rejected

ence the classification performance and the interpretability of the GRBCs. In particular, we have observed that the best results in terms of accuracy are obtained when granulation tuning and learning are used simultaneously. On the other hand, the GRBCs so generated are characterized by a higher complexity in terms of total rule length and number of rules. This is basically due to the use of a maximum number of granules for each variable higher than the number of granules used when only granulation tuning is used.

References

1. Lotfi A Zadeh. Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy sets and systems*, 90 (2):111–127, 1997.
2. Jing Tao Yao, Athanasios V Vasilakos, and Witold Pedrycz. Granular computing: perspectives and challenges. *Cybernetics, IEEE Transactions on*, 43(6):1977–1989, 2013.
3. Yiyu Yao. Interval sets and interval-set algebras. In *ICCI'09. 8th IEEE International Conference on Cognitive Informatics*, pages 307–314. IEEE, 2009.
4. Caihui Liu, Duoqian Miao, and Jin Qian. On multi-granulation covering rough sets. *International Journal of Approximate Reasoning*, 55(6):1404–1418, 2014.
5. Witold Pedrycz, Rami Al-Hmouz, Ali Morfeq, and Abdullah Saeed Balamash. Building granular fuzzy decision support systems. *Knowledge-Based Systems*, 58:3–10, 2014.
6. Witold Pedrycz and George Vukovich. Granular computing with shadowed sets. *International Journal of Intelligent Systems*, 17(2):173–197, 2002.
7. Witold Pedrycz. *Granular computing: analysis and design of intelligent systems*. CRC Press, 2013.
8. Witold Pedrycz, Giancarlo Succi, Alberto Sillitti, and Joana Iljazi. Data description: A general framework of information granules. *Knowledge-Based Systems*, 2015.
9. P. Ducange and F. Marcelloni. Multi-objective evolutionary fuzzy systems. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6857 LNAI: 83–90, 2011.
10. Michela Fazzolari, Rafael Alcalá, and Francisco Herrera. A multi-objective evolutionary method for learning granularities based on fuzzy discretization to improve the accuracy-complexity trade-off of fuzzy rule-based classification systems: D-mofarc algorithm. *Applied Soft Computing*, 24:470–481, 2014.
11. Maria José Gacto, Rafael Alcalá, and Francisco Herrera. Integration of an index to preserve the semantic interpretability in the multiobjective evolutionary rule selection and tuning of linguistic fuzzy systems. *IEEE Transactions on Fuzzy Systems*, 18(3):515–531, 2010.

12. Marco Cococcioni, Pietro Ducange, Beatrice Lazzerini, and Francesco Marcelloni. A Pareto-based multi-objective evolutionary approach to the identification of Mamdani fuzzy systems. *Soft Computing*, 11(11):1013–1031, 2007.
13. Alessio Botta, Beatrice Lazzerini, Francesco Marcelloni, and Dan C. Stefanescu. Context adaptation of fuzzy systems through a multi-objective evolutionary approach based on a novel interpretability index. *Soft Computing*, 13(5):437–449, 2009.
14. Pedro Villar, Alberto Fernandez, Ramon A. Carrasco, and Francisco Herrera. Feature selection and granularity learning in genetic fuzzy rule-based classification systems for highly imbalanced data-sets. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 20(03):369–397, 2012.
15. Michela Antonelli, Pietro Ducange, Beatrice Lazzerini, and Francesco Marcelloni. Multi-objective evolutionary learning of granularity, membership function parameters and rules of Mamdani fuzzy systems. *Evolutionary Intelligence*, 2(1-2):21–37, 2009.
16. Michela Antonelli, Pietro Ducange, Beatrice Lazzerini, and Francesco Marcelloni. Learning concurrently partition granularities and rule bases of mamdani fuzzy systems in a multi-objective evolutionary framework. *International Journal of Approximate Reasoning*, 50(7):1066–1080, 2009.
17. Ramon E Moore. *Interval analysis*, volume 4. Prentice-Hall Englewood Cliffs, 1966.
18. L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.
19. Hisao Ishibuchi, Tomoharu Nakashima, and Manabu Nii. *Classification and Modeling with Linguistic Information Granules: Advanced Approaches to Linguistic Data Mining (Advanced Information Processing)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2004.
20. Oscar Cordon, Maria Jose del Jesus, and Francisco Herrera. A proposal on reasoning methods in fuzzy rule-based classification systems. *International Journal of Approximate Reasoning*, 20(1):21–45, 1999.
21. S. Guillaume. Designing fuzzy inference systems from data: An interpretability-oriented review. *IEEE Transactions on Fuzzy Systems*, 9(3):426–443, jun 2001.
22. J.V. de Oliveira. Semantic constraints for membership function optimization. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 29(1):128–138, jan 1999.
23. C. Mencar and A.M. Fanelli. Interpretability constraints for fuzzy information granulation. *Information Sciences*, 178(24):4585 – 4618, 2008.
24. José María Alonso, L. Magdalena, and G. González-Rodríguez. Looking for a good fuzzy system interpretability index: An experimental approach. *International Journal of Approximate Reasoning*, 51(1):115–134, 2009.
25. ShangMing Zhou and John Q. Gan. Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modelling. *Fuzzy Sets and Systems*, 159(23):3091–3131, 2008.

26. M. J. Gacto, R. Alcalá, and F. Herrera. Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures. *Information Sciences*, 181(20):4340–4360, 2011.
27. G.A. Miller. The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, 63(2): 81–97, 1956. doi: 10.1037/h0043158.
28. W. Pedrycz and F. Gomide. *Fuzzy Systems Engineering: Toward Human-Centric Computing*. John Wiley & Sons, Inc., 2007. doi: 10.1002/9780470168967.
29. F. Klawonn. Reducing the number of parameters of a fuzzy system using scaling functions. *Soft Computing*, 10(9):749–756, 2006.
30. Michela Antonelli, Pietro Ducange, Beatrice Lazzerini, and Francesco Marcelloni. Learning concurrently data and rule bases of Mamdani fuzzy rule-based systems by exploiting a novel interpretability index. *Soft Comput.*, 15(10):1981–1998, 2011.
31. Michela Antonelli, Pietro Ducange, Beatrice Lazzerini, and Francesco Marcelloni. Learning knowledge bases of multi-objective evolutionary fuzzy systems by simultaneously optimizing accuracy, complexity and partition integrity. *Soft Computing*, 15(12):2335–2354, 2011.
32. M. Antonelli, P. Ducange, and F. Marcelloni. Genetic training instance selection in multi-objective evolutionary fuzzy systems: A co-evolutionary approach. *IEEE Trans. Fuzzy Syst.*, 20(2):276–290, 2012.
33. Joshua D Knowles and David W Corne. Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary computation*, 8(2):149–172, 2000.
34. J Alcalá-Fdez, Alberto Fernández, Julián Luengo, Joaquín Derrac, Salvador García, L Sánchez, and F Herrera. Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *Multiple-Valued Logic and Soft Computing*, 17(2-3):255–287, 2011.
35. Rafael Alcalá, Pietro Ducange, Francisco Herrera, Beatrice Lazzerini, and Francesco Marcelloni. A multiobjective evolutionary approach to concurrently learn rule and data bases of linguistic fuzzy-rule-based systems. *IEEE Transactions on Fuzzy Systems*, 17(5):1106–1122, 2009.
36. 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(200):675–701, 1937.
37. R. L. Iman and J. H. Davenport. Approximations of the critical region of the friedman statistic. *Communications in Statistics - Theory and Methods Part A*, 9:571–595, 1980.
38. S. Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6:65–70, 1979.