Information Sciences 136 (2001) 135–157
INFORMATION SCIENCES
AN INTERNATIONAL JOURNAL
www.elsevier.com/locate/ins

Genetic feature selection in a fuzzy rule-based classification system learning process for high-dimensional problems

J. Casillas [a], O. Cordón [a], M.J. Del Jesus [b,*], F. Herrera [a]

[a] *Department of Computer Science and Artificial Intelligence, University of Granada, 18071 Granada, Spain*
[b] *Department of Computer Science, University of Jaén, 23071 Jaén, Spain*

Abstract

The inductive learning of a fuzzy rule-based classification system (FRBCS) is made difficult by the presence of a large number of features that increases the dimensionality of the problem being solved. The difficulty comes from the exponential growth of the fuzzy rule search space with the increase in the number of features considered in the learning process. In this work, we present a genetic feature selection process that can be integrated in a multistage genetic learning method to obtain, in a more efficient way, FRBCSs composed of a set of comprehensible fuzzy rules with high-classification ability. The proposed process fixes, a priori, the number of selected features, and therefore, the size of the search space of candidate fuzzy rules. The experimentation carried out, using Sonar example base, shows a significant improvement on simplicity, precision and efficiency achieved by adding the proposed feature selection processes to the multistage genetic learning method or to other learning methods.

Keywords: Fuzzy rule-based classification systems; Inductive learning; Feature selection; Fuzzy reasoning methods

* Corresponding author.
E-mail addresses: casillas@decsai.ugr.es (J. Casillas), ocordon@decsai.ugr.es (O. Cordón), mjjesus@ujaen.es (M.J. Del Jesus), herrera@decsai.ugr.es (F. Herrera).

PII: S 0 0 2 0 - 0 2 5 5 (0 1) 0 0 1 4 7 - 5

1. Introduction

The inductive learning of a fuzzy rule-based classification system (FRBCS) starts from a set of instances or patterns, and determines a set of fuzzy rules and a fuzzy inference method that generalises the knowledge extracted from the data in order to classify new patterns. Each one of these patterns is described by a set of features, also called variables or characteristics.

An FRBCS learning process must solve different problems to obtain a linguistic FRBCS with an accurate behaviour, such as:
1. the obtaining of a fuzzy rule set with an adequate co-operation level between the rules,
2. the selection of the inference method, which determines the way of combining the information provided by the fuzzy rules in the classification of the examples.
3. the establishment and tuning – if necessary – of the fuzzy partitions for the linguistic variables, and
4. when managing high-dimensional problems, the fuzzy rule set suffers an exponential growth in its size when a large number of input variables are considered.

The first three problems, related to the knowledge extraction process, have been solved by different learning processes based on iterative methods [9,30,46], Neural Networks [33,44,45] or genetic algorithms (GAs) [21,29,31,58], among others.

The fourth problem can be tackled from a double perspective:
- Via the compactness and reduction of the rule set, minimising the number of fuzzy rules included in it. Unnecessary rules can be eliminated with the aim of having a more co-operative rule set in order to obtain an FRBCS with better performance.
- Via a feature selection process which reduces the number of features used by the FRBCS, thus reducing the fuzzy rule search space.

Rule reduction methods act combining rules and/or selecting a subset of them from a given rule set to achieve the goal of minimising the number of rules used while maintaining (or even improving) the FRBCS performance. The badly defined and conflicting rules are eliminated through the method since their existence degrades the system performance.

Rule reduction methods have been formulated using Neural Networks, clustering techniques and orthogonal transformation methods, similarity measures [10,23,34,35,48–50,55,57], as well as using GA-based rule selection processes to get a co-operative set of rules from a candidate rule set [15,16,24,26,32,37,47]. From a different point of view, in [11] an attempt to reduce the growth of the rule set by proposing a disjunctive form for the fuzzy rules (a rule combination method) is considered.

In [13], a multistage genetic learning process was presented, based on the MOGUL methodology [15], that deals with the four said problems. It learns fuzzy rules covering the example set, integrates the inference method together with a rule set genetic selection process, getting high co-operation rule sets, and applies a genetic tuning process to obtain the final membership function definitions.

We must remark that, for high-dimensional problems and problems where a high number of instances is available, it is difficult for the reduction and selection approaches to get small rule sets, and therefore the system comprehensibility and interpretability may not be as good as desired.

For high-dimensional classification problems, a feature selection process, that determines the most relevant variables before or during the FRBCS inductive learning process, can be considered. It reduces the fuzzy rule search space and increases the efficiency and accuracy of the learning and classification stages.

Our objective is to develop a feature selection process to be integrated, in a proper way, into the said multistage genetic learning process [13], to obtain FRBCSs composed of a set of comprehensible fuzzy rules with high-classification ability in a more efficient way. For this task, we devise a genetic feature selection process that fixes, a priori, the number of selected features and, therefore, the size of the search space of candidate fuzzy rules.

In this way, the extended multistage genetic learning process deals with the fourth said problem, high dimensionality, from a double point of view, getting a small number of features, and getting compact rule sets with an appropriate co-operation level via genetic selection integrating inference methods.

To carry out this task, this paper is organised as follows. In Section 2, some preliminaries are introduced: the FRBCS components and a brief description of the multistage genetic learning process for FRBCSs. In Section 3, the dimensionality problem in this learning process is presented. As a solution to this problem, Section 4 describes the integration of a feature selection process in the multistage FRBCS learning process. The proposals for the feature selection stage are explained in Section 5. Section 6 shows the results of the experiments with Sonar data base. In Section 7, some conclusions are pointed out.

2. Preliminaries

2.1. Fuzzy rule-based classification systems

An FRBCS is an automatic classification system that uses fuzzy rules as knowledge representation tool.

The FRBCS design implies two processes, which are graphically shown in Fig. 1:

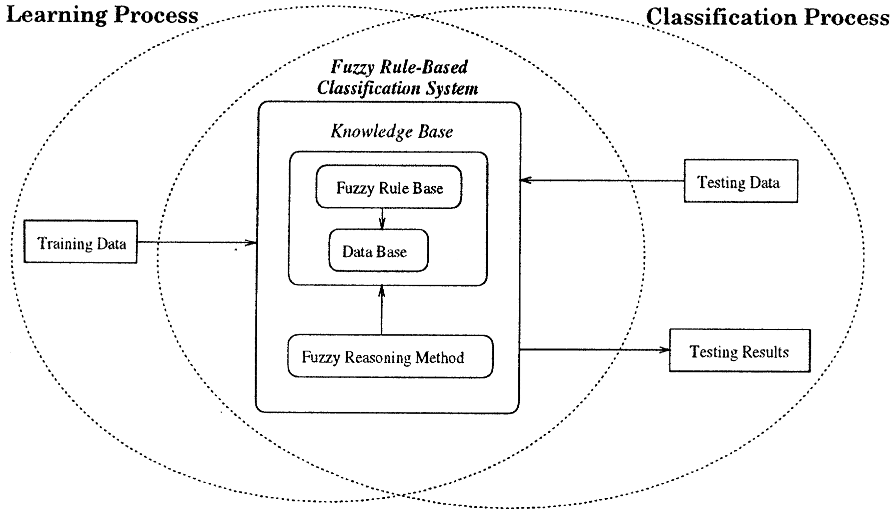

Fig. 1. Design of an FRBCS (learning/classification).

- The *inductive learning process* of the FRBCS, which starts from a set of problem instances with known class and obtains an automatic classification system that generalises the knowledge extracted from the data in order to classify new instances.
- The *classification process*, which uses the FRBCS to determine its classification capability with unknown problem instances.

As can also be seen in Fig. 1, in an FRBCS two different components are distinguished:

1. The *Knowledge base* (KB), composed of:
   - a *Data base* (DB), which contains the fuzzy set definitions related to the linguistic terms used in the fuzzy rules, and
   - a *Rule base* (RB), comprised by a set of fuzzy rules that in this work are considered to have the following structure:

   $$R_k: \text{ If } X_1 \text{ is } A_1^k \text{ and } \ldots \text{ and } X_N \text{ is } A_N^k \text{ then, } Y \text{ is } C_k \text{ with } r^k,$$

   where $X_1, \ldots, X_N$ are features considered in the problem and $A_1^k, \ldots, A_N^k$ are linguistic labels employed to represent the values of the variables.

   This kind of fuzzy rule represents, in the antecedent part, a subspace of the complete pattern space by means of a linguistic label for each considered variable and, in the consequent part, has a class label (C_k) and a certainty degree (r^k). This numerical value indicates the degree of certainty of the classification in that class for the examples belonging to the fuzzy subspace delimited by the antecedent part.

2. The *fuzzy reasoning method* (FRM), an inference procedure, which combines the information provided by the fuzzy rules related with the example to be classified, determines the class to which it belongs.

The majority of FRBCSs (see [1,9,21,30,43] among others) use the classical FRM that classifies a new example with the consequent of the fuzzy rule having the greatest degree of association. By using this reasoning method, the FRBCS loses the information provided by other rules with different linguistic labels which also represent this value in the pattern attribute, although probably to a lesser degree. Other FRMs that use the information provided by all the rules compatible with the example (or a subset of them) had been developed [4,9,14,28]. In [12,14], a general FRM framework for FRBCSs is described and different proposals are presented to improve the performance of an FRBCS in the classification stage.

In [13], a genetic learning process is presented, that considers the fuzzy rule co-operation and the FRM used in the classification stage to select fuzzy rules and to tune the fuzzy partitions. We must note that this multistage learning process does not tackle the need of reducing the problem dimensionality when the number of features is high. In the following section, we will briefly describe this learning process, in order to observe its main characteristics, to study its limitations in managing the problem dimensionality, and to analyse the integration of a feature selection stage.

2.2. Multistage genetic learning of fuzzy rule-based classification systems

The multistage genetic learning process for FRBCSs consists of three stages [13]:

1. *An iterative fuzzy rule generation process* that obtains a linguistic RB which represents the knowledge extracted from the training examples and verifies the completeness and k-consistency properties [20,24].
2. *A genetic multiselection process* that generates different KBs by the selection of different rule subsets and the learning of different linguistic modifier sets, considering the FRM used in the classification stage.
3. *A genetic tuning process* that leads to the best membership function definitions for the fuzzy rules.

2.2.1. Fuzzy rule generation process

The iterative fuzzy rule generation process has two components, a rule generating method and an iterative covering method:

- *The fuzzy rule generating method* obtains, in each iteration, a candidate fuzzy rule set, generating for each training example the fuzzy rule which represents the fuzzy subspace to which it belongs. From this set of rules, the best one is selected by means of a multicriteria selection function, which considers criteria related to the rule frequency, τ-completeness and k-consistency [15].

- *The covering method* applies the generation method to obtain the best rule for the training examples, and considers the relative covering that this rule causes in them, eliminating those examples that are covered to a degree higher than a maximum value previously specified.

These processes are applied repeatedly until the training set becomes empty.

It must be noted that in this generation process, the FRM is not considered for the obtaining of the RB.

2.2.2. Genetic multiselection process

The fuzzy rule generation process, which does not consider the relationship among the different rules, can obtain an RB with an inappropriate co-operation level. To solve this problem, one objective of the genetic multiselection process is the selection of fuzzy rule subsets with optimal co-operation in the classification stage depending on the FRM used. Besides this, the multiselection process allows the precision of the FRBCS to be increased by selecting a linguistic modifier set for the linguistic terms used in the RB.

This multimodal optimisation problem is solved with a GA [19,25] that uses the *sequential niche technique* [6] to induce niches in the search space and obtain different KB definitions by means of the basic genetic selection process.

As said, the *basic genetic selection process* has a double objective: the selection of a rule subset with a good co-operation among them, considering the FRM, and the selection of a linguistic modifier set related to the fuzzy subsets used by the fuzzy rules. The latter can be done in two different forms: selecting a linguistic modifier for each fuzzy subset defined in the DB, or determining a linguistic modifier for each fuzzy subset related to each linguistic variable in each fuzzy rule. In this genetic process, the fitness associated to each solution (KB definition) is based on two criteria, a global classification error measure and a criterion penalising the non-satisfaction of the τ-completeness property.

Each time this basic genetic selection process is executed, the solution obtained is optimised using a high-climbing process. Finally, the search space zone in which the solution has been obtained is penalised, to get different KBs in posterior executions of the basic genetic selection process.

We should note that the basic rule selection genetic process could be extended in different ways. For example:

- Another criterion related to the complexity of the RB to be selected can be also considered in the optimisation process. This criterion is usually based on minimising the number of rules composing the encoded RB and can be incorporated to the GA in two different forms. It can be directly included in the fitness function, as done in [24] where both criteria are aggregated into a single measure by means of a weighted combination, or a more sophisticated GA structure can be considered to obtain a multiobjective GA

optimising both criteria at the same time by generating different solutions in the pareto set [26].
- Since the problem of selecting rules is difficult and the search space is strongly multimodal, a more sophisticated *niching GA* [19] can be considered (see [17]).

For the sake of simplicity, none of these extensions will be considered for our rule selection process in this paper. Although better results could be obtained in other case, we prefer not to include additional experimentations related to this point. We think that the criterion considered, the classification error over a data set, will be enough to obtain good results due to the fact that it is directly related to the accuracy of the linguistic model, which is significantly affected by an excessive number of rules.

2.2.3. Genetic tuning process

The genetic tuning process leads to optimise the fuzzy partition of the linguistic variables, determining the best membership function parameter values in a common way to all the fuzzy rules.

This process uses a real coded GA based on the parametric representation of the membership functions. It demands, as the multiselection process does, the verification of the τ-completeness property.

3. The dimensionality problem in the multistage genetic learning process

As said, the design of an FRBCS for a classification problem with a large number of features entails efficiency and/or effectiveness problems, if the learning algorithm searches in the complete search space as the multistage genetic learning process does. These problems come from the exponential growth of the fuzzy rule search space with the increase in the number of features considered in the learning process.

In order to show, in an empirical way, the effects of the dimensionality problem in the multistage genetic learning process, we have applied it to an example base with a high-feature number, Sonar data set [5,22], which has 208 instances of a sonar objective classification problem. Each one of these instances is described by 60 features to discriminate between a sonar output corresponding to a cylindrical metal or an approximately cylindrical rock.

If we use five linguistic labels per variable to solve this problem, the search space for the learning process is composed of 5^{60} candidate fuzzy rules. The results obtained after the generation stage are shown in Table 1 (results with different FRMs that are described in [12,14]). In brackets, we expose the values for the FRMs parameters.

In this table, we can observe that the correct classification percentage with five linguistic labels per variable is the same independently of the FRM used.

Table 1
Results with a KB obtained after the generation process for the Sonar problem

FRM	5 Labels		3 Labels	
	311 Rules		331 Rules	
	Tra.	Test	Tra.	Test
Classical	100	43.27	99.04	75.00
Normalised sum	100	43.27	98.08	73.08
Arithmetic mean	100	43.27	96.15	72.11
Quasiarithmetic mean ($p = 10$)	100	43.27	99.04	75.00
SOWA or-like ($\alpha = 0.3$)	100	43.27	98.08	76.92
Badd ($p = 10$)	100	43.27	99.04	75.00
OWA ($a = 0$, $b = 0.3$)	100	43.27	98.08	75.96
QuasiOWA ($a = 0$, $b = 0.3$, $p = 10$)	100	43.27	99.04	75.96

This is because the wrong classified examples do not belong to any fuzzy rule and the FRBCS cannot classify them. This problem can be lessened with the use of a more compensated t-norm than the one used, the minimum, or considering fuzzy partitions with a smaller number of linguistic labels. In Table 1, columns 4 and 5, we can see the results obtained by the generation process considering only 3 linguistic labels per linguistic variable.

Nevertheless, the results obtained in the first stage of the learning process, in both situations, show that the use of the complete set of features leads to the design of an FRBCS that is overfitted to the training examples and covers only a small proportion of the complete example space.

Since the proposed learning process does not select the relevant features and uses all the available ones, when the feature number is large, the fuzzy rule search space becomes huge, the learning process turns slower and the FRBCS finally obtained is overfitted to the training examples. This fact limits the chances of improving the classification system for the postprocessing (multi-selection and tuning) stages.

4. The integration of a feature selection stage in the multistage genetic learning process

The integration of a feature selection stage in the multistage genetic learning process will reduce the problem dimensionality before the fuzzy rule generation stage.

In short, the feature selection stage is executed as follows: first, a genetic feature selection process determines a set of feature subsets by means of the chromosomes in the final population with the best fitness value. Since these feature subsets have been selected without considering the heuristics of the inductive learning method, we use the efficient fuzzy rule generation process as an intermediate stage to determine the best feature subset for the problem being solved. In this form, we include the heuristic and bias of the learning process in an efficient way to obtain a feature subset, which properly works with the fuzzy rules and the FRM used to classify new examples. Finally, we execute the postprocessing stages of the multistage genetic learning method (that is to say, multiselection and tuning) over the RB obtained with the selected feature subset.

The resulting FRBCS learning process consists of the following stages:

1. A *genetic feature selection process*, which gets a feature subset to learn the FRBCS from it. The proposed feature selection process uses a GA as search algorithm and it has wrapper nature [36]. We also use a feature selection algorithm with filter nature [39–41] that searches for a variable cardinality feature subset to obtain the chromosome length for our proposal of genetic feature selection process. We will explain this process in detail in the following section.
2. An *iterative fuzzy rule generation process*, which – using only the selected features – obtains an RB independently of the FRM used in the classification stage.

   We use this efficient fuzzy rule generation process as an intermediate stage to determine the best feature subset (consequently, the best KB) and the best FRM for the problem to solve.
3. A *genetic multiselection process* of different KBs with a good co-operation level among the fuzzy rules, considering the FRM selected in the previous step.
4. A *genetic tuning process*, which modifies the fuzzy set definitions in a common way for all rules to obtain a linguistic FRBCS.

The resulting FRBCS learning process is graphically described in Fig. 2. We can see how the genetic feature selection process starts from the complete set of features and provides us some feature subsets with a fixed cardinality (corresponding to different chromosomes with the maximum value for the fitness function in the final population). With each one of these variable subsets, we run the generation process to obtain different RBs regardless of the FRM used in the classification process. The prediction capability of these RBs is measured with different FRMs to choose a number of the FRBCSs with the best behaviour (in this paper, we will work with the best two ones). As said, we must note that, at this point, the feature selection stage finishes adding some of the heuristics and bias of the inductive learning method to the selection stage. These FRBCSs are improved by means of the postprocessing stages (that is to say, multiselection and tuning stages).

In Section 5, we will explain in depth our proposals for the feature selection stage.

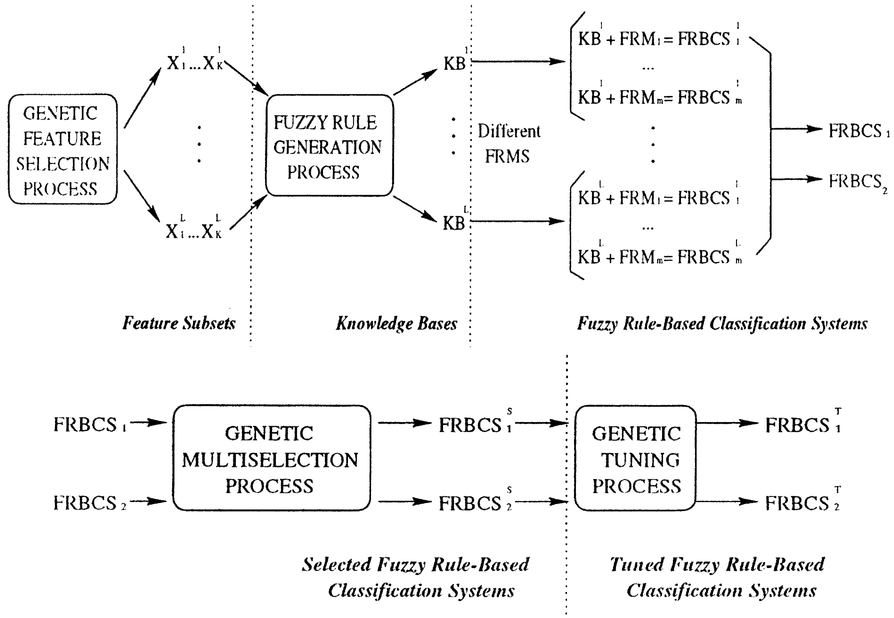

Fig. 2. Stages in the FRBCS learning process.

5. Feature selection process

5.1. Feature selection approaches

The main objective of the genetic feature selection stage is to reduce the dimensionality of the problem before the supervised inductive learning process. This fact implies that the feature selection algorithm must determine – without the necessity of the FRBCS construction – the best features for its design.

There are two kinds of feature selection algorithms:

- *Filter feature selection algorithms* [39], which remove the irrelevant characteristics without using a learning algorithm. They are efficient processes but, on the other hand, the feature subsets obtained by them may not be the best ones for a specific learning process because of the exclusion of the heuristic and bias of the learning process in the selection procedure [36].
- *Wrapper feature selection algorithms* [36,39]. This kind of feature selection algorithms selects feature subsets by means of the evaluation of each candidate subset with the precision estimation obtained by the learning algorithm. In this form, they obtain feature subsets with the best behaviour in the FRBCS design. Their problem is their inefficiency since they must build the FRBCS for each evaluation of a candidate feature subset.

5.2. Genetic feature selection proposal

We propose a genetic feature selection stage that combines both kinds of feature selection algorithms in two steps:

1. The first step looks for a feature subset with variable size considering class separability measures to determine an optimal feature number for a specific classification problem. In this study, we employ the filter algorithm Las Vegas filter (LVF) [39–41], which is based on the inconsistency rate, proposed by Liu and Setiono (which is described in Appendix A). This filter feature selection method provides us a cardinality for the feature subset with minimum number of inconsistencies.

   Besides this cardinality, we use the feature subset size given by an expert since, for some classification problems, it is necessary to enforce a fixed reduction of the feature subset cardinality before the learning process.
2. The results of the previous feature selection algorithm and the expert opinion provide us an adequate feature subset cardinality used as chromosome length for a wrapper genetic feature selection process that determines a feature subset from which an accurate FRBCS can be obtained. To increase the efficiency while maintaining the effectiveness of the wrapper feature selection algorithms, our proposal uses the precision estimation provided by the k-nearest neighbour rule (k-NN) [18,38], which is very sensitive to the resence of irrelevant characteristics.

Before describing the components of the genetic feature selection process, four considerations on the process must be pointed out:

- The feature selection process follows this outline because it is necessarily an efficient feature selection process, which provides an adequate variable subset for the FRBCS design.
- A wrapper feature selection process that uses the k-NN rule in each feature subset evaluation, is an efficient approach because this classification rule does not have learning time and it is very sensitive to irrelevant variables.
- If the feature selection process would have only the wrapper selection step, the resulting variable subset may not have the minimum cardinality since the k-NN rule is not sensitive to redundant characteristics.
- The determination of the feature subset size carried out by the filter algorithm or the expert helps the wrapper feature selection process to select only relevant variables and to, effectively and efficiently, reduce the complexity of the classification problem.

The feature selection problem is an optimisation problem with restrictions that has been solved by means of GAs in different proposals ([3,8,27,52,56]). In the following section, we describe our genetic proposal for the wrapper feature selection process.

5.3. Steady-state GAs for feature selection

The first stage of the learning process is carried out by means of a feature selection algorithm based on a GA with a variant of the pure steady-state reproduction mechanism [53]. As said, this feature selection process is a wrapper feature selection algorithm that uses a precision measure provided by the k-NN considering only the features included in the candidate feature subset as evaluation function.

The GA is described by its components:

1. *Coding scheme*. The feature selection process objective is to get an optimal feature subset with a fixed cardinality, so the integer coding using fixed length allows us to represent a candidate subset containing H variables in a chromosome of length H in which, the ith gen represents the ith selected variable.

   The proposed algorithm permits the incorporation of the available knowledge, that is to say, feature subsets provided by an expert or another feature selection algorithm, in the initial population. The remainder of that population is randomly generated.
2. *Fitness function*. To increase the speed of the feature selection stage, the estimation of the attainable precision is calculated by the k-NN rule. This test precision estimation is obtained by the training random resampling technique for wrapper feature selection algorithms [36]. It uses five training-test partitions obtained from the original training set, and calculates the adaptation measure with the arithmetic mean of the five test correct classification results. In this way, we can estimate the generalisation capability of a feature subset without using the test set employed to validate the finally obtained feature subset.
3. *Reproduction scheme*. The proposed GA uses a variant of the steady-state reproduction scheme that does not substitute one or two chromosomes from the population in each generation, but a fixed and larger number of them. We propose a reproduction scheme that has the following steps:
   - To generate an intermediate population by means of linear ranking and the universal stochastical sampling [2].
   - To apply the crossover and mutation operators to some individuals from this intermediate population. The number of chromosomes to be created will be determined by the crossover and mutation probabilities.
   - To substitute the worst adapted chromosomes from the *original population* by the new ones so created.

The generation of more than two new chromosomes allows to have more diversity in the new population than the pure steady-state reproduction scheme. Nevertheless, it maintains the steady-state characteristics because the new population only differs from the previous one on these generated chromosomes, which substitute the worst adapted ones.

4. *Crossover operator*. We propose two different crossover operators that allow us to consider two GAs that only differ in this operator:
   - The first one is the *partially complementary crossover operator* [42]. (The associated GA is identified by SSGA_I).

     This operator exploits the search space in the following way: given two chromosomes from the population, $P(t)$, $C_v^t = (c_1, \ldots, c_M)$ and $C_w^t = (c'_1, \ldots, c'_M)$, it generates two descendants H_1 and H_2 with the genes common to both parents and the rest selected randomly among the remaining genes of the parents:

     $$H_1 = (d_1, \ldots, d_k, h_{k+1}, \ldots, h_M),$$
     $$H_2 = (d_1, \ldots, d_k, h'_{k+1}, \ldots, h'_M),$$

     where $d_1, \ldots, d_k$ are the common genes to the two chromosomes selected to be crossed, and $h_{k+1}, \ldots, h_M$, and $h'_{k+1}, \ldots, h'_M$ are genes randomly selected among the remainder.

     In this way, the descendants maintain the parents' common variables and randomly combine the remaining information. They are valid solutions and do not need any repairing algorithm.
   - The second one is the *two-point crossover with repair* operator. (The associated GA is identified by SSGA_II).

     An analysis of the SSGA_I lets us note that it can evolve to a population without enough diversity. To solve this problem, we propose the second algorithm, SSGA_II, based on the use of the two-point crossover operator with repair, which not only exploits the information given by the parents but also introduces diversity in the descendants. This operator works as follows: for a pair of chromosomes selected to be crossed, it randomly determines two cross points and interchanges the genes between them. This process can generate non-valid individuals because of the variable repetition. To solve this problem, whenever needed, it executes a repairing algorithm which substitutes each repeated gene (in a non-valid chromosome) with a non-selected variable.

     This two-point crossover operator has the usual inheritance and refinement properties of the crossover operators and also – when the descendants have repeated variables – the exploration property which is very suitable in this evolutionary process.
5. *Mutation operator*. With regard to the mutation operator, the *uniform mutation* is used, which arbitrarily modifies one or more genes from an individual, removing the corresponding variable, and substituting it for another one which is not present in the chromosome, thus introducing diversity among the population.

6. Experimentation and result analysis

To describe the behaviour of the feature selection proposals in the multistage learning process and regardless of it, we will organise the experimentation with Sonar data base in the following way:

1. Firstly, we will show the results obtained by means of the complete multistage learning process.
2. Then, we will describe the performance of the FRBCSs designed by the generation, multiselection and tuning processes with other feature selection methods.
3. Finally, to analyse the capability of the feature selection processes regardless of the multistage genetic learning method, we will show the results obtained using another learning algorithm starting with the feature subsets selected by our genetic feature selection processes.

6.1. Results obtained using the complete learning process

As said, the feature selection process selects variable subsets with a previously determined fixed cardinality computed by executing a filter feature selection algorithm that searches for an optimal and minimum feature set, LVF [39–41]. This algorithm provided us a proper feature set size, 15 variables.

In [5], the use of 6 and 12 variables is considered. We also consider 3 variables to get a more descriptive FRBCS. In this form, we use four feature set sizes (3, 6, 12 and 15), which reduce the fuzzy rule search space for the fuzzy rule generation, selection and tuning processes by a 95%, 90%, 80% and 75%, respectively.

According to the previously exposed multistage learning process, we execute the feature selection algorithms SSGA_I and SSGA_II with these different sizes. We build FRBCSs with the generation process – starting with the feature subsets obtained before – and analyse the results to determine the two variable subsets (and consequently two KBs) and FRMs with the best performance for each cardinality. In this way, and efficiently due to the iterative nature of the generation process, we significantly reduce the problem dimensionality by limiting the fuzzy rule and FRM space to be considered in following stages.

Finally, we execute the multiselection and tuning processes to get FRBCSs that reach the results showed in Table 2. In this table, the first column indicates the feature selection algorithm; the second, the feature subset cardinality; the third, the FRM selected for the FRBCS; and the last three columns stand for the number of rules and correct classification results with training and test examples. The parameters of the complete learning process are described in Table 3.

Table 2
Results for an FRBCS built using 3, 6, 12 and 15 features

Algorithm	NF	FRM	NR	Tra.	Test
SSGA_I	3	Arithmetic mean	47	82.69	82.69
SSGA_II	3	Normalised sum	23	78.85	83.65
SSGA_I	6	OWA ($a = 0$, $b = 0.3$)	55	94.23	89.42
SSGA_II	6	Arithmetic mean + weighting	58	93.27	90.38
SSGA_I	12	SOWA or-like	183	92.31	94.23
SSGA_II	12	OWA ($a = 0$, $b = 0.5$)	45	91.35	90.38
SSGA_I	15	Classical	125	96.15	94.23
SSGA_II	15	OWA($a = 0$, $b = 0.5$) + weighting	94	99.04	92.31

Table 3
Genetic multistage learning process parameters

Stage	Parameter	Value
Feature selection	Chromosomes	61
	Generations	500
	Crossover probability	0.4
	Mutation probability	0.1
	Neighbours (K in k-NN rule)	1
Generation	Linguistic labels per variable	3
	Covering degree for positive examples (ω)	0.05
	Percentage of negative examples (k-consistency)	0.1
	Maximum covering degree (ϵ)	1.5
Multiselection	Minimum covering degree for the KB (τ-completeness)	0.1
	α (sequential niche technique)	0.5
	Niche radius (sequential niche technique)	0.025
	Solutions to generate	3
Multiselection and tuning	Linguistic labels per variable	3
	Chromosomes	61
	Generations	500
	Crossover probability	0.6
	Mutation probability	0.1

If we compare these results with the best test classification percentage obtained using the complete feature set in Table 1 (76.92% with 331 fuzzy rules and the FRM SOWA OR-Like), we can point out that, with the complete learning process we have increased the correct test classification percentages more than a 15%, and we have overcome the overfitting and efficiency problems, obtaining a simpler and more interpretable FRBCS.

Table 4
Results for an FRBCS built using feature subsets provided by MIFS and LVF algorithms

Algorithm	NF	FRM	NR	Tra.	Test
MIFS	3	Quasiarithmetic mean ($p = 20$)	9	69.23	78.85
MIFS	6	Normalised sum + weighting	27	77.88	76.92
MIFS	12	Classical	10	74.04	79.81
LVF	15	Normalised sum + weighting	218	98.07	81.73

6.2. Results obtained using other feature selection algorithms

To observe the behaviour of the generation, multiselection and tuning processes with other feature selection algorithms, we have executed these processes with different feature subsets composed of

- 15 variables provided by the LVF algorithm, and
- 3, 6 and 12 variables, selected by a filter greedy algorithm (MIFS) based on a forward selection search using the mutual information measure, developed by Battiti [5] (and described in Appendix B).

The best results are shown in Table 4.

The selection of a feature subset with the SSGA_I and SSGA_II feature selection processes leads to better results in the FRBCS multistage learning process although they do not use the attainable precision estimation employed in the learning process (but the k-NN rule). We can see this in Table 4, where the correct test classification percentages are worse than the results obtained using the feature selection algorithms SSGA_I and SSGA_II (Table 2).

We must point out that the use of class separability measures like mutual information in the feature selection process may lead to feature subsets from which simpler FRBCSs can be built. The LVF algorithm, which is based on the inconsistency rate, obtains feature subsets from which we can design a more accurate FRBCS but with a greater set of fuzzy rules.

6.3. Results obtained using another learning algorithm

The feature selection stage that we have proposed can be used as a preprocessing stage in other learning algorithms. In this subsection, we show the results obtained by the extension for classification problems of the Wang and Mendel's fuzzy rule generation process [9,54] (described in Appendix C). In Table 5, we can see the best results without feature selection stage – that is, considering the 60 features – and with feature subsets of 3, 6, 12 and 15 variables selected with the SSGA_I and SSGA_II algorithms.

The use of SSGA_I and SSGA_II algorithms allows to obtain, using the Wang and Mendel learning method, an FRBCS with a fewer number of fuzzy rules than the FRBCS built with the complete set of features, and with a

Table 5
Results for a Wang and Mendel FRBCS built using feature subsets provided by SSGA_I and SSGA_II algorithms

Algorithm	NF	FRM	NR	Tra.	Test
None	60	Arithmetic mean + weighting	104	97.11	77.88
SSGA_I	3	Arithmetic mean	38	72.11	79.81
SSGA_II	3	Normalised sum	18	67.31	81.73
SSGA_I	6	OWA ($a = 0$, $b = 0.5$)	66	75.00	82.69
SSGA_II	6	Badd ($p = 10$)	87	91.35	82.69
SSGA_I	12	OWA ($a = 0$, $b = 0.3$) + weighting	98	81.73	87.50
SSGA_II	12	SOWA or-like ($\alpha = 0.3$)	93	86.54	85.78
SSGA_I	15	OWA ($a = 0$, $b = 0.3$) + weighting	99	90.38	89.42
SSGA_II	15	Arithmetic mean + weighting	100	93.27	87.50

greater prediction capability. This allows us to show that the selection ability of the feature selection processes does not depend on the learning process although they have wrapper nature. They can be used in combination with other learning algorithms.

7. Conclusions

Usually, the following problems must be individually or jointly solved in the FRBCS design:

- The selection of the most relevant features for the considered classification problem.
- The fuzzy partition definitions for the linguistic variables.
- The generation of an RB that represents the sample information and verifies two desired properties in any RB, the completeness and consistency.
- The generation of an RB with a good co-operation level among the fuzzy rules with respect to the FRM used in the classification stage.

By way of conclusions, we can stress that the extended multistage genetic learning process of FRBCSs considers the said problems in different stages obtaining an FRBCS with the following characteristics:

- with a linguistic nature,
- with a good generalisation level,
- using only the most informative features for the classification problem,
- with an RB which verifies the completeness and k-consistency properties,
- with fuzzy rules with a suitable co-operation level depending on the FRM, and
- with an optimised definitions of the fuzzy sets.

The extended multistage genetic learning process deals with the high-dimensionality problem, from a double point of view, getting a small number of features, and getting compact rule sets with an appropriate co-operation level via genetic selection integrating inference methods.

The inclusion of a feature selection stage in the multistage genetic learning process effectively and efficiently reduces the complexity of the FRBCS design process due to the reduction of the fuzzy rule space before the post-processing stages. The combination of the feature selection process with the fuzzy rule generation and the FRM selection processes allows to determine a set of variables which consider some characteristics of the fuzzy rules, the learning method and the FRM used. The use of the attainable precision estimation provided by the k-NN rule makes more efficient the wrapper feature selection process.

The proposals for the feature selection stage can be used with other learning algorithms to obtain a feature subset that allows to the increase of the classification system accuracy, simplicity and linguistic description, and to the reduction of the learning efforts.

Acknowledgements

This research has been supported by CICYT under project PB98-1319.

Appendix A. Las Vegas filter feature selection algorithm

In [39–41], Liu and Setiono describe the algorithm LVF. It is a feature selection algorithm based on Las Vegas probabilistic search [7] that minimises the inconsistency rate introduced by the candidate feature subset to obtain a feature subset with a high-discrimination capacity.

The algorithm starts with a random feature subset. If it has a number of features smaller than the best feature subset found (the complete feature set at the beginning), the process calculates if the inconsistency criterion is achieved. In that case, the best feature subset is updated. The process continues during a number of iterations depending on the total feature number of the problem.

The inconsistency rate of a feature subset is calculated as follows:

1. To calculate the inconsistent instances, considering that two problem instances are inconsistent if they have the same values in the selected features and they belong to different classes. We must note that it is necessary to discretise continuous features by assigning intervals.
2. To obtain the inconsistency count for each different set of values for the feature subset, as the number of inconsistent instances minus the number of inconsistent instances which has maximum value among the classes.

3. To calculate the inconsistency rate as the sum of inconsistency counts divided by the total number of problem instances.

The discriminating power is reversibly proportional to the inconsistency rate. This is the reason why the inconsistency criterion is the main aspect of the algorithm: it specifies when a dimensionality reduction is acceptable.

Appendix B. Greedy filter feature selection algorithm

In [5], Battiti describes a filter feature selection algorithm called MIFS (Mutual Information based Feature Selection) based on a forward greedy search using the mutual information measure [51] with regard to the class. This algorithm selects the most informative feature about the class which cannot be predicted with the already selected features.

The algorithm starts with an empty feature subset and chooses the next feature as the one that maximises the information about the class with a penalisation corresponding to a quantity proportional to the average mutual information about the class without being predictable from the current feature subset. That is to say, in order to be selected, a feature must be informative about the class without being predictable from the already selected features. The process works in this way until the cardinality of the feature subset reaches a fixed value.

Appendix C. Wang and Mendel's generating method

The extension of Wang and Mendel's fuzzy rule generation algorithm [54] to classification problems [9] begin with a set of input–output data pairs (the training data set) with the following structure:

$$\begin{aligned} E^1 &= (e_1^1, \ldots, e_N^1, C^1), \\ E^2 &= (e_1^2, \ldots, e_N^2, C^2), \\ &\vdots \\ E^p &= (e_1^p, \ldots, e_N^p, C^p), \end{aligned}$$

where C^h is the class label for the pattern E^h.

Its objective is to generate a set of fuzzy rules from the training data set that describes the relationship between the system variables and determines a mapping D between the feature space S^N and the class set $C = \{C_1, \ldots, C_M\}$.

The method consists of the following steps:

- *Fuzzifying the feature space.* Finding the domain intervals of the features and partition each domain into X_i regions ($i = 1, \ldots, N$). A membership function is adopted for each fuzzy region. In our experiments we use membership functions with triangular shapes.

- *Generating fuzzy rules from given data pairs.* For each training data $E^h = (e_1^h, \ldots, e_N^h, C^h)$, we have
  - To determine the membership degrees of e_i^h in different input fuzzy subsets.
  - To assign the input $e_1^h, \ldots, e_N^h$ to the region with the maximum membership degree.
  - To produce a fuzzy rule from E^h, with the if-part that represents the selected fuzzy region and the consequent with the class determined by C^h and a certainty degree r^h. This numerical value is calculated by this expression MDS/MDS_h, where MDS is the sum of membership degrees for every example to the fuzzy region determined by the antecedent part, and MDS_h is the same sum but only for the examples of class C_h.

    We must note this method does not repeat the fuzzy rules.

References

[1] S. Abe, R. Thawonmas, A fuzzy classifier with ellipsoidal regions, IEEE Trans. Fuzzy Systems 5 (3) (1997) 358–368.

[2] J.E. Baker, Reducing bias and inefficiency in the selection algorithms, in: Proceedings of the Second International Conference on Genetic Algorithms (ICGA'87), Hillsdale, 1987, pp. 14–21.

[3] J. Bala, K. de Jong, J. Huang, H. Vafaie, H. Wechsler, Using learning to facilitate the evolution of features for recognizing visual concepts, Evol. Comput. 4 (3) (1997) 297–311.

[4] A. Bardóssy, L. Duckstein, Modeling with Applications to Geophysical, Biological and Engineering Systems, Systems Engineering Series, CRC Press, Boca Raton, 1995.

[5] R. Battiti, Using mutual information for selecting features in supervised neural net learning, IEEE Trans. Neural Networks 5 (4) (1994) 537–550.

[6] D. Beasly, D.R. Bull, R.R. Martin, A sequential niche technique for multimodal function optimization, Evol. Comput. 1 (1993) 101–125.

[7] G. Brassard, P. Bratley, Fundamentals of Algorithms, Prentice-Hall, Englewood Cliffs, NJ, 1996.

[8] F.Z. Brill, D.E. Brown, W.N. Martin, Fast genetic selection of features for neural network classifiers, IEEE Trans. Neural Networks 3 (2) (1992) 324–328.

[9] Z. Chi, H. Yan, T. Pham, Fuzzy Algorithms with Applications to Image Processing and Pattern Recognition, World Scientific, Singapore, 1996.

[10] S. Chiu, Fuzzy model identification based on cluster estimation, J. Intelligent Fuzzy Systems 2 (1994) 267–278.

[11] W.E. Combs, J.E. Andrews, Combinatorial rule explosion eliminated by a fuzzy rule configuration, IEEE Trans. Fuzzy Systems 6 (1) (1998) 1–11.

[12] O. Cordón, M.J. del Jesus, F. Herrera, Analyzing the reasoning mechanisms in fuzzy rule based classification systems, Mathware and Soft Comput. 5 (2&3) (1998) 321–332.

[13] O. Cordón, M.J. del Jesus, F. Herrera, Genetic learning of fuzzy rule-based classification systems co-operating with fuzzy reasoning methods, Int. J. Intelligent Systems 13 (10/11) (1998) 1025–1053.

[14] O. Cordón, M.J. del Jesus, F. Herrera, A proposal on reasoning methods in fuzzy rule-based classification systems, Int. J. Approx. Reasoning 20 (1) (1999) 21–45.

[15] O. Cordón, M.J. del Jesus, F. Herrera, M. Lozano, MOGUL: a methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning approach, Int. J. Intelligent Systems 14 (11) (1999) 1123–1143.
[16] O. Cordón, F. Herrera, A three-stage evolutionary process for learning descriptive and approximative fuzzy logic controller knowledge bases from examples, Int. J. Approx. Reasoning 17 (4) (1997) 369–407.
[17] O. Cordón, F. Herrera, Hybridizing genetic algorithms with sharing scheme and evolution strategies for designing approximate fuzzy rule-based systems, Fuzzy Sets and Systems 118 (2) (2001) 235–255.
[18] B.V. Dasarathy, Nearest Neighbour (NN) Norms: Nn Pattern Classification Techniques, IEEE Computer Society Press, Silver Spring, 1990.
[19] D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989.
[20] A. González, R. Pérez, Completeness and consistency conditions for learning fuzzy rules, Fuzzy Sets and Systems 96 (1998) 37–51.
[21] A. González, R. Pérez, SLAVE: a genetic learning system based on an iterative approach, IEEE Trans. Fuzzy Systems 7 (2) (1999) 176–191.
[22] R.P. Gorman, T.J. Sejnowski, Analysis of hidden units in a layered network trained to classify sonar targets, Neural Networks 1 (1988) 75–89.
[23] S. Halgamuge, M. Glesner, Neural networks in designing fuzzy systems for real world applications, Fuzzy Sets and Systems 65 (1) (1994) 1–12.
[24] F. Herrera, M. Lozano, J.L. Verdegay, A learning process for fuzzy control rules using genetic algorithms, Fuzzy Sets and Systems 100 (1998) 143–158.
[25] J.H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, MI, 1975.
[26] H. Ishibuchi, T. Murata, I.B. Turksen, Single-objective and two-objective genetic algorithms for selecting linguistic rules for pattern classification problems, Fuzzy Sets and Systems 89 (1997) 135–150.
[27] H. Ishibuchi, T. Nakashima, Multi-objective pattern and feature selection by a genetic algorithm, in: Proceedings of the Genetic and Evolutionary Computation Conference, 2000, pp. 1069–1076.
[28] H. Ishibuchi, T. Nakashima, T. Morisawa, Voting in fuzzy rule-based systems for pattern classification problems, Fuzzy Sets and Systems 103 (1999) 223–238.
[29] H. Ishibuchi, T. Nakashima, T. Murata, A fuzzy classifier system that generates fuzzy if–then rules for pattern classification problems, in: Proceedings of the Second International IEEE Conference on Fuzzy Systems, 1995, pp. 759–764.
[30] H. Ishibuchi, K. Nozaki, H. Tanaka, Distributed representation of fuzzy rules and its application to pattern classification, Fuzzy Sets and Systems 52 (1992) 21–32.
[31] H. Ishibuchi, K. Nozaki, H. Tanaka, Construction of fuzzy classification systems with rectangular fuzzy rules using genetic algorithms, Fuzzy Sets and Systems 65 (1994) 237–253.
[32] H. Ishibuchi, K. Nozaki, N. Yamamoto, H. Tanaka, Selecting fuzzy if-then rules for classification problems using genetic algorithms, IEEE Trans. Fuzzy Systems 3 (3) (1995) 260–270.
[33] J.-S.R. Jang, Anfis: adaptive-network-based fuzzy inference system, IEEE Trans. Syst. Man Cybern. 23 (3) (1993) 665–685.
[34] Y. Jin, Fuzzy modeling of high-dimensional systems: Complexity reduction and interpretability improvement, IEEE Trans. Fuzzy Systems 8 (2) (2000) 212–220.
[35] Y. Jin, W. vonSeelen, B. Sendhoff, On generating FC^3 fuzzy rule systems from data using evolutionary strategies, IEEE Trans. Syst. Man Cybern. – Part B: Cybernetics 29 (6) (1999) 829–845.

[36] R. Kohavi, G.H. John, Wrappers for feature subset selection, Artificial Intelligence 97 (1997) 273–324.
[37] A. Krone, P. Krause, T. Slawinski, A new rule reduction method for finding interpretable and small rule bases in high dimensional search spaces, in: Proceedings of the Ninth IEEE International Conference on Fuzzy Systems), vol. 2, San Antonio, Texas, May 2000, pp. 694–699.
[38] L.I. Kuncheva, L.C. Jain, Nearest neighbour classifier: Simultaneous editing and feature selection, Pattern Recognition Letters 20 (1999) 1149–1156.
[39] H. Liu, H. Motoda, Feature Selection for Knowledge Discovery and Data Mining, Kluwer Academic Publishers, Dordrecht, 1998.
[40] H. Liu, R. Setiono, A probabilistic approach to feature selection. A filter solution, in: L. Saitta (Ed.), Proceedings of International Conference on Machine Learning (ICML-96), Morgan Kaufmann, Los Altos, CA, 1996, pp. 319–327.
[41] H. Liu, R. Setiono, Incremental feature selection, Appl. Intelligence 9 (1998) 217–230.
[42] W. Liu, M. Wang, Y. Zhong, Selecting features with genetic algorithm in handwritten digits recognition, in: Proceedings of the International IEEE Conference on Evolutionary Computation (ICEC'95), vol. 1, 1995, pp. 396–399.
[43] D.P. Mandal, C.A. Murthy, S.K. Pal, Formulation of a multivalued recognition system, IEEE Trans. Syst. Man Cybern. 22 (4) (1992) 607–620.
[44] S. Mitra, S.K. Pal, Fuzzy self-organization, inferencing and rule generation, IEEE Trans. Syst. Man Cybern. 26 (1996) 608–619.
[45] D. Nauck, R. Kruse, A neuro-fuzzy method to learn fuzzy classification rules from data, Fuzzy Sets and Systems 89 (1997) 277–288.
[46] S.K. Pal, D.P. Mandal, Linguistic recognition system based on approximate reasoning, Inform. Sci. 61 (1992) 135–161.
[47] H. Roubos, M. Setnes, Compact fuzzy models through complexity reduction and evolutionary optimization, in: Proceedings of the Ninth IEEE International Conference on Fuzzy Systems, vol. 2, San Antonio, Texas, May 2000, pp. 762–767.
[48] R. Rovatti, R. Guerrieri, G. Baccarani, Fuzzy rules optimization and logic synthesis, in: Proceedings of Second IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'93), vol. 2, San Francisco, USA, March 1993, pp. 1247–1252.
[49] M. Setnes, R. Babuska, U. Kaymak, H.R. van Nauta-Lemke, Similarity measures in fuzzy rule base simplification, IEEE Trans. Syst. Man Cybern. – Part B: Cybernetics 28 (1998) 376–386.
[50] M. Setness, H. Hellendoorn, Orthogonal transforms for ordering and reduction of fuzzy rules, in: Proceedings of the Ninth IEEE International Conference on Fuzzy Systems), vol. 2, San Antonio, Texas, May 2000, pp. 700–705.
[51] C.E. Shannon, W. Weaver, The Mathematical Theory of Communication, University of Illinois Press, Urbana, IL, 1949.
[52] W. Siedlecki, J. Sklansky, A note on genetic algorithms for large-scale feature selection, Pattern Recognition Letters 10 (1989) 335–347.
[53] G. Syswerda, A study of reproduction in generational and steady-state genetic algorithms, in: G. Rawlins (Ed.), Foundations of Genetic Algorithms, Morgan Kaufmann, Los Altos, CA, 1991, pp. 94–112.
[54] L.X. Wang, J.M. Mendel, Generating fuzzy rules by learning from examples, IEEE Trans. Syst. Man Cybern. 25 (2) (1992) 353–361.
[55] Y. Yam, P. Baranyi, C.-T. Yang, Reduction of fuzzy rule base via singular value decomposition, IEEE Trans. Fuzzy Systems 7 (1999) 120–132.
[56] J. Yang, V. Honavar, Feature subset selection using a genetic algorithm, in: Feature Extraction, Construction and Selection, 1998, pp. 118–136 (Chapter 8).

[57] J. Yen, L. Wang, Simplifying fuzzy rule-based models using orthogonal transformation methods, IEEE Trans. Syst. Man Cybern. – Part B: Cybernetics 29 (1999) 13–24.
[58] Y. Yuan, H. Zhuang, A genetic algorithm for generating fuzzy classification rules, Fuzzy Sets and Systems 84 (1996) 1–19.