Genetic Feature Selection for Fuzzy Discretized Data

Luciano Sánchez
Univ. Oviedo, D. Informática
33071 Gijón, Asturias
luciano@uniovi.es

José R. Villar
Univ. Oviedo, D. Informática
33071 Gijón, Asturias
villarjose@uniovi.es

Inés Couso
Univ. Oviedo, D. Estadística
33071 Oviedo, Asturias
couso@uniovi.es

Abstract

A wrapper-type evolutionary feature selection algorithm, able to use fuzzy data, is proposed. In the context of Genetic Learning of Fuzzy Rule-based Classifier Systems (FRBCS), this new algorithm has been applied to a particular kind of instances, comprising fuzzy discretized data (FDD). This data is obtained when passing crisp data through the fuzzification interface of the FRBCS under study.

We have compared the properties of the algorithm proposed here to other approaches, over FDD and crisp data. In case the preprocessed data is intended to be used by a Genetic Learning FRBCS, we can conclude that those algorithms able to use FDD are preferred over the crisp ones, even though there is not fuzziness in the training data being used. Besides, they also are the only alternative when the datasets are imprecise, although this last case is not elaborated in this study.

Keywords: Genetic Fuzzy Systems, Fuzzy Data, Feature Selection.

1 Introduction

The selection of a subset of features for classification problems can be solved either with wrapper or filter methods. Wrappers consider that the classification algorithm is a black box, used by the search algorithm to evaluate each feature subset. Instead, filter methods are independent of the classifier and select features based on properties that good feature sets are supposed to have. Filter methods can produce wrong results, because they do not have into account the learning algorithm. In contrast, the main problem with wrappers is the computing time. If the learning algorithm is fast, binary coded genetic algorithms can be used to search subsets of features with good results [14]. Otherwise, the genetic algorithm can be combined with a different classifier which is faster to learn, but then some of the advantages of the wrapper algorithms over filters are lost. Both approaches have also been combined. For instance, in [1, 20] genetic search and filters are put together.

Most of the feature selection algorithms used in the design of FRBCS are suitable for precise, numerical data, without observation error neither missing values. However, many real-world datasets are coarsely measured. Also, missing values, or incomplete inputs, can appear in otherwise precise data. Lastly, the fuzzification interface of a FRBCS converts crisp data into fuzzy subsets of the set of linguistic labels. For example, a crisp value "20" can be converted into the fuzzy subset 0.8/COLD+0.2/HOT of the set $\{\mathrm{COLD}, \mathrm{HOT}\}$. We will call this last kind of data "fuzzy discretized data" (FDD).

In previous works, we have advocated the adoption of of fuzzy techniques [9], and, importantly, the use of fuzzy fitness functions, for extracting rules from different types of data, including FDD, in classification and regression problems [8, 11]. This is because the same algorithms that can process imprecise data can handle FDD, and this fact allows us to exploit the advantages of these fuzzy techniques on crisp problems. We have also studied how to preprocess imprecise databases [10, 12]. In these last works, we proposed a filter-type evolutionary algorithm, that used a mutual information measure to perform feature selection from vague data. Conversely, in this work we will propose a wrapper-type evolutionary feature selection algorithm for vague data, and we will analyze how well it is suited to FDD. As we will show in Section 2, the new proposal is based on our own extension to the fuzzy case of the k-NN classification algorithm, thus in this sense our algorithm can be regarded as a fuzzy generalization of the SSGA algorithm [1], and therefore it is named Fuzzy-SSGA (FSSGA). Indeed, there are many recent works dealing with feature selection procedures that use fuzzy techniques [17, 15] or are designed to be used in combination with fuzzy systems [19, 18]. However (to the best of our knowledge) other than ours, the only paper where the feature selection of fuzzified continuous data has been studied is [6], where a filter method, based on a similarity function, was used. We are not aware of other works where wrapper-type algorithms have been proposed.

This paper is organized as follows: In Section 2 we introduce our extension of the k-NN algorithm for fuzzy data, that will be wrapped in the GA that performs the selection. In Section 3 we describe this genetic search of the set of features, and in Section 4 we include some benchmarks. The paper finishes with the concluding remarks and the future work.

2 Use of the k-NN in a wrapper algorithm with uncertain data

The most frequent use of the term “fuzzy k-NN” is described in [3]. In that paper, a membership value for each crisp example in the training dataset is introduced, and the class of the object is assigned to the class with higher certainty, in a procedure similar to a statistical kernel classifier. Many publications extend this definition or apply it to practical problems [16]. Even though some of these extensions use a fuzzy set for defining the class of an object [13], we could not find publications where a k-NN algorithm making use of imprecise data is defined.

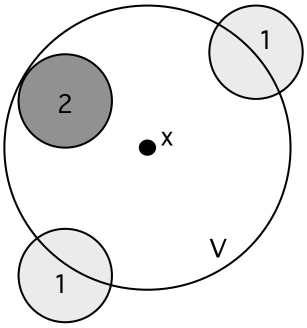

Figure 1: Interval-valued data: The smallest ball centered in x that contains for sure one object has non-null intersection with two other objects. Therefore, $P(1|x) \in [0, 2/3]$ and $P(2|x) \in [1/3, 1]$ and we can not label x (i.e., we assign the set of labels $\{1, 2\}$ to x).

2.1 An extended definition of the k-NN criterion for fuzzy data

From an statistical point of view, the k-NN rule can be derived from the Bayes rule. Let us assume that each object ω is of class(ω). We want to define a decision rule that maps any set of measurements $X(\omega)$ to class(ω), with the lowest number of errors. This rule is known to be

$$c(x) = \arg\max_{i=1\dots,M} P(\text{class}(\omega) = i \mid X(\omega) = x) \quad (1)$$

where M is the number of classes. Let us suppose we are given a sample of size N, where N_i elements belong to the i-th class, and $N = \sum_{i=1,\dots,M} N_i$. For estimating $P(i|x)$ from data, we rewrite eq. (1) first,

$$P(i|x) = \frac{f(x|i)p(i)}{f(x)} \quad (2)$$

then estimate both density functions from the sample. Let V be the smallest ball that

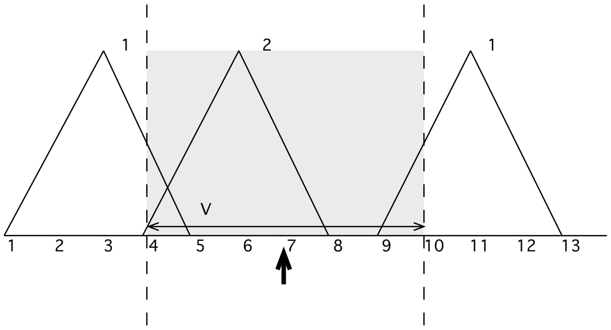

Figure 2: Fuzzy data: The smallest volume centered in 7 that completely contains one object is the interval $V = [4, 10]$. For α-cuts lower than 0.5, the estimations of $P(1|x)$ and $P(2|x)$ intersect. For $\alpha > 0.5$, $P(2|x) > P(1|x)$. Therefore, the class of x is the fuzzy set $0.5/1 + 1/2$.

contains k objects of the sample. Let n_i, $k = \sum_{i=1,\dots,M} n_i$ be the number of objects of the i-th class contained in V. If V is small enough, then

$$\frac{f(x|i)p(i)}{f(x)} \approx \frac{\frac{n_i}{N_i V}\frac{N_i}{N}}{\frac{k}{NV}} = \frac{n_i}{k}. \tag{3}$$

Hence, eq. (1) reduces to the k-NN rule: we label x as the class that appears the most in the smallest ball, centered in x, that contains k objects in the sample.

Let us suppose now that we can not precisely observe x, but an interval that contains it. For example, consider the situation in Figure 1: the smallest volume V that contains one element of the sample also intersects two other objects, but it does not completely contain them. In this example, the application of eq. (1) requires deciding whether $[0, 2/3] \gtrless [1/3, 1]$, and we do not have information enough to know the response, thus we will label the example x with the whole set $\{1, 2\}$.

The fuzzy case is an extension of the interval case. Let us consider the data displayed in Figure 2. We have a one-dimensional problem, where we want to label the point $x = 7$, according to the nearest neighbor rule. We have three imprecisely measured objects surrounding x: two of them, the triangular fuzzy numbers $(1; 3; 5)$ and $(9; 11; 13)$, belong to class 1. A third one, $(4; 6; 8)$, belong to class 2. The smallest volume, centered in 7, that contains one element of the sample, is the interval $[4, 10]$. Now observe that each α-cut of these three sets forms an interval-valued classification problem. For $\alpha \leq 0.5$, V intersects with the three objects. For levels greater than 0.5, the only object in V is that of class 2. Therefore, our knowledge about the class of the point x is given by the fuzzy set

$$0.5/1 + 1/2. \tag{4}$$

Summarizing, in case we are given a sample $(S_1, c_1), \dots, (S_N, c_N)$ of classified objects, where the measurements taken over each object are crisp sets S_i and the class of each object is an element of the set $\{1, \dots, M\}$, we define first the values $\hat{P}_*(c|x)$ and $\hat{P}^*(c|x)$ as the minimum and maximum of the set

$$\left\{ \frac{\sum_{c_j=c} a_j}{\sum a_j} : a_j \in \left\{ \begin{array}{ll} \{0\} & S_j \cap V = \emptyset \\ \{1\} & S_j \subseteq V \\ \{0,1\} & \text{else} \end{array} \right. \right\} \tag{5}$$

where V is the smallest sphere, centered in x, that completely contains k objects of the sample. Our extended k-NN rule assigns to each point x a subset C of $\{1, \dots, M\}$, defined as follows:

$$C(x) = \{i : \hat{P}^*(i|x) \geq \hat{P}_*(c|x), \quad \forall c \neq i\}. \tag{6}$$

In case the measurements are fuzzy sets $\widetilde{X}_i$, the extended k-NN rule assigns to the point x a fuzzy set of classes. Let us define the level cut α of the sample as the interval-valued dataset $([\widetilde{X}_1]_\alpha, c_1), \dots, ([\widetilde{X}_N]_\alpha, c_N)$. If we applied the preceding rule for classifying x on the basis of a level cut α of the sample, we would obtain a (crisp) set of classes $C_\alpha(x)$. We propose that the class of x is a fuzzy subset μ_x of $\{1, \dots, M\}$ defined by the membership functions

$$\mu_x(i) = \sup\{\alpha : i \in C_\alpha(x)\}. \tag{7}$$

It is emphasized that, for classifying either a crisp or a fuzzy set instead of a point, we have to make sure that the volume V completely contains k elements of the sample but also that it contains the whole area being classified. However, we have a certain freedom in the definition of some of the properties of V (for instance, that of V being centered in the point being classified) as soon as V is small enough for the approximation in eq. (3) making sense.

2.2 Symbolic data

The expression (2) holds when x is a vector of real numbers. Instead, when x is an element in a finite space, we have to assume some degree of smoothness in $p(x|c)$, or else we cannot estimate its value in points which do not appear in the sample. Usually, we admit that eq. (3) still holds when the volume V is defined wrt a certain distance. The most common distance is the count of features that match, although there are more complex approaches based on distance tables between features [2].

We will process FDD and, in particular, we are interested in the case where all the sets of linguistic labels form Ruspini's fuzzy partitions. For example, the fuzzification stage can convert a numerical value of 45 degrees into a fuzzy subset like $\{0.0/\mathrm{COLD} + 0.2/\mathrm{WARM} + 0.8/\mathrm{HOT}\}$. Observe that rule based-systems could also manage subsets like $\{0.1/\mathrm{COLD} + 0.3/\mathrm{WARM} + 0.9/\mathrm{HOT}\}$ or $\{0.5/\mathrm{COLD} + 0.5/\mathrm{WARM} + 0.5/\mathrm{HOT}\}$, that do not match any numerical value. Because of space reasons, in this paper we will not include experiments related to these last two cases, however we have included the treatment of such kind of data in the definition of our algorithm.

2.2.1 FDD from crisp instances

We will interpret that the memberships of each crisp piece of data to the elements of the Ruspini's partition define a probability distribution over the set of linguistic labels, thus the fuzzified data is a fuzzy random variable (frv). Let $\widetilde{X}$ and $\widetilde{Y}$ be two of such fuzzified measurements of crisp vectors. Each measurement has n coordinates: $\widetilde{X} = (\widetilde{X}_1, \ldots, \widetilde{X}_n)$ and $\widetilde{Y} = (\widetilde{Y}_1, \ldots, \widetilde{Y}_n)$. We propose that the distance between $\widetilde{X}$ and $\widetilde{Y}$ is the euclidean distance

$$d(\widetilde{X}, \widetilde{Y}) = \left(\bigoplus_{i=1}^{n} d(\widetilde{X}_i, \widetilde{Y}_i)^2 \right)^{0.5} \qquad (8)$$

The distance $d(\widetilde{X}_i, \widetilde{Y}_i)$ between each component depends, in turn, of the probabilities that each label was assigned in the fuzzification. Let us name $(p_1, \ldots, p_l)$ and $(q_1, \ldots, q_l)$ to the probabilities of the l values of the linguistic variable defined for the i-th coordinate. We propose that the distance between them is the distance between both probability distributions:

$$d(\widetilde{X}_i, \widetilde{Y}_i) = \sum_{j=1}^{l} (p_j - q_j)^2. \qquad (9)$$

This distance generalizes the count of features that match. It is emphasized that, when this last distance is used, if two different input values are assigned the same set of memberships in the fuzzification interface, then the distance between them is zero nonetheless.

2.2.2 FDD from instances with missing values and vague data

The memberships of either a missing value or an imprecisely measured data can be understood as a family of probability distributions. We can determine a lower and an upper bound of the distances between these pieces of information as the interval

$$\begin{aligned} d(x_i, y_i) \quad &= \Big\{ \sqrt{\textstyle\sum_{j=1}^{l} (p_j - q_j)^2} : \\ & \quad p_j \in [p_{j*}, p_j^*], \; q_j \in [q_{j*}, q_j^*] \Big\} \end{aligned} \qquad (10)$$

Note that, in this case, the situation is equivalent to the case studied in the preceding section, when the data was imprecisely measured and the imprecision was defined by means of crisp sets. Let us call r to the radius of the volume V in the preceding section. We can define $\hat{P}_*(c|x)$ and $\hat{P}^*(c|x)$ as the minimum and maximum of the set

$$\left\{ \frac{\sum_{c_j = c} a_j}{\sum a_j} : a_j \in \left\{ \begin{array}{ll} \{0\} & \min\{d(S_j, x)\} > r \\ \{1\} & \max\{d(S_j, x)\} < r \\ \{0, 1\} & \text{else} \end{array} \right. \right\} \qquad (11)$$

and use the rule in eq. (6) to obtain the set of classes that the object is assigned.

2.3 Measurement of the error rate of a classifier with imprecise data

For computing the error rate of the classifier over a dataset we want to count the number of misclassifications. However, since the output of the classifier is a set of classes, it is not always possible to decide whether the point is being correctly classified. Generally speaking, the error rate will also be a fuzzy set.

Let us assume that the output of the fuzzy classifier for the j-th element of the dataset is the vector $(\mu_{c_1}, \ldots, \mu_{c_M})$. Let q be the index of the modal point of this set, and let b the index of the second highest membership. According to our proposal in [9], this classification contributes to the total error as much as

$$E_j = \begin{cases} 1/0 + \mu_{c_b}/1 & \text{if } c_j = c_q \\ \mu_{c_j}/0 + 1/1 & \text{else} \end{cases} \tag{12}$$

i.e., the number of errors is the sum, with fuzzy arithmetic, of the values E_j.

3 Genetic Search of the best subset of features

When the data is interval-valued, fuzzy or crisp with missing values, the fitness function is a fuzzy number, and we will need a specially crafted multicriteria genetic algorithm [8] in order to solve the problem. Otherwise, a conventional GA could be used. We give a brief explanation of some parts of these algorithms in the remainder of this section.

3.1 Representation and Genetic Operators

We have used the same representation and operators proposed in [1]. The subsets have fixed cardinality, and we use integer coding for representing the subset of features, because it is more efficient in space than using a binary representation. The length of a chromosome is therefore the number of features, and each allele represents one variable.

Two different crossover operators are used: partial complementary crossover [7] and two point crossover with repair operation (i.e., repeated features are replaced by a non-selected variable, selected at random).

Our mutation operator consists in changing one allele, chosen at random, by another random value that is not repeated.

3.2 Fitness function

The quality of a given subset is given by the average error rate in the test set of the k-NN classifier. Five 50% training-test partitions are used in this evaluation. The fitness value is a fuzzy number: the sum (by means of fuzzy arithmetic operators) of the costs E_j of each test data, as defined in eq. (12).

3.3 Generational scheme

As described in [8], we have used a generational approach with the multiobjective NSGA-II replacement strategy, binary tournament selection based on rank and crowding distance, and a precedence operator that assumes a uniform prior. The nondominated sorting depends on the product of the so-obtained probabilities of precedence. Lastly, the crowding is based on the Hausdorff distance.

4 Numerical analysis of FDD from crisp data

In this section we will show that, when designing GFS, a feature selection algorithm that can use vague data is better than a conventional one, even when our training data is crisp. We have claimed this result before, in [10, 12], where we shown that the relevance of the input variables was dependent on the fuzzy definition of the antecedents of the rules. That is to say, we want to detect the features where the fuzzy discretization of the inputs has lost relevant information, and not to use them. In order to do this, we have to fuzzify the inputs first, and then use a feature selection algorithm that can work with the fuzzified data.

We limit ourselves to FDD from crisp data in the numerical experimentation. The compared performance of both FSSGA, and our former algorithm FMIFS [10][12], over coarse data and data with missing values, are left for future works.

Thirteen different fuzzy rule learning algorithms have been considered, both heuristic and genetic algorithms-based. The heuristic classifiers are described in [4]: no weights (HEU1), same weight as the confidence (HEU2), differences between the confidences (HEU3, HEU4, HEU5), weights tuned by reward-punishment (REWP) and analytical learning (ANAL). The genetic classifiers are: Selection of rules (GENS), Michigan learning (MICH) –with population size 25 and 1000 generations–, Pittsburgh learning (PITT) –with population size 50, 25 rules each individual and 50 generations–, and Hybrid learning (HYBR) –same parameters than PITT, macromutation with probability 0.8– [4]. Lastly, two iterative rule learning algorithms are studied: Fuzzy Ababoost (ADAB) –25 rules of type I, fuzzy inference by sum of votes– and Fuzzy Logitboost (LOGI) –10 rules of type III, fuzzy inference by sum of votes– [10]. All the experiments have been repeated ten times for different permutations of the datasets.

In Table 1 we have compared the results of the new algorithm FMIFS, for five crisp datasets, to those of the original MIFS algorithm, RELIEF [5] and the evolutionary algorithm SSGA [1]. The results of the filter-type, fuzzy feature selection FMIFS [10, 12] are also included. If a fuzzy method (either FMIFS or FSSGA) improves all the crisp methods, the corresponding number is boldfaced. In all cases, a uniform partition of size 3 was used for all the variables, and 5 input variables were selected. Observe that:

- There exist problems (ION, SONAR) where the use of a fuzzy feature selection improves the results for all the fuzzy classifiers that have been tested. This means that some highly informative crisp variables are not anymore relevant after passing through the fuzzification interface.
- The advantages of the FMIFS and FSSGA are clearer in heuristic classifiers than they are in GFS. In particular, Logitboost and Adaboost, which are based on a sum-product based inference, seem to be less influenced by the loss of information in the fuzzification.
- FMIFS consistently outperforms MIFS, and FSSGA outperforms SSGA in PIMA, ION and SONAR. However, in general, FMIFS seems to be slightly more efficient than FSSGA, thus there does not seem to be a definite advantage of wrappers over filters over FDD (remember that wrappers are also slower than filters).
- Any fuzzy algorithm can be safely applied to crisp data (FDD). If the problem is not best suited for a fuzzy algorithm, the results will not be too different than those obtained by a crisp algorithm, as we can see in PIMA, WINE and GERMAN, but the performance will not be degraded.

To sum up, there exist datasets for which, given a certain fuzzy partition, a feature selection algorithm able to use FDD improves significantly the results. Otherwise, in case the GFS optimizes the partition, or the training data is homogeneously distributed, the gain is not relevant.

5 Concluding remarks

The preprocessing of databases with imprecise data is hardly found in the literature. In this paper we have proposed a wrapper-type evolutionary feature selection algorithm, that equals or outperforms other filter and wrapper algorithms. However, there is not evidence that it improves our own fuzzy mutual information based algorithms. In addition, we have shown that there are problems where we obtain an uniform improvement for the whole catalog of learning algorithms that were tested. Intuitively, the method proposed here should be applied when the input partition has few elements or it has not been opti-

	HEU1	HEU2	HEU3	HEU4	HEU5	REWP	ANAL	GENS	MICH	PITT	HYBR	ADAB	LOGI
PIMA - RELIEF	0.289	0.289	0.276	0.276	0.276	0.269	0.269	0.263	0.355	**0.230**	0.256	0.243	0.250
PIMA - SSGA	0.302	0.289	0.263	0.263	0.263	0.263	**0.263**	0.263	0.355	0.243	**0.243**	**0.217**	**0.217**
PIMA - MIFS	0.276	0.276	0.276	0.276	0.276	0.276	0.276	0.269	0.355	0.256	0.276	0.223	0.243
PIMA -FMIFS	0.302	0.289	0.263	0.263	0.263	0.263	**0.263**	**0.243**	0.355	0.250	0.276	**0.217**	**0.217**
PIMA - FSSGA	**0.273**	**0.271**	**0.257**	**0.257**	**0.257**	**0.256**	0.267	0.244	**0.349**	0.240	0.260	0.222	0.235
WINE - RELIEF	0.500	0.411	0.235	0.205	0.176	0.088	0.235	**0.029**	0.647	0.205	**0.029**	0.058	0.058
WINE - SSGA	**0.176**	0.176	0.147	0.235	**0.147**	**0.058**	**0.088**	0.147	**0.147**	0.058	**0.029**	**0.000**	**0.029**
WINE - MIFS	0.323	0.323	0.264	0.205	0.176	0.117	0.235	0.176	0.617	0.058	0.176	0.058	0.058
WINE - FMIFS	**0.176**	**0.147**	**0.117**	0.176	**0.147**	**0.058**	0.147	0.117	0.176	**0.029**	0.088	0.058	0.058
WINE - FSSGA	0.247	0.194	**0.141**	**0.152**	**0.147**	0.088	0.164	0.088	0.205	0.082	0.111	0.070	0.052
GERMAN - RELIEF	0.295	0.285	0.275	0.275	0.275	0.280	0.275	0.270	0.295	0.285	0.295	0.290	0.260
GERMAN - SSGA	0.265	**0.255**	**0.250**	**0.255**	**0.255**	**0.250**	0.260	0.255	0.295	**0.275**	**0.255**	**0.260**	0.255
GERMAN - MIFS	0.280	0.265	0.265	0.265	0.265	0.265	0.260	0.265	0.295	**0.275**	0.285	0.265	**0.250**
GERMAN - FMIFS	**0.255**	**0.255**	0.255	**0.255**	**0.255**	0.260	**0.245**	**0.250**	0.305	**0.275**	**0.255**	0.265	0.270
GERMAN - FSSGA	**0.263**	0.270	0.263	0.263	0.263	0.262	**0.247**	**0.254**	**0.291**	**0.263**	0.267	0.275	0.263
ION - RELIEF	0.328	0.314	0.285	0.285	0.285	0.200	0.257	0.157	0.428	0.228	0.214	0.114	0.142
ION - SSGA	0.200	0.185	0.157	0.157	0.157	0.142	0.157	0.128	0.328	**0.114**	**0.114**	0.514	0.100
ION - MIFS	0.200	0.200	0.200	0.200	0.200	0.185	0.185	0.185	0.357	0.157	0.142	0.514	0.171
ION - FMIFS	**0.185**	**0.142**	**0.128**	**0.128**	**0.128**	**0.128**	0.171	**0.100**	**0.200**	**0.114**	0.128	0.514	**0.085**
ION - FSSGA	**0.174**	**0.145**	**0.142**	**0.142**	**0.142**	**0.117**	**0.137**	**0.120**	**0.214**	0.128	0.125	**0.108**	0.122
SONAR - RELIEF	0.300	0.275	0.250	0.250	0.250	0.275	0.375	0.300	0.300	**0.275**	0.325	0.300	0.250
SONAR - SSGA	0.300	0.325	0.250	0.250	0.250	0.300	0.325	0.250	0.300	0.300	0.250	0.250	0.250
SONAR - MIFS	0.350	0.325	0.300	0.300	0.300	0.350	0.350	0.250	0.350	0.325	0.350	0.350	0.325
SONAR - FMIFS	**0.225**	**0.200**	**0.175**	**0.175**	**0.175**	**0.200**	**0.225**	**0.175**	0.300	**0.275**	**0.225**	**0.150**	**0.200**
SONAR - FSSGA	**0.265**	**0.260**	**0.205**	**0.205**	**0.205**	**0.225**	**0.280**	**0.230**	**0.280**	**0.275**	**0.255**	**0.210**	0.255

Table 1: Average test error, 5x2 cross validation-based design, of different fuzzy rule-based classifiers after a feature selection was performed. 5 input variables and 3 linguistic labels were used for each variable. The last two rows (FMIFS and FSSGA) are fuzzy feature selection algorithms, the remaining are crisp. If a fuzzy method (either FMIFS or FSSGA) improves all the crisp methods, the corresponding number is boldfaced.

mized, but further work is needed to characterize this family of problems.

Acknowledgements

This work was funded by Spanish M. of Science and Technology, grants TIN2005-08036-C05-05 and TIN2007-67418-C03-03.

References

[1] Casillas, J., Cordón, O., del Jesus, M. J., Herrera, F. Genetic Feature Selection in a Fuzzy Rule-Based Classification System Learning Process for high-dimensional problems, Information Sciences 136 (2001) 135-157.

[2] Cost, S., Salzberg, S. A weighted nearest neighbor algorithm for learning with symbolic features. Machine Learning 10 (1), 57-78. 1993

[3] Keller J.M.,Gray M.R. , Givens Jr., J.A., A fuzzy k -nearest neighbor algorithm, IEEE Trans. Systems Man Cybernet. vol 15, 4, pp. 580–585. 1985.

[4] Ishibuchi, H., Nakashima, T., and Nii, M. Classification and Modeling with Linguistic Information Granules. Springer. 2004.

[5] Kira, K. and Rendell, L. A practical approach to feature selection. In: Sleeman and P. Edwards (Eds) Proceedings of the Ninth International Conference on Machine Learning (ICML-92), Morgan Kaufmann, 249-256. 1992.

[6] Li, Y., Wu, Z.-F. Fuzzy feature selection based on min-max learning rule and extension matrix, Pattern Recognition 41, pp. 217–226. 2008.

[7] Liu, W., Wang, M., Zhong, Y., Selecting features with genetic algorithm in handwritten digits recognition. Proc Int. IEEE Conf. Evol. Comp. (ICEC'95), 1, 1995, 396-399

[8] Sánchez, L., Couso, I., Casillas, J. "Modelling vague data with genetic fuzzy systems under a combination of crisp and imprecise criteria". Proc. 2007 IEEE MCDM, Honolulu, USA,. 2007

[9] Sánchez, L., Couso, I., Advocating the use of imprecisely observed data in genetic fuzzy systems, IEEE Transactions on Fuzzy Systems, vol. 15, pp. 551-562, 2007.

[10] Sánchez, L., Suárez, M. R., Villar, J. R., Couso, I. Some Results about Mutual Information-based Feature Selection and Fuzzy Discretization of Vague Data, FUZZ-IEEE 2007, London. 2007.

[11] Sánchez, L., Otero, J., Learning fuzzy linguistic models from low quality data by genetic algorithms. Proc. FUZZ-IEEE 2007, London. 2007.

[12] Sánchez, L., Suárez, M. R., Villar, J. R., Couso, I. Mutual Information-based Feature Selection and Partition Design in Fuzzy Rule-based Classifiers from Vague Data. Int. J. Approximate Reasoning, submitted.

[13] Sarkar, M., Fuzzy-rough nearest neighbor algorithms in classification, Fuzzy Sets and Systems, vol. 158, 2007, 2134–2152

[14] Siedlecki, W., Sklansky, J.: A note on genetic algorithms for large-scale feature selection. Pattern Recognition Letters 10 (1989) 335-347

[15] Sun, H.-J., Sun, M., Mei, Z. Feature selection via fuzzy clustering. Proc. 2006 Int. Conf. on Machine Learning and Cybernetics 2006, art. no. 4028283, pp. 1400-1405. 2006.

[16] Toduka, K., Endo, Y. Fuzzy K-Nearest Neighbor and its Application to Recognize of the Driving Environment, In the 2006 IEEE International Conference on Fuzzy Systems, Vancouver, BC, Canada, pp. 751–756

[17] Uncu, O., Türksen, I.B. A novel feature selection approach: Combining feature wrappers and filters, Information Sciences, 177 (2), pp. 449-466. 2007.

[18] Xiong, N., Funk, P. Construction of fuzzy knowledge bases incorporating feature selection, Soft Computing, 10 (9), pp. 796-804. 2006.

[19] Zhang, Y., Wu, X.-B., Xiang, Z.-R., Hu, W.-L. Design of high-dimensional fuzzy classification systems based on multi-objective, evolutionary algorithm, Journal of System Simulation, 19 (1), pp. 210-215. 2007.

[20] Zhang, P., Verma, B., Kumar, K. Neural vs. statistical classifier in conjunction with genetic algorithm based feature selection. Pattern Recognition Letters (2008) In press