2009 Ninth International Conference on Intelligent Systems Design and Applications

An interpretability-guided modeling process for learning comprehensible fuzzy rule-based classifiers

José M. Alonso, Luis Magdalena
European Centre for Soft Computing (ECSC)
Edificio Científico-Tecnológico, Gonzalo Gutiérrez Quirós s/n,
33600 Mieres, Asturias, Spain
{*jose.alonso,luis.magdalena*}@*softcomputing.es*

***Abstract*—This work presents a new process for building comprehensible fuzzy systems for classification problems. Firstly, a feature selection procedure based on crisp decision trees is carried out. Secondly, strong fuzzy partitions are generated for all the selected inputs. Thirdly, a set of linguistic rules are defined combining the previously generated linguistic variables. Then, a linguistic simplification procedure guided by a novel interpretability index is applied to get a more compact and general set of rules without losing accuracy. Finally, an efficient and simple local search strategy increases the system accuracy while preserving the high interpretability. Results obtained in several benchmark classification problems are encouraging because they show the ability of the new methodology for generating highly interpretable fuzzy rule-based classifiers while yielding accuracy comparable to that achieved by other methods like neural networks and C4.5.**

***Keywords*-interpretability; classification; fuzzy modeling;**

I. Introduction

Comprehensible intelligent systems are more and more demanded for all kind of applications. However, interpretability is really appreciated and it even becomes a strong requirement when dealing with humanistic systems, defined by Zadeh as *those systems whose behavior is strongly influenced by human judgment, perception or emotions* [1].

This paper focuses on classification problems where comprehensibility is of prime concern. Of course, accuracy can not be neglected because, at least at a given level, it is a prerequisite since a system which is not able to achieve a minimum accuracy is useless. Nevertheless, some applications can tolerate a reasonable loss of accuracy if it means getting a transparent and comprehensible model. Sometimes, both criteria (accuracy and interpretability) can be satisfied to a high degree, but usually it is not possible because they represent conflicting goals. Thus, looking for a good interpretability-accuracy trade-off is one of the most difficult and challenging tasks in system modeling.

Fuzzy Logic (FL) was introduced by Zadeh (in 1965) and nowadays interpretability is widely admitted to be the most valuable property of fuzzy rule-based systems (FRBSs). They are pointed out as gray-boxes against other techniques such as neural networks which are viewed as black-boxes. FL represents a useful tool to tackle with the problem of building comprehensible systems. In addition, it is especially useful to handle the intrinsic uncertainty of real-world problems where the available information is usually vague. Notice that, the use of FL favors the comprehensibility of the final model but it is not enough to guarantee it [2]. Two main aspects must be taken into consideration when regarding interpretability of FRBSs (*Description* and *Explanation*). On the one hand, the system description has to be transparent enough to present the system as a whole describing its global behavior and trend. On the other hand, system explanation must consider all possible individual situations, explaining specific behaviors for specific events. Thus, comprehensibility of a FRBS depends on all its components, i.e., it depends on the knowledge base (including both variables and rules) transparency but also on the inference mechanism understanding.

Main aspects affecting to the readability of fuzzy systems have been thoroughly analyzed [3]. In addition, a complete study on the interpretability constraints most frequently used in fuzzy modeling has been recently published [4]. Finally, in the fuzzy modeling literature there are two main trends regarding the search of the optimum interpretability-accuracy trade-off: (1) those first focused on interpretability and then on accuracy [5]; (2) those who give priority to accuracy and then try to improve interpretability [6].

This work proposes a new fuzzy modeling process with the aim of getting a good interpretability-accuracy trade-off when building FRBSs for classification tasks also called fuzzy rule-based classifiers (FRBCs).

The rest of the paper is structured as follows. Section II describes the proposed modeling process. Section III explains the experiments made and the obtained results. Finally, section IV draws some conclusions and future works.

II. Methodology

The starting point is the HILK (Highly Interpretable Linguistic Knowledge) fuzzy modeling methodology [7] which focuses on making easier the design process of interpretable FRBSs. It offers an integration framework for combining both expert knowledge and knowledge extracted

978-0-7695-3872-3/09 $26.00 © 2009 IEEE
DOI 10.1109/ISDA.2009.24

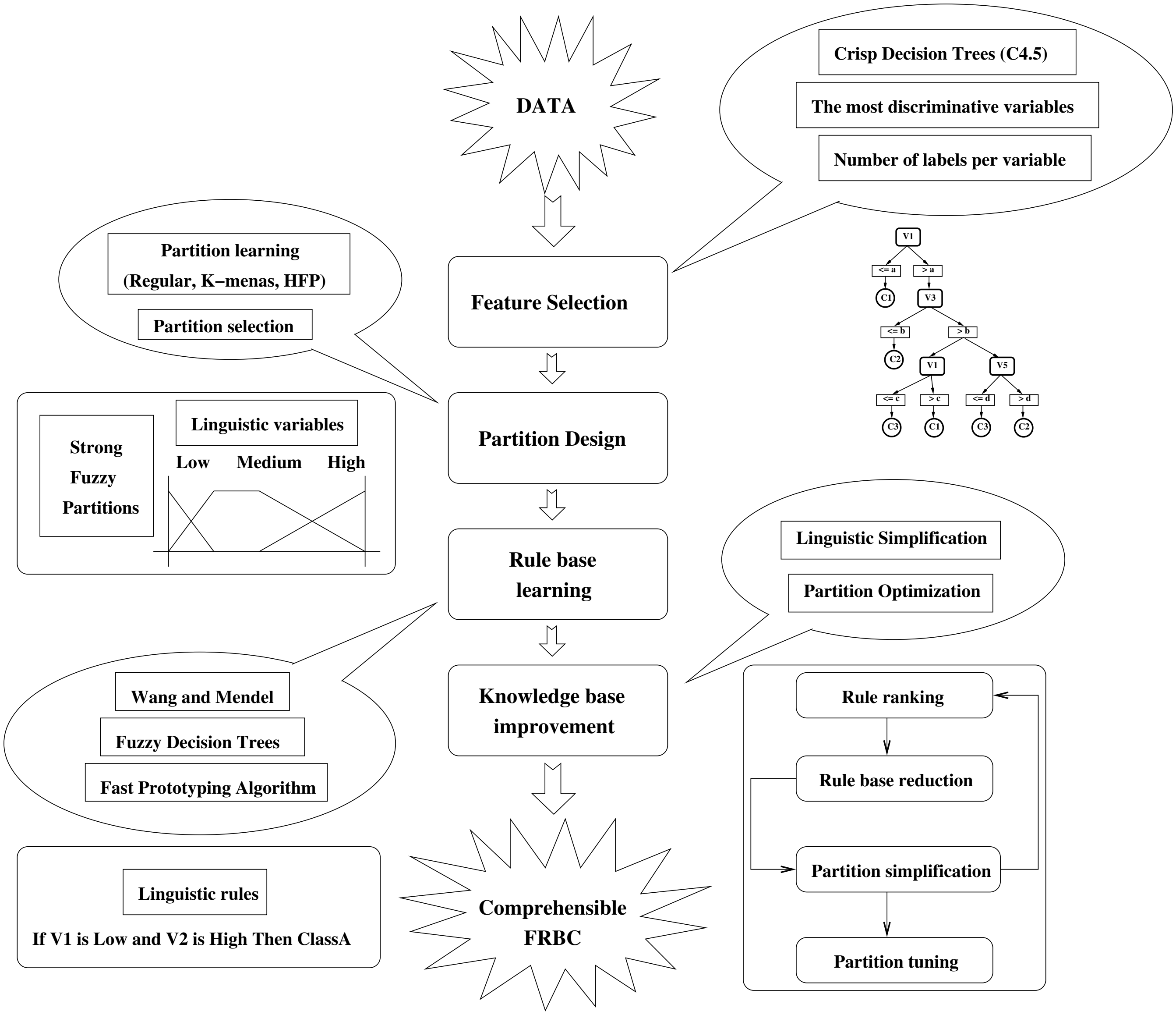

Figure 1. Scheme of the proposed fuzzy modeling process.

from data, which is likely to yield robust and compact systems. This paper focuses on automatic learning from data taking profit of the general framework provided by HILK (strong fuzzy partitions, global semantics, Mamdani rules, linguistic simplification, partition tuning, etc.) and adding some new functionalities (feature selection, interpretability-guided simplification, etc.) in order to get comprehensible FRBCs. Figure I shows graphically the global scheme of the proposed fuzzy modeling process. The whole process is made up of four main steps (the most relevant components will be detailed in the following sections):

- **Feature selection**. Finding out the most discriminative variables and the most suitable number of labels.
- **Partition design**. The readability of fuzzy partitioning is a prerequisite to build interpretable FRBCs. It includes automatic generation of fuzzy partitions from data and partition selection.
- **Rule base learning**. Linguistic rules are automatically extracted from data.
- **Knowledge base improvement**. Iterative refinement process of both partitions and rules.

A. *Feature selection*

We have implemented a feature selection procedure based on the popular C4.5 algorithm introduced by Quinlan in [8]. This algorithm lets us discover the most discriminative variables. In addition, generated crisp decision trees can be easily translated into rules reading them from the root to the leaves [9]. Moreover, the number of breaking values per variable appearing in a tree gives an estimation of the number of fuzzy labels needed for that variable. Figure II-A shows a simple example.

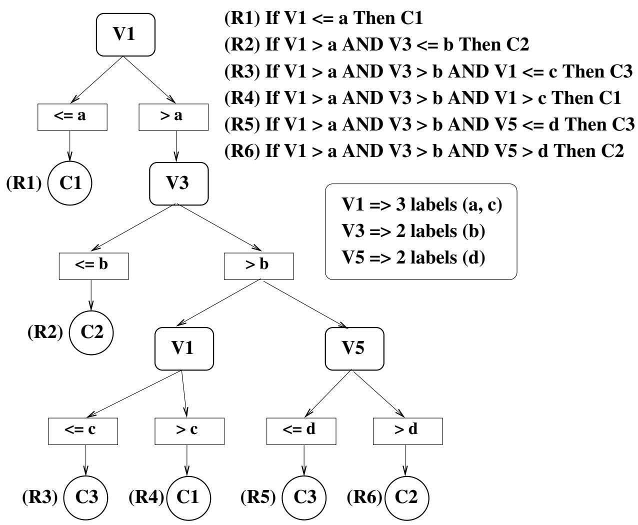

Figure 2. Crisp decision tree.

B. Partition learning and selection

Once selected the most influential variables and the number of fuzzy labels for each of them, the next step is generating the best fitted fuzzy partitions. The use of strong fuzzy partitions (SFPs) satisfies semantic constraints demanded to get comprehensible partitions. Three different partitions are generated for each variable: (1) REG, uniformly distributed partition on the universe of discourse; (2) KM, partition defined by the centroids provided by the K-means algorithm [10]; and (3) HFP, partition generated by a fuzzy method guided by interpretability [11]. Then, generated partitions are compared according to three quality criteria:

$$PE = -\frac{1}{n}\left\{\sum_{k=1}^{n}\sum_{i=1}^{M}\left[u_{ik}\log_a(u_{ik})\right]\right\} \tag{1}$$

$$PC = \frac{\sum\limits_{k=1}^{n}\sum\limits_{i=1}^{M} u_{ik}^2}{n} \tag{2}$$

$$\begin{aligned} ChI \;=\; & \frac{1}{n}\sum_{k=1}^{n}\max_i u_{ik} \\ & - \frac{2}{M(M-1)}\sum_{i=1}^{M-1}\sum_{j=i+1}^{M}\frac{1}{n}\sum_{k=1}^{n}\min(u_{ik},u_{jk}) \end{aligned} \tag{3}$$

The notation is as follows: u_{ik} is the degree of membership of the *k-th* element of the data set to the *i-th* element of the fuzzy partition, M stands for the number of terms of the fuzzy partition, and n represents the cardinality of the data set. An absolute majority voting process is applied. The partition winning at least two criteria is selected. A good partition should minimize the partition entropy (*PE*) [12] while maximizing both the partition coefficient (*PC*) [12] and the Chen index (*ChI*) [13].

C. Rule base learning

After designing all the fuzzy partitions it is time to describe the system behavior in the form of linguistic rules [14]:

If X_a is A_a^i **AND** ... **AND** X_z is A_z^j **Then** *Y* is C^n

where X_a is the name of the input variable a, while A_a^i represents the label i of such variable. C^n represents the output class. Note that we are imposing global semantics, i.e., all the rules use the same linguistic terms defined by the same fuzzy sets. Three different algorithms have been selected to generate rules from data with the previously defined fuzzy partitions:

- **Wang and Mendel (WM)** [15]. It starts by generating one rule for each data pair of the training set but new rules will compete with existing ones. As a result, WM generates complete rules (considering all the available variables) which are quite specific.
- **Fuzzy Decision Tree (FDT)** [16]. It generates a decision tree (directly from data) which is translated into quite general incomplete rules (only a subset of input variables is considered). In addition, inputs are sorted according to their importance (minimizing the entropy).
- **Fast Prototyping Algorithm (FPA)** [17]. It generates rules more general than the ones produced by WM, but at the same time more specific than the ones generated by FDT. It starts generating a grid with all possible combinations of input labels and then, in an iterative process, outputs are defined removing redundancies and inconsistencies. If the number of inputs (and labels defined per input) is high then FPA is quite inefficient. Therefore it needs a previous feature selection process in order to tackle with complex problems.

D. Linguistic simplification

With the aim of getting a more compact and general rule base HILK offers a powerful and flexible simplification procedure which affects to the whole knowledge base (KB) including both rule base simplification and partition reduction. It starts looking for redundant elements (labels, inputs, rules, etc.) that can be removed without altering the system accuracy. Then, it tries to merge elements always used together. Finally, it forces removing elements apparently needed but not contributing too much to the final accuracy.

Thanks to the use of global semantics rule comparison can be directly made at linguistic level. In addition, the process is absolutely deterministic. As a result, it is human-oriented and quite intuitive. However, final results depend on the rule ranking. Therefore we have upgraded the simplification procedure of HILK adding a new rule ranking step previous to each simplification task. Such ranking is based on a novel interpretability index:

$$RBC = \sum_{j=1}^{NR} \left[\prod_{a=1}^{NI} \left(1 - \frac{LT_a^j}{NL_a} \right) \right] \tag{4}$$

A rule base (RB) is made up of a set of rules, so the total rule base complexity (RBC) is given as the addition of all the r-complexity indices measured for the NR rules. Each rule involves a set of premises, so the complexity of a rule is measured as the product of all the p-complexity indices for the NI inputs used in the rule. A p-complexity index evaluates the complexity of a premise and it is computed regarding all the involved linguistic propositions of form $X_a \ is \ A_a^i$ where the linguistic term A_a^i assigned to variable X_a can correspond to one of the NL_a elementary terms defined in the fuzzy partition of the input X_a, but it can also be a convex hull of elementary terms corresponding to OR and NOT combinations (only combinations of adjacent elementary terms are allowed) which turn up as result of the merging of rules and linguistic terms made by the simplification procedure. LT_a^j counts the number of elementary terms in A_a^i:

- Zero when input X_a is not considered in the rule.
- One for elementary terms.
- Number of elementary terms combined with OR. For instance, it is equals two for $Low \ OR \ Medium$.
- NL_a minus one half for NOT composite terms what penalizes NOT against OR composite terms involving all the elementary terms minus one.

This new interpretability index is based on conclusions derived from a web poll study devoted to discover the main influential aspects when assessing interpretability of fuzzy systems [18]. In short, people usually prefer rules free of NOT composite terms. In addition, the increase of rule complexity perceived by people is not linear with the number of involved premises, so we have used product for combining complexity of premises.

E. Local optimization

The last step in the whole fuzzy modeling process is devoted to increase the system accuracy while preserving the high interpretability previously achieved. It consists of a membership function tuning constrained to keep the SFP property. System inputs are ranked regarding their frequency of use in the rule base. Thus, the optimization procedure starts with the inputs most frequently used. The detailed algorithm is described in [17]. To sum up, it is a hill climbing method with memorization of the previous successes made label by label and based on the classical local search strategy proposed by Solis and Wetts [19]. It lets increase accuracy in only a few iterations but it does not guarantee to find the global optimum. The algorithm stops when the maximum number of iterations is achieved, or the fitness function is under a predefined threshold. After modifying one label, the process comes back to the starting point.

III. EXPERIMENTS

The new proposal was evaluated with four benchmark classification problems freely available from the UCI (University of California, Irvine) repository[1]. IRIS and WINE are well-known general purpose classification problems, while WBCD and NEWTHYROID are related to medical applications. All of them correspond to problems where the comprehensibility of the classifier is highly appreciated:

- **IRIS**. Database created by Fisher. It is likely to be the most famous database in the pattern recognition literature. The goal is to classify three varieties of the iris plant (150 instances, 4 attributes, and 3 classes).
- **WINE**. Chemical analysis of wines grown in the same region in Italy but derived from three different cultivars (178 instances, 13 attributes, and 3 classes).
- **WBCD**. Classification of two cancer states (benign or malignant), obtained from the University of Wisconsin Hospitals (683 instances, 9 attributes, and 2 classes).
- **NEWTHYROID**. Predicting the type of patient's thyroid (215 instances, 5 attributes, and 3 classes).

Table I summarizes the averaged results after running 10-fold cross-validation. The same process is repeated for each problem. The data set is divided into 10 parts of equal size keeping the original distribution (percentage of elements for each class) in the whole set. One part is used as test set whereas the remainder are used as training set.

All tested algorithms are implemented in Fispro[2] and/or KBCT[3], two free software tools for designing FRBSs. In addition, for comparison purpose the two first rows of the tables show results provided by other methods implemented in Weka[4]: MP (Multilayer Perceptron) which yields very accurate neural network classifiers (disregarding comprehensibility), and C45 (Quinlan's decision trees) which provides good interpretability-accuracy trade-offs with accuracy smaller than MP. Accuracy (ACC) is computed as the ratio of samples correctly classified for train and test. Interpretability is characterized by the number of rules (NR); the total rule length (TRL) computed as the total number of premises for all the rules; and average rule length (ARL) defined as the average number of premises per rule. The best interpretability-accuracy trade-offs obtained by our method are remarked with symbol [*] while the most comprehensible solutions are identified with [+]. Notice that, C45

[1]http://www.ics.uci.edu/-mlearn/MLSummary.html
[2]http://www.inra.fr/internet/Departements/MIA/M/fispro/
[3]http://www.mat.upm.es/projects/advocate/kbct.htm
[4]http://www.cs.waikato.ac.nz/ml/weka/

Table I
RESULTS OF THE EXPERIMENTATION.

IRIS					
Method	ACC(train)	ACC(test)	NR	TRL	ARL
MP	**0.987**	**0.973**			
C45	**0.98**	**0.96**	**4.7**	**12.5**	**2.66**
WM	0.8376	0.8202	4.6	9.2	2
S	0.8376	0.8202	3	3.6	1.2
S-IC	0.8376	0.8202	3	3.6	1.2
S-DC	0.8376	0.8202	3	3.6	1.2
S-IC-DC	**0.8376**	**0.8202**	**3**	**3.6**	**1.2**
O-SW	**0.9595**	**0.9466**	**3**	**3.6**	**1.2**
FDT	0.9073	0.9001	3.7	3.7	1
S	0.9073	0.9001	3	3	1
S-IC	0.9073	0.9001	3	3	1
S-DC	0.9073	0.9001	3	3	1
S-IC-DC	**0.9073**	**0.9001**	**3**	**3**	**1**
O-SW	**0.9609**	**0.94**	**3**	**3**	**1**
FPA	0.9073	0.9001	5	10	2
S	0.9073	0.9001	3	3	1
S-IC	0.9073	0.9001	3	3	1
S-DC	0.9073	0.9001	3	3	1
S-IC-DC	**0.9073**	**0.9001**	**3**	**3**	**1**
O-SW [*] [+]	**0.9609**	**0.94**	**3**	**3**	**1**

WBCD					
Method	ACC(train)	ACC(test)	NR	TRL	ARL
MP	**0.992**	0.9604			
C45	**0.9801**	**0.9604**	**10.5**	**41.9**	**3.99**
WM	0.9662	0.9518	92.3	532.3333	6
S	0.9681	0.9531	12.8	60.7	4.2979
S-IC	0.9674	0.959	13	61.1	4.3577
S-DC	**0.9674**	**0.9546**	**11.7**	**48.6**	**3.878**
S-IC-DC	0.9668	0.9546	11.8	49.8	3.8897
O-SW	**0.977**	**0.9503**	**11.7**	**48.6**	**3.878**
FDT	0.9644	0.9619	13.5	45.7	3.1104
S	**0.9649**	**0.9589**	**7.4**	**20.2**	**2.6824**
S-IC	0.9649	0.9589	7.6	20.9	2.7094
S-DC	0.9649	0.9589	7.5	20.7	2.7163
S-IC-DC	0.9649	0.9574	7.5	20.9	2.7413
O-SW [*]	**0.9724**	**0.9708**	**7.4**	**20.2**	**2.6824**
FPA	0.9647	0.962	142.8	549.6667	6
S	**0.9666**	**0.9457**	**6**	**18.8**	**3.0091**
S-IC	0.9668	0.9546	6.6	22.4	3.1745
S-DC	0.9663	0.9545	6.9	22.6	3.1612
S-IC-DC	0.9669	0.9517	6.6	20.6	2.9809
O-SW [+]	**0.9722**	**0.9487**	**6**	**18.8**	**3.0091**

WINE					
Method	ACC(train)	ACC(test)	NR	TRL	ARL
MP	**1**	**0.9719**			
C45	**0.9881**	**0.9385**	**5.4**	**13.7**	**2.537**
WM	0.8714	0.8712	13.1	54.2	3.9
S	0.8738	0.8712	5.1	15	2.7744
S-IC	**0.8738**	**0.8712**	**5**	**14.8**	**2.8208**
S-DC	0.8726	0.8712	5.1	14.4	2.6755
S-IC-DC	0.8726	0.8712	5.2	14.7	2.6823
O-SW	**0.9662**	**0.9103**	**5**	**14.8**	**2.8208**
FDT	0.8788	0.8546	5.1	11.7	2.2252
S	0.8788	0.8546	4.4	9.7	2.1401
S-IC	0.8807	0.849	4.3	9.5	2.15
S-DC	0.8788	0.8546	4.4	9.7	2.1401
S-IC-DC	**0.8813**	**0.8546**	**4.3**	**9.4**	**2.1334**
O-SW [*] [+]	**0.9569**	**0.9101**	**4.3**	**9.4**	**2.1334**
FPA	0.8896	0.8825	15.7	65.6	3.9
S	0.8908	0.8714	4.7	12.9	2.6621
S-IC	0.8908	0.8769	4.6	12.3	2.6217
S-DC	0.892	0.877	4.8	12.9	2.6171
S-IC-DC	**0.892**	**0.8714**	**4.5**	**11.3**	**2.485**
O-SW	**0.947**	**0.9043**	**4.5**	**11.3**	**2.485**

NEWTHYROID					
Method	ACC(train)	ACC(test)	NR	TRL	ARL
MP	**0.9886**	**0.9675**			
C45	**0.9829**	**0.9205**	**8**	**31.3**	**3.9125**
WM	0.8668	0.8375	14.1	63.5	4.4
S	**0.8698**	**0.8468**	**5.3**	**16.1**	**2.9516**
S-IC	0.8718	0.8515	5.3	16.8	3.0766
S-DC	0.873	0.847	5.7	17.5	3
S-IC-DC	0.8724	0.847	5.5	16.4	2.88
O-SW	**0.9329**	**0.8978**	**5.3**	**16.1**	**2.9516**
FDT	0.8733	0.856	7.4	14.1	1.8833
S	0.8738	0.856	6	11.1	1.8151
S-IC	0.8738	0.856	6	11.1	1.8151
S-DC	0.8738	0.856	6	11.1	1.8151
S-IC-DC	**0.8738**	**0.856**	**6**	**11.1**	**1.8151**
O-SW [*]	**0.9386**	**0.907**	**6**	**11.1**	**1.8151**
FPA	0.874	0.8696	13.8	62.2	4.4
S	**0.8797**	**0.8701**	**4.6**	**14.4**	**3.0633**
S-IC	0.8781	0.8606	4.7	14.9	3.0616
S-DC	0.8776	0.8746	5.2	16.1	3.0762
S-IC-DC	0.8771	0.8746	5.2	15.4	2.9386
O-SW [+]	**0.9256**	**0.8932**	**4.6**	**14.4**	**3.0633**

sometimes obtains a solution non-dominated by our method, i.e., a more accurate (but less comprehensible) solution.

First, the feature selection procedure identifies the inputs to consider as well as their number of labels. Second, for each input three partitions (REG, KM, and HFP) are generated, and then compared to select the best one according to data distribution. Third, rules are induced from data (WM, FDT, and FPA). Then, each KB is simplified four times exploring four different rule ranking options: (1) S stands for simplification without changing the rule ranking provided by the rule induction algorithm; (2) S-IC means rule ranking from most simple to most complex rules; (3) S-DC is just the inverse ranking (decreasing complexity); (4) S-IC-DC tries the tree previous strategies in parallel. At each intermediate simplification step the solution yielding the smallest complexity is selected as the KB to be simplified in the next step. Finally, simplified KBs are compared and the most compact one is selected (set in boldface in the tables) to be optimized (O-SW).

In all analyzed problems, our method is able to yield classifiers more robust, compact and comprehensible than the ones obtained with C4.5. However, such gain of interpretability is obtained at the cost of a loss of accuracy. Comparing C4.5 with the best trade-offs obtained by our method, we lose, in mean (for the four problems), around 2.5% regarding training patterns and around 1.3% with respect to test. Furthermore, in the case of the simplest classifiers the loss is around 2.9% for training and around 2.2% for testing. This reduction of accuracy seems absolutely reasonable taking into account the simplicity and comprehensibility of the classifiers. In addition, our method exhibits a good generalization ability obtaining similar accuracy for both training and test. This fact is mainly due to the combination of simplification and optimization procedures.

Finally, comparing the proposed rule ranking options, S-IC-DC seems to be the best one because it gives the best results in most cases (six over twelve). Nevertheless, we observe that many times (four over twelve) preserving the rule ranking provided by rule induction (S) yields better results. This fact may be due to a memory effect. Every simplification step has an influence in the whole series of subsequent ones. However, S-IC-DC only analyzes one step. Keeping a temporal sliding window could yield better results but it would increase significantly the computational cost.

IV. Conclusions

This paper has proposed new functionalities (C45-based feature selection and interpretability-guided rule ranking) as well as a new way of combining tools (partition design, rule generation, simplification, and optimization) already provided by HILK methodology. In consequence we have upgraded HILK to a new and powerful version HILK++ able to get very encouraging results. However, more work still remains to be done. We will try our process with more benchmark databases. In addition, it is necessary to explore other feature selection procedures as well as other optimization strategies. For instance, a GA-based approach could help to overcome the intrinsic problems of local search, like falling in local minimum, yielding even better results.

Acknowledgment

This work has been partially funded by the Foundation for the Advancement of Soft Computing (Mieres, Asturias, Spain) and Spanish government (CICYT) under project: TIN2008-06890-C02-01.

References

[1] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *Parts I, II, and III. Information Sciences*, vol. 8, 8, 9, pp. 199–249, 301–357, 43–80, 1975.

[2] J. M. Alonso and L. Magdalena, "A conceptual framework for understanding a fuzzy system," in *IFSA-EUSFLAT 2009*, pp. 119–124.

[3] S. Guillaume, "Designing fuzzy inference systems from data: An interpretability-oriented review," *IEEE Transactions on Fuzzy Systems*, vol. 9(3), pp. 426–443, 2001.

[4] C. Mencar and A. M. Fanelli, "Interpretability constraints for fuzzy information granulation," *Information Sciences*, vol. 178, pp. 4585–4618, 2008.

[5] J. Casillas et al., *Accuracy improvements in linguistic fuzzy modeling*. Studies in Fuzziness and Soft Computing, Springer-Verlag, Heidelberg, 2003, vol. 129.

[6] ——, *Interpretability issues in fuzzy modeling*. Studies in Fuzziness and Soft Computing, Springer-Verlag, Heidelberg, 2003, vol. 128.

[7] J. M. Alonso, L. Magdalena, and S. Guillaume, "HILK: A new methodology for designing highly interpretable linguistic knowledge bases using the fuzzy logic formalism," *Int. Journal of Intelligent Systems*, vol. 23(7), pp. 761–794, 2008.

[8] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[9] J. Abonyi, J. A. Roubos, and F. Szeifart, "Data-driven generation of compact, accurate, and linguistically sound fuzzy classifiers based on a decision-tree initialization," *International Journal of Approximate Reasoning*, vol. 32, pp. 1–21, 2003.

[10] J. A. Hartigan and M. Wong, "A k-means clustering algorithm," *Applied Statistics*, vol. 28, pp. 100–108, 1979.

[11] S. Guillaume and B. Charnomordic, "Generating an interpretable family of fuzzy partitions," *IEEE Transactions on Fuzzy Systems*, vol. 12(3), pp. 324–335, 2004.

[12] J. C. Bezdek, "Pattern recognition with fuzzy objective functions algorithms," *Plenum Press*, 1981.

[13] M.-Y. Chen, "Establishing interpretable fuzzy models from numeric data," in *4th World Congress on Intelligent Control and Automation IEEE*, 2002, pp. 1857–1861.

[14] E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic systems," *IEEE Transactions on Computers*, vol. 26(12), pp. 1182–1191, 1977.

[15] L.-X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22 (6), pp. 1414–1427, 1992.

[16] H. Ichihashi, T. Shirai, K. Nagasaka, and T. Miyoshi, "Neuro-fuzzy ID3: A method of inducing fuzzy decision trees with linear programming for maximizing entropy and an algebraic method for incremental learning," *Fuzzy Sets and Systems*, vol. 81, pp. 157–167, 1996.

[17] P.-Y. Glorennec, *Algorithmes d'apprentissage pour systèmes d'inférence floue*. Editions Hermès, Paris, 1999.

[18] J. M. Alonso and L. Magdalena, "An experimental study on the interpretability of fuzzy systems," in *IFSA-EUSFLAT 2009*, pp. 125–130.

[19] F. Solis and J. Wetts, "Minimization by random search techniques," *Mathematics of Operation Research*, vol. 6, 1981.