ALM: A Methodology for Designing Accurate Linguistic Models for Intelligent Data Analysis *

Oscar Cordón and Francisco Herrera

Dept. of Computer Science and Artificial Intelligence, E.T.S. de Ingeniería Informática, University of Granada, 18071 - Granada, Spain,
`ocordon, herrera@decsai.ugr.es`,
WWW home page: `http://decsai.ugr.es/~ocordon, ~herrera`

Abstract. In this paper we introduce *Accurate Linguistic Modelling*, an approach to design linguistic models from data, which are accurate to a high degree and may be suitably interpreted. Linguistic models constitute an Intelligent Data Analysis structure that has the advantage of providing a human-readable description of the system modelled in the form of linguistic rules. Unfortunately, their accuracy is sometimes not as high as desired, thus causing the designer to discard them and replace them by other kinds of more accurate but less interpretable models. ALM has the aim of solving this problem by improving the accuracy of linguistic models while maintaining their descriptive power, taking as a base some modifications on the interpolative reasoning developed by the Fuzzy Rule-Based System composing the model. In this contribution we shall introduce the main aspects of ALM, along with a specific design process based on it. The behaviour of this learning process in the solving of two different applications will be shown.

1 Introduction

Nowadays, one of the most important areas for the application of Fuzzy Set Theory as developed by Zadeh in 1965 [14] are Fuzzy Rule-Based Systems (FRBSs). These kinds of systems constitute an extension of classical Rule-Based Systems, because they deal with fuzzy rules instead of classical logic rules.

In this approach, fuzzy IF-THEN rules are formulated and a process of fuzzification, inference and defuzzification leads to the final decision of the system. Although sometimes the fuzzy rules can be directly derived from expert knowledge, different efforts have been made to obtain an improvement on system performance by incorporating learning mechanisms guided by numerical information to define the fuzzy rules and/or the membership functions associated to them. Hence, FRBSs are a suitable tool for Intelligent Data Analysis where the structure considered to represent the available data is a Fuzzy Rule Base.

From this point of view, the most important application of FRBSs is *system modelling* [10], which in this field may be considered as an approach used to

* This research has been supported by CICYT TIC96-0778

model a system making use of a descriptive language based on Fuzzy Logic with fuzzy predicates [11]. In this kind of modelling we may usually find two contradictory requirements, accuracy and interpretability.

When the main requirement is the accuracy, descriptive Mamdani-type FRBSs [7] are considered which use fuzzy rules composed of linguistic variables that take values in a term set with a real-world meaning. This area is called *Fuzzy Linguistic Modelling* due to the fact that the linguistic model consists of a set of linguistic descriptions regarding the behaviour of the system being modelled [11]. Nevertheless, the problem is that sometimes the accuracy of these kinds of models is not sufficient to solve the problem in a right way. In order to solve this problem, in this paper, we introduce *Accurate Linguistic Modelling* (ALM), a Linguistic Modelling approach which will allow us to improve the accuracy of linguistic models without losing its interpretability to a high degree.

To do so, this contribution is set up as follows. In Section 2, a brief introduction to FRBSs is presented with a strong focus on descriptive Mamdani-type ones. Section 3 is devoted to introduce the basis of ALM. In Section 4, a Linguistic Modelling process based on it is proposed. In Section 5, the behaviour of the linguistic models generated to solve two different applications is analysed. Finally, in Section 6, some concluding remarks will be pointed out.

2 Fuzzy Rule-Based Systems

An FRBS presents two main components: 1) the *Inference Engine*, which puts into effect the fuzzy inference process needed to obtain an output from the FRBS when an input is specified, and 2) the *Fuzzy Rule Base*, representing the known knowledge about the problem being solved in the form of fuzzy IF-THEN rules.

The structure of the fuzzy rules in the Fuzzy Rule Base determines the type of FRBS. Two main types of fuzzy rules are usually found in the literature:

1. *Descriptive Mamdani-type fuzzy rules* [7] —also called linguistic rules— which present the expression:

$$\text{IF } X_1 \text{ is } A_1 \text{ and ... and } X_n \text{ is } A_n \text{ THEN } Y \text{ is } B_i$$

   with $X_1, \ldots, X_n$ and Y being the input and output linguistic variables, respectively, and $A_1, \ldots, A_n$ and B being linguistic labels, each one of them having associated a fuzzy set defining its meaning.
2. *Takagi-Sugeno-Kang (TSK) fuzzy rules* [12], which are based on representing the consequent as a polynomial function of the inputs:

$$\text{IF } X_1 \text{ is } A_1 \text{ and ... and } X_n \text{ is } A_n \text{ THEN } Y = p_1 \cdot X_1 + \ldots + p_n \cdot X_n + p_0$$

   with $p_0, p_1, \ldots, p_n$ being real-valued weights.

The structure of a descriptive Mamdani-type FRBS is shown in Figure 1. As can be seen, and due to the use of linguistic variables, the Fuzzy Rule Base

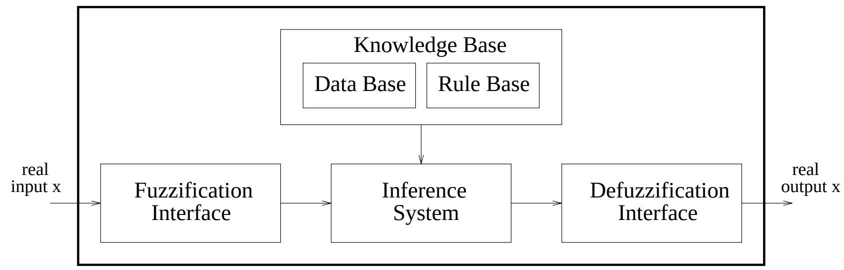

Fig. 1. Generic structure of a descriptive Mamdani-type Fuzzy Rule-Based System

becomes a Knowledge Base (KB) composed of the Rule Base (RB), constituted by the collection of linguistic rules joined by means of the connective *also*, and of the Data Base (DB), containing the term sets and the membership functions defining their semantics.

On the other hand, the Inference Engine is comprised by three components: a *Fuzzification Interface*, which has the effect of transforming crisp input data into fuzzy sets, an *Inference System*, that uses these together with the KB to perform the fuzzy inference process, and a *Defuzzification Interface*, that obtains the final crisp output from the individual fuzzy outputs inferred.

The *Inference System* is based on the application of the Generalized Modus Ponens, extension of the classical logic Modus Ponens. It is done by means of the Compositional Rule of Inference, which in its simplest form is reduced to [2]:

$$\mu_{B'}(y) = I(\mu_{A_i}(x_0), \mu_B(y))$$

with $x_0 = (x_1, \ldots, x_n)$ being the current system input, $\mu_{A_i}(x_0) = T(A_1(x_1), \ldots, A_n(x_n))$ being the matching degree between the rule antecedent and the input —T is a conjunctive operator (a t-norm)— and I being a fuzzy implication operator.

The Compositional Rule of Inference is applied to each individual rule, thus obtaining an output fuzzy set $B_i^{'}$ from each rule in the KB. The *Defuzzification Interface* aggregates the information provided by these fuzzy sets and transforms it into a single crisp value by working in one of the two following ways [2]:

1. *Mode A: Aggregation first, defuzzification after:* The individual fuzzy sets inferred are aggregated to obtain a final fuzzy set B' by means of a fuzzy aggregation operator G —which models the *also* operator that relates the rules in the base—. Then, a defuzzification method D is applied to transform the latter into a crisp value y_0 that will be given as system global output:

$$\mu_{B'}(y) = G\left\{\mu_{B_1'}(y), \mu_{B_2'}(y), \ldots, \mu_{B_n'}(y)\right\} \quad ; \quad y_0 = D(\mu_{B'}(y))$$

   Usual choices for G and D are, respectively, the minimum and maximum operators and the Centre of Gravity and Mean of Maxima defuzzification methods.

2. *Mode B: Defuzzification first, aggregation after:* In this case, the contribution of each fuzzy set inferred is individually considered and the final crisp value is obtained by means of an operation (an average, a weighted average, or the selection of one of them, among others) performed on a crisp characteristic value of each one of the individual fuzzy sets.
   The most commonly used characteristic values are the Centre of Gravity and the Maximum Value Point. Several importance degrees are considered to select or weight them, the matching degree of the rule and the area or the height of the consequent fuzzy set among others [2].

3 ALM: An Approach for Generating Accurate Linguistic Models for Intelligent Data Analysis

One of the most interesting features of an FRBS is the interpolative reasoning it develops, which plays a key role in its high performance and is a consequence of the *cooperation among the fuzzy rules composing the KB*. As mentioned in the previous Section, the output obtained from an FRBS is not usually due to a single fuzzy rule but to the cooperative action of several fuzzy rules that have been fired, because they match the input to the system to some degree.

ALM will deal with the way in which the linguistic model make inference in order to improve its accuracy while not losing its description. Hence, it will be based on two main aspects that will be described in the two following subsections. The remaining one in this Section analyses some interesting remarks of the proposed approach.

3.1 A New Descriptive Knowledge Base Structure for Locally Improving the Model Accuracy

Some problems derived from the inflexibility of the concept of linguistic variable (see [1]) makes the usual linguistic model structure shown in the previous Section present low accuracy when working with very complex systems. Due to this reason, we consider obtaining a new more flexible KB structure that allows us to improve the accuracy of linguistic models without losing their interpretability.

In [9], an attempt was made to put this idea into effect first by designing a fuzzy model based on simplified TSK-type rules, i.e., rules with a single point in the consequent, and then transforming it into a linguistic model, which has to be as accurate as the former. To do so, they introduced a secondary KB, in addition to the usual KB, and proposed an Inference Engine capable of obtaining an output result from the combined action of both Fuzzy Rule Bases. Hence, what the system really does is to allow a specific combination of antecedents to have two different consequents associated, the first and second in importance, thus avoiding some of the said problems associated to the linguistic rule structure.

Taking this idea as a starting point, we allow a specific combination of antecedents to have two consequents associated, the first and second in importance in the fuzzy input subspace, but only in those cases in which it is really necessary to improve the model accuracy in this subspace, and not in all the possible ones as in [9]. Therefore, the existence of a primary and a secondary Fuzzy Rule Base is avoided, and the number of rules in the single KB is decreased, which makes easier to interpret the model.

These double-consequent rules will locally improve the interpolative reasoning performed by the model allowing a shift of the main labels making the final output of the rule lie in an intermediate zone between the two consequent fuzzy sets. They do not constitute an inconsistency from a Linguistic Modelling point of view due to the fact that they have the following interpretation:

$$\text{IF } x_1 \text{ is } A_1 \text{ and } \ldots \text{and } x_n \text{ is } A_n \text{ THEN } y \text{ is } \textit{between } B_1 \textit{ and } B_2$$

Other advantages of our approach are that we do not need the existence of a previous TSK fuzzy model and that we work with a classical fuzzy Inference Engine. In this contribution, we shall use the Minumum t-norm in the role of conjunctive and implication operator (although any other fuzzy operator may be considered for either of the two tasks). The only restriction is to use any defuzzification method working in mode B and considering the matching degree of the rules fired. We shall work with the *Centre of Gravity weighted by the matching degree* [2], whose expression is shown as follows:

$$y_0 = \frac{\sum_{i=1}^{T} h_i \cdot y_i}{\sum_{i=1}^{T} h_i}$$

with T being the number of rules in the KB, h_i being the matching degree between the ith rule and the current system input (see Section 2) and y_i being the centre of gravity of the fuzzy set inferred from that rule.

3.2 A New Way to Generate Fuzzy Rules for Globally Improving the Cooperation Between Them

The previous point deals with the local improvement of the fuzzy reasoning accuracy in a specific fuzzy input subspace. On the other hand, the second aspect deals with the cooperation between the rules in the KB, i.e., with the overlapped space zones that are covered by different linguistic rules. As is known, the generation of the best fuzzy rule in each input subspace does not ensure that the FRBS will perform well due to the fact that the rules composing the KB may not cooperate suitably. Many times, the accuracy of the FRBS may be improved if other rules different than the primary ones are generated in some subspaces because they cooperate in a better way with their neighbour rules.

Hence, we shall consider an operation mode based on generating a preliminary fuzzy rule set composed of a large number of rules, which will be single or double-consequent ones depending on the complexity of the specific fuzzy input subspace —no rules will be generated in the subspaces where the system is not defined—. Then, all these fuzzy rules will be treated as single-consequent ones (each double-consequent rule will be decomposed in two simple rules) and the subset of them with best cooperation level will be selected in order to compose the final KB.

3.3 Some Important Remarks about ALM

We may draw two very important conclusions from the assumptions made in the previous subsections. On the one hand, it is possible that, although the preliminary fuzzy rule set generated has some double-consequent rules, the final KB does not contain any rule of this kind after the selection process. In this case, the linguistic model obtained has taken advantage of the way in which the fuzzy rules has been generated because many rule subsets with different cooperation levels have been analysed. This is why it will present a KB composed of rules cooperating well, a fact that may not happen in other inductive design methods, such us Wang and Mendel's (WM-method) [13] and the Explorative Generation Method (EGM) [4] — an adaptation of Ishibuchi et al's fuzzy classification rule generation process [6] able to deal with rules with linguistic consequents— both of which are based on directly generating the best consequent for each fuzzy input subspace.

On the other hand, it is possible that the KB obtained presents less rules than KBs generated from other methods thanks to both aspects: the existence of two rules in the same input subspace and the generation of neighbour rules with better cooperation may mean that many of the rules in the KB are unnecessary to give the final system response. These assumptions will be corroborated in view of the experiments developed in Section 5.

4 A Linguistic Modelling Process Based on ALM

Following the assumptions presented in the previous Section, any design process based on ALM will present two stages: a preliminary *linguistic rule generation method* and a *rule selection method*. The composition of both stages in the learning process presented in this contribution, which takes as a base the WM-method, is shown in the next two subsections. Another ALM process based on the EGM is to be found in [4].

4.1 The Linguistic Rule Generation Method

Let E be an input-output data set representing the behaviour of the system being modelled. Then the RB is generated by means of the following steps:

1. *Consider a fuzzy partition of the input variable spaces*: It may be obtained from the expert information —if it is availaible— or by a normalization process. In this paper, we shall work with symmetrical fuzzy partitions of triangular membership functions (see Figure 2).
2. *Generate a preliminary linguistic rule set*: This set will be formed by the rule best covering each example —input-ouput data pair— contained in E. The structure of the rule $R_l = IF\ x_1\ is\ A_1^l\ and \ldots and\ x_n\ is\ A_n^l\ THEN\ y\ is\ B_l$ generated from the example $e_l = (x_1^l, \ldots, x_n^l, y^l)$ is obtained by setting each rule variable to the linguistic label associated to the fuzzy set best covering every example component.

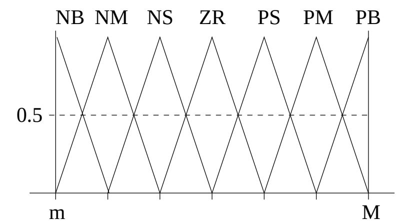

Fig. 2. Graphical representation of the type of fuzzy partition considered

3. *Give an importance degree to each rule*: The importance degree associated to R_l will be obtained as follows:

$$G(R_l) = \mu_{A_1^l}(x_1^l) \cdot \ldots \cdot \mu_{A_n^l}(x_n^l) \cdot \mu_{B_l}(y^l)$$

4. *Obtain a final RB from the preliminary linguistic rule set*: This step is the only one differing from the original WM-method. Whilst in that method the rule with the highest importance degree is the only one chosen for each combination of antecedents, in our case we allow the two most important rules in each input subspace —if they exist— to form part of the RB.
   Of course, a combination of antecedents may have no rules associated (if there are no examples in that input subspace) or only one rule (if all the examples in that subspace generated the same rule). Therefore, *the generation of double-consequent rules is only addressed when the problem complexity*, represented by the example set, *shows that it is necessary.*

4.2 The Rule Selection Genetic Process

In order to obtain a final KB composed of rules cooperating well and to achieve that more than a single rule is used only in those zones where it is really necessary, we shall use a rule selection process with the aim of selecting the best subset of rules from the initial linguistic rule set.

The selection of the subset of linguistic rules best cooperating is a combinatorial optimization problem [11]. Since the number of variables involved in it, i.e., the number of preliminary rules, may be very large, we consider an approximate algorithm to solve it, a Genetic Algorithm (GA) [5]. However, we should note that any other kind of technique can be considered without any change in ALM. Our rule selection genetic process [3] is based on a binary coded GA, in which the selection of the individuals is performed using the stochastic universal sampling procedure together with an elitist selection scheme, and the generation of the offspring population is put into effect by using the classical binary two-point crossover and uniform mutation operators.

The coding scheme generates fixed-length chromosomes. Considering the rules contained in the linguistic rule set derived from the previous step counted from 1 to T, a T-bit string $C = (c_1, ..., c_T)$ represents a subset of candidate rules to form the RB finally obtained as this stage output, B^s, such that,

$$\textit{If } c_i = 1 \textit{ then } R_i \in B^s \textit{ else } R_i \notin B^s$$

The initial population is generated by introducing a chromosome representing the complete previously obtained rule set, i.e., with all $c_i = 1$. The remaining chromosomes are selected at random.

As regards the fitness funtion, $F(C_j)$, it is based on a global error measure that determines the accuracy of the FRBS encoded in the chromosome, which depends on the cooperation level of the rules existing in the KB. We usually work with the mean square error (SE), although other measures may be used. SE over the training data set, E, is represented by the following expression:

$$F(C_j) = \frac{1}{2|E|} \sum_{e_l \in E} (y^l - S(x^l))^2$$

where $S(x^l)$ is the output value obtained from the FRBS using the RB coded in C_j, when the input variable values are $x^l = (x_1^l, \ldots, x_n^l)$, and y^l is the known desired value.

5 Examples of Application

With the aim of analysing the behaviour of the proposed ALM process, we have chosen two different applications: the fuzzy modelling of a three-dimensional function [3] and the problem of rice taste evaluation [9]. In both cases, we shall compare the accuracy of the linguistic models generated from our process with the ones designed by means of other methods with different characteristics: two methods based on generating the RB rule by rule, i.e., without considering the cooperation among linguistic rules —the one proposed by Nozaki et al. (N-method) in [9], that has been mentioned in Section 3, and the simple WM-method— and another process based on working at the level of the whole KB —NEFPROX, the Neuro-Fuzzy approach proposed in [8].

5.1 Fuzzy Modelling of a Three-dimensional Function

The expression of the selected function, the universes of discourse considered for the variables and its graphical representation are shown as follows. It is a simple unimodal function presenting two discontinuities at the points $(0,0)$ and $(1,1)$.

$$F(x_1, x_2) = 10 \cdot \frac{x_1 - x_1 x_2}{x_1 - 2x_1 x_2 + x_2},$$
$$x_1, x_2 \in [0,1], F(x_1, x_2) \in [0,10]$$

In order to model this function, a training data set composed of 674 data uniformly distributed in the three-dimensional definition space has been obtained experimentally. On the other hand, another set composed of 67 data (a ten percent of the training set size) has been randomly generated for its use as a test

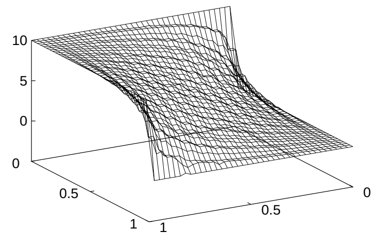

Fig. 3. Graphical representation of the function considered

set for evaluating the performance of the design methods. Of course, the latter set is only emploied to measure the generalization ability of the generated model, i.e., it is not considered in the learning stage. The DB used for all design methods is constituted by three normalised fuzzy partitions formed by *seven triangular-shaped fuzzy sets* (as shown in Fig. 2). The linguistic term set considered is $\{ES, VS, S, M, L, VL, EL\}$, standing E for Extremely, V for Very, and S, M, and L for Small, Medium and Large, respectively. Finally, the parameters considered for the rule selection genetic process are: Number of generations: 500, Population size: 61, Crossover probability: 0.6 and Mutation probability: 0.1 (per individual).

The results obtained in the experiments developed are collected in Table 1 where $\#R$ stands for the number of simple rules of the corresponding KB, and SE_{tra} and SE_{tst} for the values obtained in the SE measure computed over the training and test data sets, respectively. As may be observed, the results obtained by our process after each stage, generation and selection, are included.

Table 1. Results obtained in the fuzzy modelling of the selected function

	Generation			**Selection**		
Method	#R	EC_{tra}	EC_{tst}	#R	EC_{tra}	EC_{tst}
N-method	98	0.175382	0.061249	–	—	—
WM-method	**49**	0.194386	0.044466	–	—	—
NEFPROX	**49**	0.505725	0.272405	–	—	—
ALM	88	0.220062	0.146529	55	**0.019083**	**0.026261**

In view of these results, we should underline the good behaviour presented by our ALM process, that generates the most accurate model in the approximation of the training and test sets. As regards the number of rules in the KBs, we should note that our linguistic model only presents a few more rules than the ones generated from the WM-method and from NEFPROX. As shown in Table 2, by only adding eight new rules (and by removing two more) to the KB generated by means of the WM-method, a significantly more accurate model is obtained with a very small loss of interpretability (as mentioned, this KB only contains eight double-consequent rules). On the other hand, our model is more accurate to a high degree than the N-method one, presenting a very much simpler KB (55 rules against 98).

Table 2. Decision tables for the linguistic models obtained for the selected function by means of the WM-method (left) and our ALM process (right)

x_2

x_1	**ES**	**VS**	**S**	**M**	**L**	**VL**	**EL**
ES	*ES*	*ES*	*ES*	*ES*	*ES*	*ES*	*ES*
VS	*EL*	*M*	*S*	*VS*	*VS*	*ES*	*ES*
S	*EL*	*L*	*M*	*S*	*VS*	*VS*	*ES*
M	*EL*	*VL*	*L*	*M*	*S*	*VS*	*ES*
L	*EL*	*VL*	*VL*	*L*	*M*	*S*	*ES*
VL	*EL*	*EL*	*VL*	*VL*	*L*	*M*	*ES*
EL	*EL*	*EL*	*EL*	*EL*	*EL*	*EL*	*ES*

x_2

x_1	**ES**	**VS**	**S**	**M**	**L**	**VL**	**EL**
ES		*ES*	*ES*	*ES*	*ES*	*ES*	*ES*
VS	*EL*	*M*	*S*	*VS*	*S* *VS*	*ES*	*ES*
S	*EL*	*L*	*M*	*S*	*S* *VS*	*VS* *ES*	*ES*
M	*EL*	*VL*	*L*	*M*	*S*	*VS*	*ES*
L	*EL*	*VL* *EL*	*L* *VL*	*L*	*M*	*S* *VS*	*ES*
VL	*EL*	*EL*	*VL* *EL*	*VL*	*L* *VL*	*M*	*ES*
EL	*EL*	*EL*	*EL*	*EL*	*EL*	*EL*	

5.2 Rice taste evaluation

Subjective qualification of food taste is a very important but difficult problem. In the case of the rice taste qualification, it is usually put into effect by means of a subjective evaluation called the *sensory test*. In this test, a group of experts, usually composed of 24 persons, evaluate the rice according to a set of characteristics associated to it. These factors are: *flavor*, *appearance*, *taste*, *stickiness*, and *toughness* [9].

Because of the large quantity of relevant variables, the problem of rice taste analysis becomes very complex, thus leading to solve it by means of modelling techniques capable of obtaining a model representing the non-linear relationships existing in it. Moreover, the problem-solving goal is not only to obtain an accurate model, but to obtain a user-interpretable model as well, capable of putting some light on the reasoning process performed by the expert for evaluating a kind of rice in a specific way. Due to all these reasons, in this Section we deal with obtaining a linguistic model to solve the said problem.

In order to do so, we are going to use the data set presented in [9]. This set is composed of 105 data arrays collecting subjective evaluations of the six variables in question (the five mentioned and the overall evaluation of the kind of rice), made up by experts on this number of kinds of rice grown in Japan (for example, Sasanishiki, Akita-Komachi, etc.). The six variables are normalised, thus taking values in the real interval $[0, 1]$.

With the aim of not biasing the learning, we have randomly obtained ten different partitions of the said set, composed by 75 pieces of data in the training set —to generate ten linguistic models in each experiment— and 30 in the test one —to evaluate the performance of the generated models—. To solve the problem, we use the same Linguistic Modelling processes considered in the previous Section. The values of the parameters of the rule selection genetic process are the same ones considered in that Section as well.

As was done in [9], we have worked with normalised fuzzy partitions (see Fig. 2) composed of a different number of linguistic labels for the six variables considered —two and three, to be precise—. The results obtained in the experiments developed are collected in Table 3. The values shown in columns SE_{tra} and SE_{tst} have been computed as an average of the SE values obtained on the training and test data sets, respectively, by the ten linguistic models generated in each case. The column $\#L$ stands for the number of labels considered in the fuzzy partitions in each experiment and $\#R$ stands for the average number of linguistic rules in the KBs of the models generated from each process.

Table 3. Results obtained in the rice taste evaluation

		Generation			**Selection**		
#L	**Method**	#R	EC_{tra}	EC_{tst}	#R	EC_{tra}	EC_{tst}
2	N-method	64	0.00862	0.00985	–	—	—
	WM-method	15	0.01328	0.01311	–	—	—
	NefProx	15	0.00633	0.00568	–	—	—
	ALM	19.8	0.02192	0.02412	**5**	**0.00341**	**0.00398**
3	N-method	364.8	0.00251	0.00322	–	—	—
	WM-method	23	0.00333	0.00375	–	—	—
	NefProx	32.2	0.00338	0.00644	–	—	—
	ALM	25.7	0.00595	0.00736	**12.2**	**0.00185**	**0.00290**

From an analysis of these results, we may again note the good behaviour presented by the proposed ALM process. The linguistic models generated from it clearly outperform the ones designed by means of the other processes in the approximation of both data sets (training and test) in the two experiments developed (using 2 and 3 labels in the fuzzy partitions). On the other hand, even following the approach of double-consequent generation proposed in Section 3, our process generates the KBs with less rules, thus making the corresponding models simpler to be interpreted. In fact, none of the 20 KBs generated finally presents double-consequent rules due to the action of the selection process.

6 Concluding Remarks

In this paper, ALM has been proposed, that is a new approach to design linguistic models in the field of Intelligent Data Analysis, which are accurate to a high degree and suitably interpretable by human-beings. An ALM process has been introduced as well, and its behaviour has been compared to other Linguistic Modelling techniques in solving two different problems. The proposed process has obtained very good results.

This leads us to conclude that, as mentioned in Section 3.3, our process has the capability of distinguishing the unnecesary rules and of generating KBs with good cooperation. The ALM operation mode based on: a) generating a preliminary fuzzy rule set with a large number of rules —considering double-consequent ones if it is necessary— and b) selecting the subset of them cooperating best allows us to obtain good results in the area of Linguistic Modelling.

References

1. Bastian, A.: How to handle the flexibility of linguistic variables with applications. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **2:4** (1994) 463-484.
2. Cordón, O., Herrera, F., Peregrín, A.: Applicability of the fuzzy operators in the design of fuzzy logic controllers. Fuzzy Sets and Systems **86** (1997) 15-41.
3. Cordón, O., Herrera, F.: A three-stage evolutionary process for learning descriptive and approximative fuzzy logic controller knowledge bases from examples. International Journal of Approximate Reasoning **17:4** (1997) 369-407.
4. Cordón, O., Herrera, F.: A Proposal for Improving the Accuracy of Linguistic Modelling. Technical Report DECSAI-98113. Dept. of Computer Science and A.I. University of Granada. Spain (May, 1998).
5. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley (1989).
6. Ishibuchi, H., Nozaki, K., Tanaka, H.: Distributed representation of fuzzy rules and its application to pattern classification. Fuzzy Sets and Systems **52** (1992) 21-32.
7. Mamdani, E.H., Applications of fuzzy algorithm for control a simple dynamic plant, Proceedings of the IEE, **121:12** (1974) 1585-1588.
8. Nauck, D., Klawonn, F., Kruse, R.: Fundations of Neuro-Fuzzy Systems. John Willey & Sons (1997).
9. Nozaki, K., Ishibuchi, H., Tanaka, H.: A simple but powerful heuristic method for generating fuzzy rules from numerical data. Fuzzy Sets and Systems **86** (1997) 251-270.
10. Pedrycz, W. (Ed.): Fuzzy Modelling: Paradigms and Practice. Kluwer Academic Press (1996).
11. Sugeno, M., Yasukawa, T.: A fuzzy-logic-based approach to qualitative modelling. IEEE Transactions on Fuzzy Systems **1:1** (1993) 7-31.
12. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its application to modelling and control. IEEE Transactions on Systems, Man, and Cybernetics **15:1** (1985) 116-132.
13. Wang, L.X., Mendel, J.M.: Generating fuzzy rules by learning from examples. IEEE Transactions on Systems, Man, and Cybernetics **22** (1992) 1414-1427.
14. Zadeh, L.A.: Fuzzy sets. Information and Control **8** (1965) 338-353.