# Improving Simple Linguistic Fuzzy Models by Means of the Weighted COR Methodology

Rafael Alcalá[1], Jorge Casillas[2], Oscar Cordón[2], and Francisco Herrera[2]

[1] Department of Computer Science, University of Jaén,
E-23071 Jaén, Spain
alcala@ujaen.es

[2] Department of Computer Science and A.I., University of Granada,
E-18071 Granada, Spain
{casillas, ocordon, herrera}@Springer.de

**Abstract.** In this work we extend the Cooperative Rules learning methodology to improve simple linguistic fuzzy models, including the learning of rule weights within the rule cooperation paradigm. Considering these kinds of techniques could result in important improvements of the system accuracy, maintaining the interpretability to an acceptable level.

## 1 Introduction

One of the problems associated with Linguistic Modeling is its lack of accuracy when modeling some complex systems. It is due to the inflexibility of the concept of linguistic variable, which imposes hard restrictions to the fuzzy rule structure. To overcome this problem, many different possibilities to improve the Linguistic Modeling have been considered in the specialized literature. All of these approaches share the common idea of improving the way in which the linguistic fuzzy model performs the interpolative reasoning by inducing a better cooperation among the rules in the learned model. There are different ways to induce rule cooperation in the learning process [2, 6, 7].

In [2], a new learning methodology was proposed as a first strategy to improve *simple linguistic fuzzy models*, preserving their structure and descriptive power, and inducing a better cooperation among the fuzzy rules: the Cooperative Rules (COR) methodology. The learning philosophy was based on the use of *ad hoc data-driven methods*[3] to determine the fuzzy input subspaces where a rule should exist and a set of candidate consequents assigned to each rule. After that, a combinatorial search was carried out in the set of candidate consequents to obtain a set of rules with good cooperation among them. In [1, 3], different combinatorial search techniques were considered with this aim.

On the other hand, other technique to improve the rule cooperation is the use of weighted fuzzy rules [4, 8, 11], in which modifying the linguistic model structure an importance factor (weight) is considered for each rule. By means

[3] A family of efficient and simple methods guided by covering criteria of the data in the example set

of this technique, the way in which these rules interact with their neighbor ones could be indicated.

In this work, we propose the hybridization of both techniques to obtain weighted cooperative fuzzy rules. Thus, the system accuracy is increased while the interpretability is maintained to an acceptable level. To do that, we present the Weighted COR (WCOR) methodology, which includes the weight learning within the original COR methodology.

To learn the subset of rules with the best cooperation and the weights associated to them, different search techniques could be considered [9]. In this contribution, we will consider a Genetic Algorithm (GA) for this purpose.

*This extended methodology can be intended as a meta-method over any other ad hoc data-driven learning method, developed to improve simple linguistic fuzzy models by considering the way in which the fuzzy rules interact.* Depending on the combination of this technique with different ad hoc data-driven methods, different learning approaches arise. In this work, we will consider the Wang and Mendel's method [10] (WM) for this purpose —approach guided by examples—.

The paper is organized as follows. In the next section the said specific ways to improve the rule cooperation are introduced, reviewing the original COR methodology. In Sect. 3, the WCOR methodology to obtain weighted cooperative rules is proposed. Experimental results are shown in Sect. 4, whilst some concluding remarks are pointed out in Sect. 5.

## 2 Preliminaries

### 2.1 The COR Methodology

The COR methodology is guided by example covering criteria to obtain antecedents (fuzzy input subspaces) and candidate consequents [2]. Following the WM approach this methodology presents the following learning scheme:

Let $E = \{e_1, \ldots, e_l, \ldots, e_N\}$ be an input-output data set representing the behavior of the problem being solved —with $e_l = (x_1^l, \ldots, x_n^l, y^l)$, $l \in \{1, \ldots, N\}$, $N$ being the data set size, and $n$ being the number of input variables—. And let $\mathcal{A}_j$ be the set of linguistic terms of the $i$-th input variable —with $j \in \{1, \ldots, n\}$— and $\mathcal{B}$ be the one of the output variable.

1. *Generate a candidate linguistic rule set.* This set will be formed by the rule best covering each example (input-output data pair) contained in the input-output data set. The structure of each rule, $RC^l$, is obtained by taking a specific example, $e_l$, and setting each one of the rule variables to the linguistic label associated to the fuzzy set best covering every example component, i.e.,

   $$RC_l = \textsf{IF } X_1 \textsf{ is } A_1^l \textsf{ and } \ldots \textsf{ and } X_n \textsf{ is } A_n^l \textsf{ THEN } Y \textsf{ is } B^l,$$

   with

   $$A_j^l = arg \max_{A' \in \mathcal{A}_j} \mu_{A'}(x_j^l) \;\text{ and }\; B^l = arg \max_{B' \in \mathcal{B}} \mu_{B'}(y^l).$$

2. *Obtain the antecedents $R_i^{ant}$ of the rules composing the final linguistic model and a set of candidate consequents $C_{R_i^{ant}}$ associated to them.* Firstly, the rules are grouped according to their antecedents. Let $R_i^{ant} = \textsf{IF } X_1 \textsf{ is } A_1^i$ $\textsf{and} \ldots \textsf{and } X_n \textsf{ is } A_n^i$ be the antecedents of the rules of the $i$-th group, where $i \in \{1, \ldots, M\}$ (with $M$ being the number of groups, i.e., the number of rules finally obtained). The set of candidate consequents for the $R_i^{ant}$ antecedent combination is defined as:
$$C_{R_i^{ant}} = \{B_k \in \mathcal{B} \mid \exists e_l \ where \ \forall j \in \{1, \ldots, n\}, \forall A'_j \in \mathcal{A}_j,$$
$$\mu_{A_j^i}(x_j^l) \geq \mu_{A'_j}(x_j^l) \ and \ \forall B' \in \mathcal{B}, \mu_{B_k}(y^l) \geq \mu_{B'}(y^{l^i})\} \ .$$
3. *Perform a combinatorial search among the sets $C_{R_i^{ant}}$ looking for the combination of consequents with the best cooperation.* An improvement in the learning process consists of adding a new term to the candidate consequent set corresponding to each rule, the *null consequent* $\mathcal{N}$, such that $C_{R_i^{ant}} = C_{R_i^{ant}} \cup \mathcal{N}$, $i = 1, \ldots, M$. If this consequent is selected for a specific rule, such rule does not take part in the model finally learned.

Since the search space tackled in step 3 of the algorithm is usually large, it is necessary to use approximate search techniques. In [3] four different well-known techniques were proposed for this purpose. In this work we will consider a GA as search technique.

### 2.2 The Use of Weighted Linguistic Rules

Using rule weights [4, 8, 11] has been usually considered to improve the way in which the rules interacts, improving the accuracy of the learned model. In this way, rule weights suppose an effective extension of the conventional fuzzy reasoning system that allow the tuning of the system to be developed at the rule level [4, 8].

When weights are applied to complete rules, the corresponding weight is used to modulate the firing strength of a rule in the process of computing the defuzzified value. From human beings, it is very near to consider this weight as an importance degree associated to the rule, determining how this rule interacts with its neighbor ones. We will follow this approach, since the interpretability of the system is appropriately maintained. In addition, we will only consider weight values in $[0, 1]$ since it preserves the model readability. In this way, the use of rule weights represents an ideal framework for extended LM when we search for a trade-off between accuracy and interpretability. In order to do so, we will follow the weighted rule structure and the inference system proposed in [8]:

$$\textsf{IF } X_1 \textsf{ is } A_1 \textsf{ and} \ldots \textsf{and } X_n \textsf{ is } A_n \textsf{ THEN } Y \textsf{ is } B \ with \ [w],$$

where $X_i$ ($Y$) are the linguistic input (output) variables, $A_i$ ($B$) are the linguistic labels used in the input (output) variables, $w$ is the real-valued rule weight, and *with* is the operator modeling the weighting of a rule.

With this structure, the fuzzy reasoning must be extended. The classical approach is to infer with the FITA (First Infer, Then Aggregate) scheme and compute the defuzzified output as the following *weighted sum*:

$$y_0 = \frac{\sum_i m_i \cdot w_i \cdot P_i}{\sum_i m_i \cdot w_i},$$

with $m_i$ being the matching degree of the $i$-th rule, $w_i$ being the weight associated to the $i$-th rule, and $P_i$ being the characteristic value of the output fuzzy set corresponding to that rule. In this contribution, the center of gravity will be considered as characteristic value and the *minimum t-norm* will play the role of the implication and conjunctive operators.

A simple approximation for weighted rule learning would consist in considering an optimization technique to derive the associated weights of the previously obtained rules (e.g., by means of ad hoc data-driven methods as WM, or even COR).

# 3 The WCOR Methodology

It is clear that the said two approaches improve the accuracy of the learned model since they induce a good cooperation among rules. Moreover, they present complementary characteristics. However, due to the strong dependency between the consequent selection and the learning of the associated weights, the said two step-based technique to obtain weighted rules is not the most useful to obtain weighted cooperative rules.

Therefore, we need to include the learning of rule weights in the combinatorial search process of cooperative rules within the COR methodology. In this way, the selection of the set of consequents with the best cooperation and the learning of the weights associated to the obtained rules should be made using global criteria that jointly consider the action of both, consequents and weights.

In this section, we present the WCOR methodology to obtain weighted cooperative rules. With this aim, we include the weight derivation within the cooperative rule learning process.

## 3.1 Operation Mode

This methodology involves an extension of the original COR methodology. Therefore, WCOR consists of the following steps:

1. *Obtain the antecedents $R_i^{ant}$ of the rules composing the final linguistic model and a set of candidate consequents $C_{R_i^{ant}}$ associated to them.*
2. *Problem representation.* For each rule $R_i$ we have:

$$R_i^{ant},\ C_{R_i^{ant}},\ \text{and } w_i \in [0,1].$$

Since $R_i^{ant}$ is kept fixed, the problem will consist of determining the consequent and the weight associated to each rule. Two vectors of size $M$ (number of rules finally obtained) are defined to represent this information, $c_1$ and $c_2$, where,

$$c_1[i] = k_i \mid B_{k_i} \in C_{R_i^{ant}}, \ \textit{and}$$
$$c_2[i] = w_i, \ \forall i \in \{1, \ldots, M\},$$

except in the case of considering rule simplification, in which $B_{k_i} \in C_{R_i^{ant}} \cup \mathcal{N}$.

In this way, the $c_1$ part is an integer-valued vector in which each cell represents the index of the consequent used to build the corresponding rule. The $c_2$ part is a real-valued vector in which each cell represents the weight associated to this rule. Finally, a problem solution is represented as follows:

$$c = c_1 \ c_2$$

3. *Perform a search on the c vector, looking for the combination of consequents and weights with the best cooperation.* The main objective will be to minimize the *mean square error:*

$$\mathrm{MSE} = \frac{1}{2 \cdot N} \sum_{l=1}^{N} (F(x_1^l, \ldots, x_n^l) - y^l)^2,$$

with $F(x_1^l, \ldots, x_n^l)$ being the output inferred from the current model when the example $e_l$ is used and $y^l$ being the known desired output.

### 3.2 Genetic Algorithm Applied to the WCOR Methodology

The proposed GA performs an approximate search among the candidate consequents with the main aim of selecting the set of consequents with the best cooperation and simultaneously learning the weights associated to the obtained rules. The main characteristics of the said algorithm are presented in the following:

- *Genetic Approach* — An elitist generational GA with the Baker's stochastic universal sampling procedure.
- *Initial Pool* — The initial pool is obtained by generating a possible combination at random for the $c_1$ part of each individual in the population. And for the $c_2$ part, it is obtained with an individual having all the genes with value '1', and the remaining individuals generated at random in $[0, 1]$.
- *Crossover* — The standard two-point crossover in the $c_1$ part combined with the max-min-arithmetical crossover in the $c_2$ part. By using the max-min-arithmetical crossover, if $c_2^v = (c[1], \ldots, c[k], \ldots, c[n])$ and $c_2^w = (c'[1], \ldots, c'[k], \ldots, c'[n])$ are crossed, the next four offspring are obtained:

$$c_2^1 = ac_2^w + (1-a)c_2^v, \qquad c_2^2 = ac_2^v + (1-a)c_2^w,$$
$$c_2^3 \text{ with } c_3[k] = \min\{c[k], c'[k]\}, \qquad c_2^4 \text{ with } c_4[k] = \max\{c[k], c'[k]\},$$

with $a \in [0,1]$ being a parameter chosen by the GA designer.
In this case, eight offspring are generated by combining the two ones from the $c_1$ part (two-point crossover) with the four ones from the $c_2$ part (max-min-arithmetical crossover). The two best offspring so obtained replace the two corresponding parents in the population.

- *Mutation* — The operator considered in the $c_1$ part randomly selects a specific fuzzy subspace ($i \in \{1,\ldots,M\}$) almost containing two candidate consequents, and changes at random the current consequent $k_i$ by other consequent $k_i{}'$ such that $B_{k_i{}'} \in C_{R_i^{ant}}$ and $k_i{}' \neq k_i$. On the other hand, the selected gene in the $C_2$ part takes a value at random within the interval $[0,1]$.

## 4 Experiments

To analyze the behavior of the proposed method, we have chosen a real-world problem to estimate the length of low voltage lines for an electric company [5].

### 4.1 Problem Description

Sometimes, there is a need to measure the amount of electricity lines that an electric company owns. This measurement may be useful for several aspects such as the estimation of the maintenance costs of the network, which was the main goal in this application [5]. Since a direct measure is very difficult to obtain, the consideration of models becomes useful. In this way, the problem involves finding a model that relates the *total length of low voltage line* installed in a rural town with the *number of inhabitants* in the town and the *mean of the distances from the center of the town to the three furthest clients* in it. This model will be used to estimate the total length of line being maintained.

To do so, a sample of 495 rural nuclei has been randomly divided into two subsets, the training set with 396 elements and the test set with 99 elements, the 80% and the 20% respectively. Both data sets considered are available at *http://decsai.ugr.es/~casillas/fmlib/*.

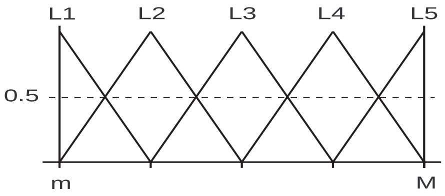

**Fig. 1.** Linguistic fuzzy partition representation

Finally, the linguistic partitions considered are comprised by *five linguistic terms* with triangular-shaped fuzzy sets giving meaning to them (see Fig. 1). The corresponding labels, $\{L_1, L_2, L_3, L_4, L_5\}$, stand for very small, small, medium, large and very large, respectively.

### 4.2 Methods

We will compare the accuracy of different linguistic models generated from our algorithm, named WCorWm [4], to the ones generated from the following methods: the well-known ad hoc data-driven WM method [10], a *method looking for the cooperation among rules* (CorWm) [2, 3] and an *algorithm for weighted rule learning* Wrl. Table 1 presents a short description of each of them.

**Table 1.** Methods considered for comparison.

| Ref. | Method | Description |
|---|---|---|
| [10] | Wm | A well-known ad hoc data-driven method |
| [3] | CorWm | GA Application to the COR methodology (WCOR $c_1$ part) |
| — | Wrl | Weighted rule learning GA on Wm and CorWm (WCOR $c_2$ part) |
| — | WCorWm | The proposed algorithm following the WCOR methodology |

The values of the parameters used in all of these experiments are presented as follows [5]: 61 individuals, 1,000 generations, 0.6 as crossover probability, 0.2 as mutation probability per chromosome, and 0.35 for the $a$ factor in the max-min-arithmetical crossover.

### 4.3 Results and Analysis

The results obtained by the analyzed methods are shown in Table 2, where $\#R$ stands for the number of rules, and $\text{MSE}_{tra}$ and $\text{MSE}_{tst}$ respectively for the error obtained over the training and test data. The best results are in boldface.

Notice that, adding weights (WRL) to the rule sets previously learned with other methods is not sufficient. It is due to the strong dependency among the learned rules and the weights associated to them. Therefore, we need to include the learning of rule weights within the rule learning process to allow an optimal behavior.

The results obtained by WCorWm improve the ones with the remaining techniques. Moreover, an appropriated balance between approximation and generalization (with and without rule simplification) has been maintained.

In the case of the simplified models, it seems that the original COR methodology removes more rules than the desired ones, achieving slight improvements

[4] With and without rule simplification

[5] With these values we have tried easy the comparisons selecting standard common parameters that work well in most cases instead of searching very specific values for each method

**Table 2.** Results obtained in the low voltage line problem.

| | | | | $2^{nd}$ stage: WRL | |
|---|---|---|---|---|---|
| **Method** | **#R** | **MSE**$_{tra}$ | **MSE**$_{tst}$ | **MSE**$_{tra}$ | **MSE**$_{tst}$ |
| WM | 13 | 298,450 | 282,029 | 242,680 | 252,483 |
| CorWM | 13 | 221,569 | 196,808 | 199,128 | 175,358 |
| WCorWM | 13 | **160,736** | 161,800 | | |
| **Considering rule simplification** | | | | | |
| CorWM | **11** | 218,675 | 196,399 | 198,630 | 176,495 |
| WCorWM | 12 | 161,414 | **161,511** | | |

in the results. The use of rule weights takes advantage of rules that at first should be removed improving the way in which they interact.

The decision tables of the models obtained by COR and WCOR are presented in Fig. 2. Each cell of the tables represents a fuzzy subspace and contains its associated output consequent, i.e., the correspondent label together with its respective rounded rule weight in the case of WCOR. These weights have been graphically showed by means of the grey colour scale, from black (1.0) to white (0.0).

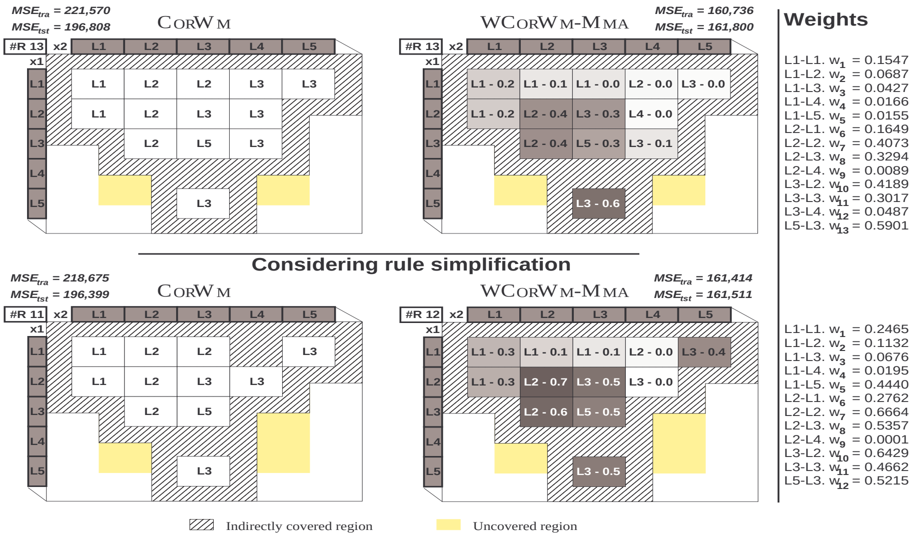

**Fig. 2.** Decision tables of the obtained models

In these tables we can observe as the use of weighted rules provokes slight changes in the consequents, improving the cooperation among the rules so obtained. Moreover, we can see as the rule in the subspace L1-L4 is maintained when an appropriate interaction level is considered.

## 5 Concluding Remarks

In this work, we present a methodology to obtain weighted cooperative rules based on the rule cooperation paradigm presented in [2]. To do that, the learning of rule weights has been included within the combinatorial search of cooperative rules. A GA to learn cooperative rules and their associated weights has been developed for this purpose.

The proposed method has been tested in a real-world problem, improving the behavior of the basic linguistic models and the ones considering cooperative rules. Moreover, an appropriated balance between approximation and generalization has been maintained by the proposed methodology.

## References

1. Alcalá, R., Casillas, J., Cordón, O., Herrera, F.: Improvement to the cooperative rules methodology by using the ant colony system algorithm. Mathware & Soft Computing **8:3** (2001) 321–335
2. Casillas, J., Cordón, O., Herrera, F.: COR: A methodology to improve ad hoc data-driven linguistic rule learning methods by inducing cooperation among rules. IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics (2002). To appear
3. Casillas, J., Cordón, O., Herrera, F.: Different approaches to induce cooperation in fuzzy linguistic models under the COR methodology. In: Bouchon-Meunier, B., Gutiérrez-Ríos, J., Magdalena, L., Yager, R.R. (Eds.): Techniques for Constructing Intelligent Systems. Springer-Verlag, Heidelberg, Germany (2002)
4. Cho, J.S., Park, D.J.: Novel fuzzy logic control based on weighting of partially inconsistent rules using neural network. Journal of Intelligent Fuzzy Systems **8** (2000) 99–110
5. Cordón, O., Herrera, F., Sánchez, L.: Solving electrical distribution problems using hybrid evolutionary data analysis techniques. Applied Intelligence **10** (1999) 5–24
6. Cordón, O., Herrera, F.: A proposal for improving the accuracy of linguistic modeling. IEEE Transactions on Fuzzy Systems **8:4** (2000) 335–344
7. Ishibuchi, H., Nozaki, K., Yamamoto, N., Tanaka, H.: Selecting fuzzy if-then rules for classification problems using genetic algorithms. IEEE Transactions on Fuzzy Systems **9:3** (1995) 260–270
8. Pal, N.R., Pal, K.: Handling of inconsistent rules with an extended model of fuzzy reasoning. Journal of Intelligent Fuzzy Systems **7** (1999) 55–73
9. Pardalos, P.M., Resende, M.G.C.: Handbook of applied optimization. Oxford University Press, NY (2002)
10. Wang, L.X., Mendel, J.M.: Generating fuzzy rules by learning from examples. IEEE Transactions on Systems, Man, and Cybernetics **22** (1992) 1414–1427
11. Yu, W., Bien, Z.: Design of fuzzy logic controller with inconsistent rule base. Journal of Intelligent Fuzzy Systems **2** (1994) 147–159