Mathware & Soft Computing 5 (1998) 333-343

New Aspects on Extraction of Fuzzy Rules using Neural Networks *

José M. Benítez, Armando Blanco, Miguel Delgado, Ignacio Requena
Dept. Ciencias de la Computación e I. A., Universidad de Granada
Avda. Andalucía, 38. 18071 - Granada. SPAIN.
e-mail: requena@decsai.ugr.es

Abstract

In previous works, we have presented two methodologies to obtain fuzzy rules in order to describe the behaviour of a system. We have used Artificial Neural Netorks (ANN) with the *Backpropagation* algorithm, and a set of examples of the system. In this work, some modifications which allow to improve the results, by means of an adaptation or refinement of the variable labels in each rule, or the extraction of local rules using distributed ANN, are showed. An interesting application on the assignement of semantic to the classes obtained in a classification without previous classes process is also included.

Keywords: Artificial Neural Networks. Learning. Fuzzy Rules. Semantic in Classification Processes.

1 Introduction.

An important problem with many applications in Artificial Intelligence is the identification and reproduction of systems. With this goal, diverse methods have been used. The most widely used method for representing knowledge, for example, to identify and reproduce the behaviour of systems, is the one based on rules. In many real problems, the use of fuzzy rules is more appropriate, due to the characteristics of the system, or because the involved information is imprecise or vague. The features of the Artificial Neural Network (ANN) make them particularly efficient for gathering the information contained in a set of data and reproducing them faithfully. Therefore the knowledge attained by an ANN may be expressed through the learning of a set of samples in the form of fuzzy rules. In previous works we have presented two procedures to obtain fuzzy rules to identify systems, using ANN.

The first one ([BEN95]) is based on a process of selection of rules, starting from all the possible rules and assigning previously to each variable a set of labels. We

*This work has been financed partly by the projects TIC96-1393-C06-04 of the CICYT, Madrid and SEDI of Caja Asturias, Oviedo.

train a ANN with the available examples of the system (reserving 30 % for test), and we form all the possible rules with all the variables, taking into account the set of labels of each variable. We check the adaptation of the rules to the trained ANN, and reject the rules that don't adapt well. On the initially valid rules, we apply a greedy algorithm that goes selecting the best rule (of those that are without select yet) in accordance with a covering coefficient on the examples, and we include it in the definitive group of rules. The process concludes when the covering of the group of rules is enough.

We have applied the procedure to the problem of a car's braking system, described in [WOL91]. We have obtained groups with same or smaller number of rules and with a similar behaviour to those obtained in [WOL91],

The second method is based on a constructive process that allows to obtain the relevant variables for each output in the system, and it doesn't need a previous set of labels for each variable. It is described with more detail in the following section.

When the number of available examples for the identification of the system is little, the behaviour of this method is not sufficiently good. So we present now some modifications that improve the identification of the system clearly (section 3).

In section 4, an interesting application in the classification processes without previous classes, to assign a concrete semantics to the obtained classes in real situations, is shwed

We also present, in section 5, a distributed methodology (local) for obtaining fuzzy rules that improves the results obtained with the previous procedure.

2 Constructive Methodology to obtain Fuzzy Rules

This method is based partly on the procedure indicated in [SES93], but considering a more appropriate representation for continue variables, using fuzzy rules instead of production rules. The labels of the variables are not previously assigned, but built to posteriori of the identification of the variables involved in each rule. We consider the data of the system like real numbers normalized in the interval [0,1] and the rules use fuzzy values (trapezoidal numbers) [1] for the variables. We suppose p variables and r classes.

The steps of the procedure, described in [BEN96], are summarized in:

a) Obtaining of a group of pre-rules

1. Train an ANN (with backpropagation), with p+r inputs, h neurons in the only hidden layer and r outputs (becuase the outputs heve been added as additional inputs).
2. For each input i and output j (all in the input of the ANN), we calculate

[1] We represent the trapezoidal fuzzy numbers with 4 parameters (a, b, c, d) where $[a, d]$ is the support, and $[b, c]$ is the core of the fuzzy number

with the weights w_{lk} (l = 1, .., p+r, k = 1, .., h)

$$S_{ij} = \sum_{k=1}^{h} (w_{ik} - w_{jk})^2$$

3. Change the original inputs of the data for its complement to one, and with these inputs (and the outputs without changing), we train by means of hebbian learning another ANN without hidden layers. The obtained weights are now V_{ij}.
4. The correlation between inputs and outputs is measured with

$$prod_{ij} = S_{ij} \cdot V_{ij}$$

For each output j, we order increasingly these products.

5. We look for an appropriate cuttoff point in the ordered products, and we consider the inputs (variables) i that are below the cuttoff point as the relevant ones for the output j. We obtain this way, the structure of the rules corresponding to the output j which we call pre-rule, and is expressed as

$$\mathrm{P}_{R_i} : X_p^i, X_q^i, X_r^i \rightarrow Y_j^i$$

b) Obtaining the rules (the labels of the variables)

1. In the training examples, we consider for each pre-rule only the values of the variables identified in the pre-rule. We make now a fuzzy classification (clustering), rejecting the examples that don't classify well. Each class (cluster) obtained will give place to an effective rule.
2. For each variable in the pre-rule, we order the data of the examples in each class, considering them as a statistical distribution.
3. The labels of the variables (trapezoidal number) included in the rule are built in the following and simple way:

   The support of the trapezoidal number comes given by the values maximum and minimum. The mode comes given by the cuartiles 1 and 3 (the median for triangular numbers) of the distribution.

This procedure is also valid when the data of the system are not real numbers, but rather they are represented by fuzzy values (see [BEN96]).

We have applied this method to the well-known problem plants IRIS. We consider 100 examples (randomly selected) as set of training, and the other 50 as test set. In the first stage we have obtained the following pre-rules:

1. *long-petal* $\rightarrow$*setosa*
2. *width-sepal*, *long-petal*, *width-petal* $\rightarrow$*versicolor*

3. *long-petal*, *width-petal* $\rightarrow$*virginica*

In order to obtain the rules, we use the method of clustering of Chiu ([CHI94]) that doesn't need the number of classes and their centers previously. Each cluster gives rise to a rule which has the structure (antecedents and output) given by the pre-rule and the antecedents fuzzy values as fuzzy trapezoidal numbers calculated out of the cluster members. In order to obtain the rules, the data are split in two parts, in a random way. The finally obtained rules have been 3 (1 for each class), with a percentage of successes (average on 10 random partitions), of 96 % in training and of 95.5 % in test. The results are good in general, classifying correctly always the type *setosa*.

Thinking that the results can improve, modifying the form of building the labels, and taking into account that in other problems, when there are few available examples, the results are not so good, we propose the modification described in the following section, that allows to obtain more effective refined rules.

3 Refinement of the rules by the construction of the labels

When the number of examples that are included in each class in the process of previous clustering is small, the building of the labels of each variable in each rule doesn't adapt enough to the reality of the problem, leading to an excessive number of errors.

To avoid this, we take into account the degree of membership from each example to the obtained classes, and we build the ordered list of values for each variable, considering that each example in the cluster has a frequency proportional to its degree of membership. This bears that when determining the cuartiles, the elements with a higher degree of membership have a higher weight, what reflects the reality of the problem better.

Also, like a sample of the examples is used for the construction of the labels, the construction of the support of the trapezoidal number by means of the values maxima and minima in the cluster can be a little compressed regarding the reality. For it, we modify the extreme values of the interval it supports, in the sense of enlarging it in a percentage of the maximum values and minima in the cluster (between 5 % and 10 %).

This modification in the construction of the labels, which behaves as a refinement of the rules obtained with the process of the previous section, has improved the results, mainly in problems with a small number of examples.

3.1 Experimental Results

In short, in the case of the plants IRIS'S problem, pointed out in the previous section, the results have been improved, reaching a 96.6% of successes in the training set, and a 96% in the test set, both in average on 10 partitions. Next table show the results on the 10 carried out executions.

Partition	Rules	$PS_{training}$	PS_{test}
0	3	94.0	100.0
1	3	97.0	98.0
2	3	96.0	98.0
3	3	99.0	90.0
4	3	96.0	94.0
5	3	98.0	94.0
6	3	95.0	100.0
7	3	97.0	96.0
8	3	97.0	96.0
9	3	97.0	94.0
average	3	96.6	96.0

We have also applied this procedure to a problem of travel agency clients classification. The data have been provided by the Centro CETT of Barcelona, and they reflect 100 examples, with 8 variables for each example (inputs), which are classified in 5 different classes (output variables).

In the process, we have considered 70 examples for training and 30 for test. When applying the process just as it was formulated originally, the procentaje of errors was between the 15 and the 20 %. When applying the modification indicated in this section, the learning was almost complete (1 example that was also in the training set, was not correctly classified), with a group of 5 rules.

4 An interesting application. To assign semantic to classes.

In the classification processes without previous classes, an interesting aspect is the one of assigning a concrete semantics to the obtained classes. In most of the examples reflected in the literature on the topic, this aspect is not of a special relevance, because what it cares fundamentally is the number of obtained classes and their respective centers.

However, in real applications, this aspect is crucial, since for the experts that must manage the obtained classes, it is fundamental to put a " name " to each class which reflects the membership from the examples to each class appropriately.

We propose the following methodology to assign a semantic to the classes obtained in a segmentation process without previous classes, to be applied in real life situations, where the opinion of the experts that later on will manage the classes has great importance.

We suppose that we have a set of examples that include the values of the variables that we will use in the classification. We don't have a previously defined group of classes. We separate the examples in a set for the learning, and another for the verification (test) of the obtained results. Then, the steps we must follow are:

1. With the available data, we carry out a segmentation without previous classes on learning examples, using an appropriate method. This is necessary, since the training of the ANN that we use in our procedure is supervised.

2. Together with the experts, the obtained segmentation is revised, using on one hand the knowledge that the experts have about the system, to analyze the intraclass homogeneity and the interclass heterogeneity, and on the other hand the set of test examples. This analysis is based fundamentally on the experience of the experts that know, of agreement with the available data, if two items can and/or should be in the same class, and if the obtained classes seem appropriate.

   Generally, the contribution of the experts is usually the necessity to add new classes that they consider are fundamental in the strategic planning of the entity.

3. With the obtained classification, we apply the constructive methodology to obtain a group of rules that describes the system built in steps 1) and 2).

4. We assign an initial semantics to each class, taking into account the following :

   - To each variable in the rules that define the class, we assign it an interval of valuation, obtained starting from the centroide of the label of the variable, so that their width to left and right picks up the form of the label.
   - Each class can be defined by one or more rules, and in each rule it can have several variables. So, the semantics will have the form:

   IF a11 < var11 < b11 AND a12 < var21 < b21 AND..... OR
   IF a21 < var21 < b21 AND a22 is valor22 AND..... OR
   . . . .
   THEN the item is in the class C1

   The variable a22 is qualitative, not numeric.

5. Again we dialogue with the experts of the system or entity, with two objectives:

   - To be sure that the experts understand well the established semantics.
   - To look together with the experts for a "language" nearer to the using that it will be made with the segmentation described with the established semantics.

6. To establish the definitive semantics and end of the process

The intervention of the experts of the system that we try to classify is fundamental in two phases of the methodology. In step 2, to be sure that the obtained classes and the assignment of the items to the classes really respond to the system. In step 5, to be sure that the form in that the classes are described is suited to the real use that will be made of that description.

Taking into account that the method to obtain fuzzy rules is valid, so much if the initial classification is crisp or fuzzy, the assignment of the semantics to the obtained classes can be carried out point for a segmentation crisp like for a fuzzy segmentation. In this case, the degree of membership of the items to the class is calculated as aggregation (using f.e. the minimum for the connectors AND and the maximum for the connectors OR) starting from a lineal assignment between the centroide (value 1) and the extremes (value 0) of the interval associated to each variable in step 4 of the described methodology.

4.1 Application to a Real Case

In the development of the investigation project SEDI, we have applied the procedure developed in the previous section, to the assignment of a semantic to the classes obtained in a process of clients' segmentation without previously established classes.

It was to verify with the indicated technology, if the segmentation that the Entity came using was ratified or not with our procedure, so much the number of considered classes, like in the assignment of the items to the different classes.

The Entity had a crisp segmentation of the clients in 8 classes. For the assignment of the items to the classes, 15 different variables were used. In the preprocessing of the data, necessary for other phases of the project, these 15 variables were summarized in 9 input variables. We had 3000 items, of those which, 600 were used for the learning, and the rest for the verification phase.

We describe the application of the process now. We have used the method of Chiu ([CHI94]) in the first step.

1. After the application of the method of Chiu, we introduce to the experts of the entity, two segmentations crisp lightly different with 6 classes each one. They were also presented a segmentation with 5 classes. These different segmentations were obtained modifying the internal parameters of the method of Chiu lightly.

2. In the I dialogue with the experts, the segmentation of 5 classes was discarded, and it was accepted in basic way, one of those of 6 classes. Although our system didn't consider them as differentiated classes, the experts added 2 classes more than they considered important, in accordance with the strategic planning of the Entity.

3. The application of the constructive method to obtain the fuzzy rules allowed us to obtain a group of 10 rules, 1 for class, except 2 class with 2 rules each one. As an example , the rule obtained for one of the classes was:

IF var1 is (0, 0, 0, 0) AND var3 is (.45, .45, .46, .47) And var7 is (0, 0, 0.01, 0.1) THEN C4

4. The descriptions of the classes were already obtained in the form signal anterioremente. As an example, the initial semantics assigned to the class C4 was:

   The initial semantics obtained for this class was:

   IF var1 aprox. 0 and var3 $<$ 17 And var 7 $<$ 900 THEN C4

5. In the dialogue with the experts of the Entity, we could observe, since it was the main objective of the experiment that the obtained classes, with their semantic ones respective they agreed basically with the initial classes that the Entity managed, taking into account the two classes added by strategic planning of the own entity.

   There were differences, important in some cases, in the composition of the classes, that is to say, in the assignment of items to some of the classes. This made that the semantic associated to the classes, valid basically, for the segmentation obtained with our procedure, was not completely valid for 3 classes of the segmentation of the Entity.

6. The semantic ones settled down definitive for the segmentation obtained with this procedure, and their understanding was ratified with experts that had not participated in the process of semantics assignment to the class.

5 A distributed procedure for fuzzy rules extraction

We think that better results can be obtained making a local construction of the rules. First, the most important areas in the behaviour of the system are identified, using a previous clustering, and then fuzzy rules are obtained for each area, as if they were different systems. It would be, in some way, a distributed process.

We can summarize the process in the following steps:

1. *Identify the important classes in the system behaviour.* This is made with a procedure of fuzzy clustering (without previous classes), on the input variables.

2. *Obtain subsets of examples.* Fixed a degree of membership α, a subset of examples is obtained for each cluster obtained in the preceding step (those that have a degree of membership to the cluster great or equal than α).

3. *Local extraction of the rules.* We apply to each subset of examples the method to obtain the rules proposed in section 3. We obtain a group of rules for each subset.

4. *Build the rule base.* The final rule base is obtained by the union of the local groups of rules obtained in the previous step.

Most relevant areas in input space have to be detected. This is to be done with an unsupervised classification method, which would only use input parts from the data. It should not be a crisp partition, but neighbor areas should influence one another. So a fuzzy classification is required. A fuzzy clustering method should be employed.

In every main area detected, a rule extraction process is applied. The classification process assigns every sample a membership degree with respect to every area. To speed up the computing, we can concentrate the efforts on working only with the most relevant to the area data. Hence an α value is chosen and the set to work on is reduced to the corresponding α-cut. The choice of α will be a trade-off between getting rules with higher influence from neighborhoods and the efficiency of the process.

The application of the method to a reduced data set results in a shorter running time. Besides, since there are no dependencies among them, they all can proceed in parallel, so an even higher improvement in the overall efficiency may be attained.

As an additional advantage, if the method applied locally performs feature selection, the obtained rules may be simpler, that is, they might need a smaller number of variables.

The rule base resulting from the method is built by joining local rule bases.

To test the given methodology several experiments were conducted. The procedure has been applied to two well-known problems: the PIMA data set and the IRIS data set.

The PIMA data set is reputed as a rather hard to learn problem. Best results report a performance of 76% on test sets. The objective is diagnosing diabetes of Pima indians. Based on personal data and the results of medical examinations, try to decide whether a Pima indian individual is diabetes positive or not. The 512 samples are composed of 8 input attributes and 2 outputs. The data set was randomly partitioned into two parts: train and test sets with 400 and 112 samples, respectively.

The Chiu algorithm [CHI94] was applied on the train data yielding two centers. The membership to the i-th fuzzy cluster is obtained as in the FCM algorithm. The two fuzzy clusters were α-cut with $\alpha = 0.5$ giving two sets with 209 and 191 samples, respectively.

Then, the rule extraction method described in previous sections, was applied to both subsets. The process produced 2 and 3 rules respectively. The performance of the final rule base with 5 rules was checked on the 112 test samples giving a performance of 76%. This result is slightly better than the one achieved using a set of rules built globally (without previous clustering).

The overall performance does not beat results obtained with other techniques, but the remarkable point is that it has been reached by using a very small number of rules, namely 5.

The distributed fuzzy rule extraction procedure has also been applied to the IRIS data set. Applying the rule extraction method without previous clustering to the problem, total success is achieved in both the train and test sample sets. When using the distributed approach, four clusters are obtained, which α-cut produce sets with 36, 18, 16, and 36 samples, respectively. The global results reach now a 97% hit ratio. This is due to the small cardinal of partial training sets. Moreover, because of the way labels are built, a perfect generalization is not attained when employing all the rules in a single base. Notwithstanding, in the pima problem (where clusters were big enough) the distributed approach does improve results.

6 Conclusions and final comments.

We have presented a procedure to obtain fuzzy rules, determining in first place, the relevant variables for each output of the system. In second place, the labels of each variable are built in each fuzzy rule, since a previous set of labels for each variable is not needed. The results are good in general, but decay if the number of available examples is small.

A refinement in the process to build the labels, by using the available information of the example membership degrees to each class obtained in the classification process to obtain the definitive rules, is presented. That allows sensitive improvements.

We point out an important application to the problem of assigning semantics to the classes obtained in a segmentation process without previous classes, and we have indicated it application to a real situation.

We have also presented a distributed procedure to obtain a fuzzy rule base, that improves the results in cases where enough examples are avalaible.

Finally, we think that the results could improve, mainly in the phase of building the fuzzy labels. The ideas go in two addresses. On one hand, to design an algorithm that allows us to obtain suitable points alternative to the cuartiles, to obtain the mode of the fuzzy numbers. On the other hand, to improve the construction of the labels using the knowledge from the negative examples to each rule.

References

[BEN95] Benítez, J.M., Blanco, A., and Requena, I. *An Empirical Procedure to Obtain Fuzzy Rules using Neural Networks.* 6th IFSA Congress. Brasil, July, (1995).

[BEN96] Benítez, J.M., Blanco, A., Delgado, M. and Requena, I. *Neural Methods for Obtaining Fuzzy Rules.* Mathware, vol III, n. 3, pp 371-382. (1996)

[CHI94] Chiu, S.L.*A cluster estimation method with extension to Fuzzy Model Identification.* IEEE Intern. Congress on Fuzzy Systems. pp.1240–1245. (1994).

[LEE90] LEE, C.C. *Fuzzy Logic in Control Systems: Fuzzy Logic Controller.* IEEE Trans. on Systems, Man, and Cybernetics. **20**(2), Part I, 404–419, Part II, 419–435. (1990).

[REQ92] REQUENA, I. Ph. Doctor Dissertation: *Redes Neuronales en Problemas de Decisión con Ambiente Difuso.* Depto. Ciencias de la Computación e I.A. Universidad de Granada. (1992).

[REQ94] REQUENA, I., DELGADO, M., AND VERDEGAY, J.L. *Automatic Ranking of Fuzzy Numbers with the Criteria of a Decision-maker Learnt by an Artificial Neural Network.* Fuzzy Sets and Systems, **64**, 1–19. (1994).

[REQ95] REQUENA, I., BLANCO, A., DELGADO, M., AND VERDEGAY, J.L. *A Decision Personal Index to Fuzzy Numbers based on Artificial Neural Networks.* Fuzzy Sets and Systems, 73, pp 185-199. (1995)

[SES93] SESTITO S. AND DILLON, T. *Knowledge Acquisition of Conjunctive Rules Using Multilayered Neural Networks.* Inter. J. Intelligent Systems, 8, 779–805. (1993).

[WOL91] WOLF, T. *Das Fuzzy-Mobil.* mc 3/91, pp. 50–63. (1991).