Automatic Fuzzy Rule Base Generation from Data

I.Rojas, H.Pomares, J.Gonzalez, M.Salmeron, J.L. Bernier

Department of Computer Architecture and Technology.

University of Granada. Spain.

Abstract

In the synthesis of a fuzzy system, two steps are generally employ: the identification of a structure and the optimization of the parameters defining it. The identification of the fuzzy system structure (number of rules and membership functions in each input variable) and the optimization of the parameters defining it are performed jointly. Starting from an initially simple fuzzy system, the numbers of membership functions in the input domain and of rules are adapted in order to reduce the approximation error.

Keywords: fuzzy, rule-base generation, automatic learning

Tópicos:

Aprendizaje Automático, Descubrimiento de Conocimiento y Minería de Datos
Ingeniería del Conocimiento: Ontologías, Adquisición, Representación, Reutilización y Compartición

PAPER TRACK

Automatic Fuzzy Rule Base Generation from Data

I.Rojas, H.Pomares, J.Gonzalez, M.Salmeron, J.L. Bernier

Department of Computer Architecture and Technology.

University of Granada. Spain.

Abstract

In the synthesis of a fuzzy system, two steps are generally employ: the identification of a structure and the optimization of the parameters defining it. The identification of the fuzzy system structure (number of rules and membership functions in each input variable) and the optimization of the parameters defining it are performed jointly. Starting from an initially simple fuzzy system, the numbers of membership functions in the input domain and of rules are adapted in order to reduce the approximation error.

1. Introduction

To synthesize a fuzzy system, two different phases must be carried out: firstly, it is necessary to establish the structure or topology of the fuzzy system to be used. Secondly, the parameters defining the fuzzy system must be established [1],[5]. The specification of the topology of the fuzzy system consists of determining the number of fuzzy rules to be constructed and the number of membership functions in the input and output spaces. It is also necessary to determine the fuzzy inference method to be used. To overcome the problem of knowledge acquisition, many fuzzy systems that automatically derive fuzzy if-then rules from sample data have been proposed in the bibliography [2],[3],[4],[6],[7]. In this paper, we propose a general learning method as a framework to obtain the optimal structure of a fuzzy system (the number of membership functions for each variable and rules of the system), and to automatically extract the parameters that define the membership function and the fuzzy IF-THEN rules, using the information available as desired I/O pairs.

2. Self-organized fuzzy system (sofs) generation

We assume that the fuzzy system considered is a multi-input single output (MISO) system, since the extension of the method to a multiple-output problem is straightforward. Thus we consider functions which have an m-dimensional input vector and a one-dimensional output. The fuzzy If-Then rules considered have the following form:

$$\widetilde{R}_{i_1 i_2 \ldots i_N} : IF\ x_1 \text{ is } \mathrm{A}_1^{i_1}\ \&\ldots\ THEN\ \mathrm{z} = \mathrm{R}_{i_1 i_2 \ldots i_N}$$

In this paper, the fuzzy inference method uses the product as T-norm and the centroid method with sum-product operator as the defuzzification strategy.

Initiation of a simple fuzzy system model. The procedure starts with a simple system comprising a small rule set. In the present paper, we start with three membership functions per input variable, corresponding to its minimum, central, and maximum values respectively.

Construction of the membership functions. We will use a Triangular Partition (TP), where only the centres of the membership functions will be stored, since the slopes of the triangles are calculated according to the centres of the surrounding membership functions.

Determination of the conclusions of the fuzzy rules. To construct the consequence of each rule, all the input vectors are taken into account. For each fuzzy subspace defined in the input domain, it is necessary to calculate which fuzzy rules relate both input and output domains. The probability that the fuzzy conclusion for a given rule $\widetilde{R}_{i_1 i_2 \ldots i_N}$, defined in the input space by the membership functions: $\mathrm{A}_1^{i_1}, \mathrm{A}_2^{i_2}, \ldots, \mathrm{A}_N^{i_N}$ will be the membership function Z^q, can be expressed using the Bayes Theorem as:

$$P(Y^{z(p)} / X_1^{s(p,1)}, \ldots, X_m^{s(p,m)}) = \frac{P(Y^{z(p)} \cap (X_1^{s(p,1)}, \ldots, X_m^{s(p,m)}))}{P(X_1^{s(p,1)}, \ldots, X_m^{s(p,m)})}$$

Thus, a distribution is obtained whose elements provide information about the construction of the consequence of each fuzzy rule. However, instead of determining a unique scalar value for a given input subset in order to construct a fuzzy rule, as is usually done, the proposed procedure provides a set of values $M(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N})$ for this input subset.

$$M(X_1^{s(p,1)} ,...) = \{ \sum_{z(p)=1}^{q} P(Y^{z(p)} / X_1^{s(p,1)} ,..., X_m^{s(p,m)}) \\ / \ Rule : R_p (X_1^{s(p,1)} ,..., X_m^{s(p,m)} \rightarrow Y^{z(p)}) \}$$

Although the consequence of a rule $\widetilde{R}_{i_1 i_2 \ldots i_N}$ can a priori be assigned to different values of the $M(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N})$ it is necessary to determine the most representative element of this fuzzy set. In this paper, the Centre Of Area (COA) defuzzifier has been used.

$$Z_{defuzz}^{q} = \frac{\sum_{q=1}^{n_z} Core(Z^q)\, \mu(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N} ; Z^q)}{\sum_{q=1}^{n_z} \mu(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N} ; Z^q)}$$

where $(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N}; Z^q)$ is the degree of membership of the fuzzy set $M(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N})$, defined by the conditional probability: $(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N}; Z^q) =$ $P(Z^q / A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N})$. $Core(Z^q)$ represents the element where the membership function Z^q reaches the maximum value (if there were more than one element, the average would be calculated).

Computation of the Controversy Index for the fuzzy rules. With the numerical element obtained from the defuzzification operator, Z_{defuzz}^{q}, the rule-base of the fuzzy system is determined completely. However the defuzzification procedure does not take into account that there may exist elements in fuzzy set $M(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N})$, with a high degree of membership, but with a value distant from the representative element selected by the defuzzification process. Additional information is required about the structure of the fuzzy set $M(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N})$. This information will be provided through the definition of an index, termed a Controversy Construction Index (CCI) of a fuzzy rule. This index is computed by the following expression for the fuzzy set $M(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N})$:

$$M(X_1^{s(p,1)} ,...) = \{ \sum_{z(p)=1}^{q} P(Y^{z(p)} / X_1^{s(p,1)} , \\ ...) / \ Rule : R_p (X_1^{s(p,1)} ,..., X_m^{s(p,m)} \rightarrow Y^{z(p)}) \}$$

This measure reflects the discrepancy between the element selected as representative by the operation of a defuzzifier method in a generic fuzzy set, and the remaining elements of this set. The greater CCI is, the more difficult it is to obtain an element that adequately represents the fuzzy set $M(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N})$, and thus with greater certainty it is possible to assert that there exists controversy in the choice of the conclusion of the fuzzy rule $\widetilde{R}_{i_1 i_2 \ldots i_N}$.

Adjust the input membership functions. We define an index that associates with each membership function of each variable, its degree of responsibility in the controversy of the rules. This new index, termed the Sum of Controversies associated with a membership function (SC), relates the membership function *j* of variable *v* to the sum of the controversies of all the rules whose antecedent in variable *v* refers to membership function *j*:

$$SC(A_v^j) = \sum_{rules / i_v = j} CCI(M(A_1^{i_1}, A_2^{i_2}, ..., A_N^{i_N}))$$

The distances between the membership functions are altered with the information stored in SC, according to the following expression:

$$d_v^j = \left(c_v^{j+1} - c_v^j\right)\left(1 - \alpha \frac{SC(A_v^j) + SC(A_v^{j+1})}{2 \cdot \sum_{j=1}^{n_v} SC(A_v^j)}\right)$$

where " is a parameter that quantifies the speed of the iterative algorithm for the alteration of the membership functions.

In this way, the distance that separates two membership functions within each variable is compressed according to the degree of controversy associated with the membership functions that identify such a distance. Thus, the greater the value of the SCs of the membership functions, the shorter the distance between the neighbouring membership functions for such a value (i.e greater is the compression of this distance), thus providing an increase in the density of fuzzy rules in this zone. Indeed, in high controversy regions for the choice of the consequences of the rules, it is necessary to reduce the distance between the membership functions, such that their support is smaller and the rule density in high controversy areas is increased.

Finally, to ensure that the domain of each input variable remains unaltered, it is necessary to readjust the distances thus obtained to maintain the new relation between them. For this purpose, the new centres are given by:

$$c_v^{j+1} = c_v^j + r_v \frac{d_v^j}{\sum_{j=1}^{n_v - 1} d_v^j}$$

where r_v is the range of variable *v*.

Obtain a new structure for the fuzzy system: increase the number of membership functions for specific input variable(s). To perform an overall analysis, we consider the distribution of Controversy Construction Index (CCI) associated with each of the membership functions of each of the input variables. Therefore, it is necessary to:

a) Determine which variable (or variables) should be incremented in the number of membership functions. To do this, we define:

$$\hat{CCI}(A_v^j) = \frac{CCI(A_v^j)}{\prod_{\substack{m=1 \\ m \neq v}}^{N} n_m}$$

The variance associated with the distribution of $\hat{CCI}(A_v^j)$ for variable *v* is given by:

$$\sigma_v^2 = \frac{\sum_{j=1}^{n_v} \left(\hat{CCI}(A_v^j) - \overline{\hat{CCI}(A_v^j)} \right)^2}{n_v},$$

$$M_{\sigma^2} = \underset{v}{Max}\left(\sigma_v^2\right)$$

$$\begin{cases} if \ (\mathrm{T}_{IMF}[j] > \rho M_{\mathrm{T}}) \ \ \xi_j = 1 \\ \\ \qquad else \ \ \xi_j = 0 \end{cases}$$

To obtain the variables which would add a new membership function, we apply a threshold function. The parameter ρ is a threshold defined in the interval [0,1]. The higher the parameter, the fewer the variables that are taken into account to increase the number of membership functions in each topology obtained by the algorithm.

b) Determine where to locate the new membership function in the selected variable to increase the number of linguistic values.

Thus, if v =1 then

1.- Calculate the centre of the new membership function c_v^* using the COA.

2.- nv =nv +1

3.- Update the new distribution of membership functions for variable v.

Stop condition. If the error of approximation reaches a threshold or if the number of membership functions in an input variable exceeds a certain limit, the algorithm finishes.

3. Experimental analysis

To demonstrate the usefulness of the algorithms and to gain an insight into the effects of the automatic distribution of the membership functions, an example of functional approximation is described in this section. For easy display, a two-input variable function was considered which is given by:

$$F(X_1, X_2) = 10 * \frac{(X_2 - 5)^2 + 10}{10 + (X_1 - 5)^2 + (X_2 - 5)^2}$$

A uniform spiral distribution was used to produce 400 samples for the two-variable function. The input and output domains are [0,10]. Just only 36 rules are needed to obtain an error indexes of MSE (Mean Squared Error) =0.012 ME= 0.101 Maximum Error= -0.39 (Fig.2). Because different configuration, in term of number of rules, are obtained by the algorithm presented. Fig.3 shows the evolution of the MSE for the different fuzzy systems.

Fig.1 Original output surface

Fig.2 Output surface obtained with 42 rules (configuration of 7x6 membership functions)

Fig.3 Evolution of the MSE. The structures are: 3x3, 4x4, 5x5, 6x5, 7x6, 8x7, 9x8, 10x9, 11x10, 12x11

As may be seen in the development of the number of rules of this example (Fig.3), variable X_1 needs a greater contribution of membership functions, since it modifies the output surface with more significance than variable X_2.

4. Conclusion

We presented a systematic approach to dealing with the problem of self-generating fuzzy rule-table. Two fundamental issues, fuzzy system structure identification and fuzzy parameters optimization, were the main concerns in this paper. A two-step strategy was suggested. First, the fuzzy rules and the membership function locations for a defined structure are jointly optimized. Second, the structure of the fuzzy system is optimized. The simulation results on the examples taking from function approximation show that the proposed approach is feasible and useful in dealing with problems where fast construction of a simple and accurate fuzzy model is required, without the help of a human expert.

References

[1] C.M.Higgins, R.M.Goodman "Fuzzy rule-based networks for control", *IEEE Trans.on Fuzzy Systems*, vol.2, no.1, February, pp.82-88, 1994.

[2] H.Ishibuchi, R.Fujioka, H.Tanaka, "Neural Networks That Learn from Fuzzy If-Then Rules", *IEEE Trans.on Fuzzy Systems*, vol.1, No.2, May 1993, pp.85-96.

[3] I.Rojas, J.J.Merelo, J.L.Bernier, A.Prieto: "A new approach to fuzzy controller designing and coding via genetic algorithms", *IEEE International Conference on Fuzzy Systems*, Barcelona, pp.1505-1510, July 1997.

[4] I.Rojas, H.Pomares, F.J. Pelayo, M.Anguita, E.Ros, A.Prieto, "New methodology for the development of adaptive and self-learning fuzzy controllers in real time", *International Journal of Approximate Reasoning*. vol. 21, no 2, pp. 109-136, 1999.

[5] H.Pomares, I.Rojas, J.Ortega, J.Gonzalez, A.Prieto, "A systematic approach to a self-generating fuzzy rule-table for function approximation", *IEEE Trans. on SMC-Part B, vol,30, no.3, June, 2000.*

[6] I.Rojas, H.Pomares, J.Ortega, A.Prieto, "Self-Organized fuzzy system generation from training examples", *IEEE Trans. on Fuzzy Systems*, vol.8, n.1, pp.23-36, 2000

[7] R.Rovatti, R.Guerrieri, "Fuzzy sets of rules for system identification", *IEEE Trans.on Fuzzy Systems*, vol.4, no.2, pp.89-102, 1996.