Fuzzy Rules Extraction Using Self-Organising Neural Network and Association Rules

Kok Wai Wong, Tamás D. Gedeon, Chun Che Fung, and Patrick M. Wong

***Abstract*— Fuzzy logic is becoming popular in dealing with data analysis problems that are normally handled by statistical approaches or ANNs. The major limitation is the difficulty in building the fuzzy rules from a given set of input-output data. This paper proposed a technique to extract fuzzy rules directly from input-output pairs. It uses a self-organising neural network and association rules to construct the fuzzy rule base. The self-organising neural network is first used to classify the output data by realising the probability distribution of the output space. Association rules are then used to find the relationships between the input space and the output classification, which are subsequently converted to fuzzy rules. This technique is fast and efficient. The results of an illustrative example show that the fuzzy rules extracted are promising and useful for domain experts.**

***Index Terms*— Association Rules, Fuzzy rules, Rule extraction, Self-organizing neural network**

I. Introduction

Fuzzy logic (FL) is becoming popular in dealing with data analysis problems that are normally handled by statistical approaches or ANNs [1]. In general, fuzzy control systems are the most important applications of fuzzy theory. This is a generalised form of expert control using fuzzy sets with fuzzy rules for modelling a system. In classical fuzzy approaches from Zadeh [2] and Mamdani [3], the basic idea is to calculate the conclusion by evaluating the degree of matches from the observation that triggered one or several rules in the model. However, conventional FL data analysis systems do not have any learning algorithms to build the analysis model. Rather, they make use of human knowledge, past experience or detailed analysis of the available data by other means in order to build the fuzzy rules for the data analysis. The advantages of using FL are the ability to handle imprecise data and to interpret the analysis model built. The data analysis model can also incorporate human knowledge by modifying the fuzzy rule base. The data analysis model can also be changed easily by modifying the fuzzy rule base. The major limitation is the difficulty in building the fuzzy rules.

However, if fuzzy rules can be extracted from measured input-output data, it will be very useful. This area has attracted much attention these few years [4, 5, 6, 7]. The usual fuzzy controller, which normally puts emphasis on the rule premises, generates a large fuzzy rule base. In a fuzzy rule base, the number of rules depends very much on the number of inputs and the number of partitions. Also the rule extraction time can be very time consuming if the input space is very large.

In this paper, we propose an integrated use of a self-organising neural network and association rules for building the fuzzy rule base. Our emphasis is placed on the rule consequents, and we then find an association concerning the premises. In this case, the partitioning of the input space is not ordinary as shown in Fig. 1, thus the number of fuzzy rules does not increase exponentially with the number of inputs. The propose technique is fast and efficient in generating a fuzzy rule base.

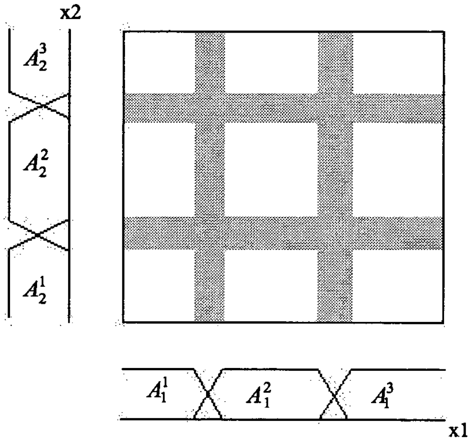

Fig. 1. Ordinary partition of input space.

K. W. Wong and T. D. Gedeon are with the School of Information Technology, Murdoch University, Western Australia 6150. (telephone: 618-9360-6599/-6430, e-mail: { kwong | tgedeon }@murdoch.edu.au).

C. C. Fung is with the School of Electrical and Computer Engineering, Curtin University, Western Australia 6102. (telephone: 618-9266-2575, e-mail: tfungcc@cc.curtin.edu.au).

P. M. Wong is with the School of Petroleum Engineering, University of New South Wales, Sydney, Australia. (telephone: 612-9385-5189, e-mail: pm.wong@unsw.edu.au).

0-7803-7101-1/ 01 / $10.00 © 2001 IEEE.

Authorized licensed use limited to: Murdoch University. Downloaded on October 30, 2009 at 03:03 from IEEE Xplore. Restrictions apply.

II. Self-organizing Neural Network

Self-organising neural networks are designed with the intention to simulate the various organisations found in various brain structures and has a relationship to brain maps [8,9]. Its main feature is the ability to visualise high dimensional input spaces onto a smaller dimensional display, usually one or two-dimensional. Self-organizing neural networks, as the name suggests, have the ability to learn and organize information without being given correct outputs for the inputs. Therefore, self-organising neural networks perform unsupervised learning as shown in Fig. 2. Let the input data space $\mathfrak{R}$ be mapped by the self-organising neural network onto a one-dimensional array with i nodes. Associated with each i node is a parametric reference vector $m_i=[\mu_{i1},\mu_{i2},\ldots,\mu_{i2}]^T \in \mathfrak{R}$, where μ_{ij} is the connection weights between node i and input j. Therefore, the input data space $\mathfrak{R}$ consisting of input vector $X=[x_1,x_2,..,x_n]^T$, i.e. $X \in \mathfrak{R}$, can be visualised as being connected to all nodes in parallel via a scalar weights μ_{ij}. The aim of the learning is to map all the n input vectors X_n onto m_i by adjusting weights μ_{ij} such that the self-organising neural network gives the best match response locations.

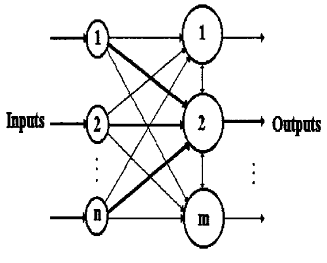

Fig. 2. Self-organising Neural Network.

A self-organising neural network can also be said to be a nonlinear projection of the probability density function $p(X)$ of the high dimensional input vector space onto the one or two-dimensional display map. Normally, to find the best matching node i, the input vector X is compared to all reference vectors m_i by searching for the smallest Euclidean distances $\| X - m_i \|$ signified by e. Therefore,

$$e = \arg\min_i \{ \| X - m_i \| \} \qquad (1)$$

or

$$\| X - m_e \| = \min_i \{ \| X - m_i \| \} \qquad (2)$$

During the learning process, the node that best matches the input vector X is allowed to learn as well as those nodes that are close to the node up to a certain distance. The learning process is expressed as:

$$m_i(t+1) = m_i(t) + h_{ei}(t)[X(t) - m_i(t)] \qquad (3)$$

where t is discrete time coordinate
and $h_{ei}(t)$ is the neighbourhood function

After the learning process has converged, the map will display the probability density function $p(X)$ that best describes all the input vectors. At the end of the learning process, an average quantisation error of the map will be generated to indicate how well the map matches the entire input vectors X_n. The average quantisation error is defined as:

$$E = \int \| X - m_e \|^2 \; p(X) dX \qquad (4)$$

Beside the average quantisation error, an individual quantisation error is also used to measure how well the input vector matches the closest node i.

III. Association Rules

Since data mining is a process of discovering useful and interesting knowledge [10], this paper explores the application of association rules [11] in finding the relationship between the input and output space. In other words, we proposed to use an association model to discover knowledge that exists in the relationship between input variables for the inference of the output variable. Such knowledge will be used for extracting the fuzzy rules that best describe the relationship between the input and output.

An association rule can be expressed as $X \Rightarrow Y$, where X and Y are set of items $I = \{i_1, i_2, \ldots, i_n\}$, such that $X \subset I$, $Y \subset I$, and $X \cap Y \neq \emptyset$. Each association rule $X \Rightarrow Y$ has a confidence, c, if $c\%$ of transactions in a database that contain X also contain Y. The support, s, of a rule $X \Rightarrow Y$ means $s\%$ of transactions in the database contain $X \cup Y$. Association rules can provide useful information about a database that cannot be done by using SQL.

However, the algorithm for finding the association rules may be time consuming if it requires multiple passes over the available data [10]. In this paper we make use of a faster algorithm [12], which uses only two full scans to find the association rules. As this algorithm is normally used to find association rules from categorical attributes, it is modified with the concept in [13] to use for quantitative attributes for this fuzzy rules extraction technique.

IV. Fuzzy Rules Extraction

A. Clustering Output Space

In a given data set, with k inputs, the given input-output data pairs with n patterns are:

$$(x_1^i, x_2^i, x_3^i, \ldots, x_k^i; y^i) \qquad (5)$$

where $i = 1, 2, 3, \ldots n$

Authorized licensed use limited to: Murdoch University. Downloaded on October 30, 2009 at 03:03 from IEEE Xplore. Restrictions apply.

For clustering using a self-organising neural network, we only make use of the output data, y^i where $i = 1, 2, 3, \dots n$. As only one variable (y^i) is used, the learning process is very fast. After the learning process, the probability density function *p(Y)* is realised and mapped onto the output neurons. In addition, the connection weights will be organized such that topologically close nodes are sensitive to y^i that are similar. For our application, only one dimensional output projection is used. The assumption made here is one output neuron represents one cluster.

After all the unsupervised learning of the self-organising neural network has been completed, the min-max values of the data in the cluster c are used as the boundary values of the class:

$$\min_c = lowest(y_c^I) \qquad (6)$$

where $I=1, 2, 3 \dots i_c$
and i_c = the number of data in c cluster

$$\max_c = highest(y_c^I) \qquad (7)$$

B. Finding Association in Input Space

Before the information from the output cluster can be used in this stage, they need to be converted to a label B:

$$y_c^I \Rightarrow B_c \qquad (8)$$

In this case, equation (5) can be re-written as:

$$[x_1^i, x_2^i, x_3^i, \dots, x_k^i; B_c(y^i)] \qquad (9)$$

where $i = 1, 2, 3, \dots n$

The first step of searching association rules is to find the appropriate partitions and the number of intervals. The equi-depth partitioning is used, as it is optimal for partial completeness [13]. The number of intervals in the partition is determined as follows.

The properties that use to decide the number of intervals are based on the rule of partial completeness level, K. It is known that the minimum confidence must be set to $1/K$ times the desired level to guarantee that close rules will be generated. The number of intervals required can be calculated by the formula:

$$No.\ of\ Intervals = \frac{2k}{m(K-1)} \qquad (10)$$

where k is the number of quantitative attributes,
m is the minimum support (as a fraction) specified by the user,
and K is the partial completeness level.

After the partitions have been identified, frequent itemsets are searched for. The frequent itemsets describe the combination of items that have transaction support above the minimum support specified by the user as shown in Fig. 3. From these frequent itemsets, association rules can be generated.

Itemset	Support
{ <Age: 20..29> }	3
{ <Age: 30..39> }	2
{ <Married: Yes> }	3
{ <Married: No> }	2
{ <NumCars: 0..1> }	3
{ <Age: 30..39> }, { <Married: Yes> }	2

Fig. 3. Example of frequent itemsets

C. Determining Fuzzy Memberships

Until this stage, all the clusters we have obtained are crisp clusters with only upper U and lower L values, and association rules that look like the following.

<Temp: 35..39> and <Wind speed: 0..10> => <Weather: hot>

In this stage, all the crisp clusters have to be converted to fuzzy membership sets. Only triangular membership functions are used as a start because they are less computationally complex and are used widely in real-time implementations. In most fuzzy inference system with fuzziness crossing at 0.5 á–cut, the values cutting below 0.5 become less and less true in that fuzzy set. Here, we propose the conversion as follows. Refer to Fig. 4 for notations used.

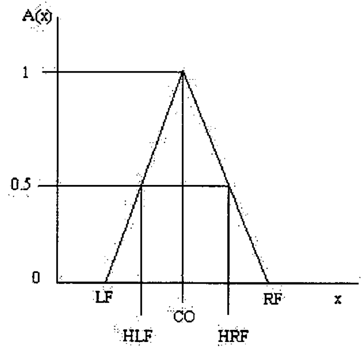

Fig. 4. Notations used in the fuzzy set

The lower boundary value of the cluster is assigned to the point where the left flank cuts at 0.5 á–cut, *HLF*:

$$HLF = L \qquad (11)$$

Authorized licensed use limited to: Murdoch University. Downloaded on October 30, 2009 at 03:03 from IEEE Xplore. Restrictions apply.

It is the same for the right flank of the fuzzy membership function. The point where á–cut is 0.5 *HRF* is assigned to the upper boundary value of the cluster.

$$HRF = U \qquad (12)$$

The core of the fuzzy membership function is taken as the average of the upper and the lower boundary values:

$$CO = \frac{HLF + HRF}{2} \qquad (13)$$

Then the left and right flanks of the fuzzy membership function finishes at the following points:

$$LF = HLF - (CO - HLF) \qquad (14)$$

$$RF = HRF + (HRF - CO) \qquad (15)$$

D. Constructing Fuzzy Rules

From the association rules and the membership functions constructed in the previous two stages, we can construct the following relation:

$$A_1(x_1^i) = A_2(x_2^i) = B(y^i) \qquad (16)$$

From the above relation, a fuzzy rule can be constructed as:

if x_1 is A_1 and x_2 is A_2, then y is B.

However, the fuzzy memberships that are found to be next to each other can be merged to form a trapezoidal fuzzy membership function. Both trapezoidal and triangular functions have the same computational efficiency and are commonly used in most real time applications. The triangular fuzzy membership function can be merged to form a trapezoidal function as follows, and shown in Fig. 5.

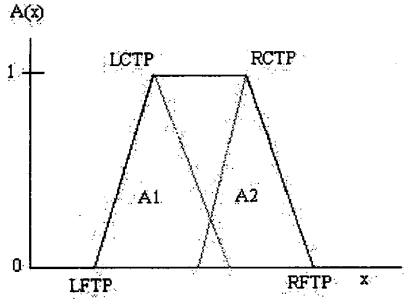

Fig. 5. Merging neighbouring membership functions

For the left flank and the left core of the trapezoidal membership function, it takes the values of the left flank and the left core of the triangular membership function of $A1$:

$$LFTP = LF_{A1} \qquad (16)$$

$$LCTP = CO_{A1} \qquad (17)$$

For the right side of the trapezoidal membership function, it can be calculated similar to the left side by taking the information from the $A2$ instead of $A1$:

$$RFTP = RF_{A2} \qquad (18)$$

$$RCTP = CO_{A2} \qquad (19)$$

V. Illustration Example

An illustrative example is used to demonstrate the applicability of the proposed fuzzy rules extraction technique. The nonlinear two input one output system with the following equation was used. The nonlinear function is shown in Fig. 6.

$$y = (1 + x_1^{-2} + x_2^{-1.5})^2\,,\ 1 \le x_1\,,\ x_2 \le 5 \qquad (20)$$

Fifty data points were used to extract the fuzzy rules to model the above nonlinear system. First the self-organizing neural network was used to cluster the output y. It was started from 4 clusters and then incremented one at a time. The increment was stopped when the minimum support for a cluster was 10% and the maximum support for a cluster was 30%. This meant stopping at 5 clusters in this example.

When looking for association rules, the minimum support was set to 20%, minimum confidence level to 25% and the maximum support to 30%. The *K* value was selected to be 5. The number of partitions was then calculated as 5. Association rules were then extracted from the data.

Fuzzy membership functions were constructed based on the merged partitions, if there were any. With the association rules and the membership functions, a total of 14 fuzzy rules were constructed. The 14 fuzzy rules extracted are shown in Fig. 7. Correlation factor and mean square error were used to measure the accuracy of the fuzzy model. The correlation factor is 0.887 and the mean square error is 0.253. The output plots of the predicted results and the actual output is shown in Fig. 8, and their cross plot is shown in Fig. 9. This model is considered to be a reasonably good model of the original nonlinear system based on equation (20). If some forms of parameter identification can be incorporated, the accuracy will be improved further. This is out of the scope of this paper.

Authorized licensed use limited to: Murdoch University. Downloaded on October 30, 2009 at 03:03 from IEEE Xplore. Restrictions apply.

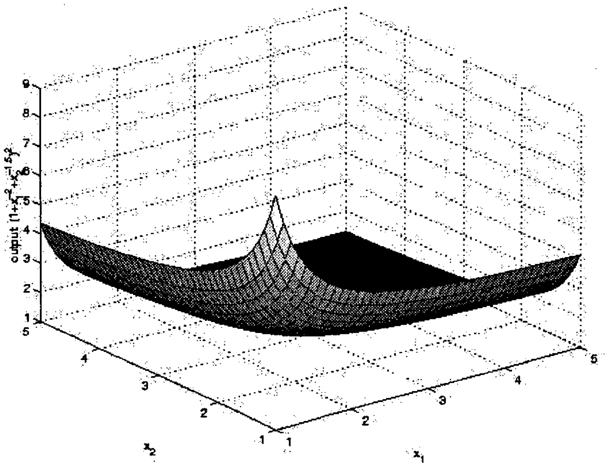

Fig. 6. The nonlinear system used for extracting fuzzy rules

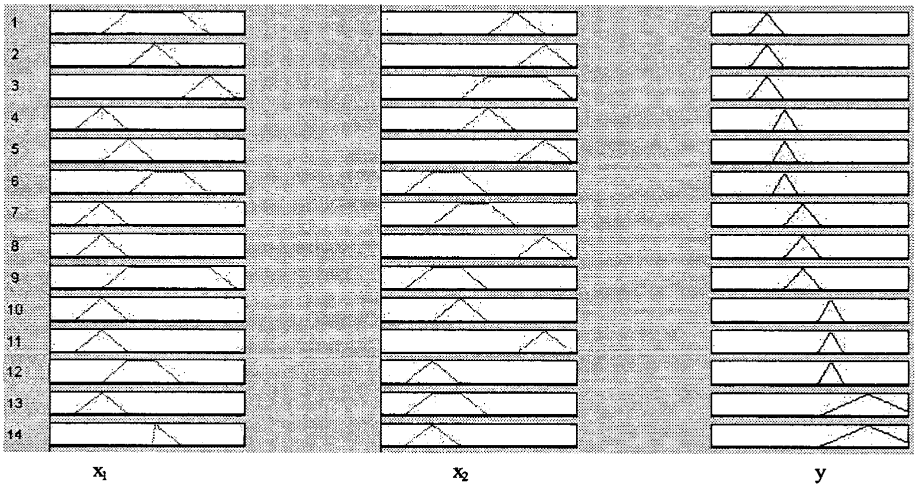

Fig. 7. Fuzzy model for the nonlinear system with 14 fuzzy rules

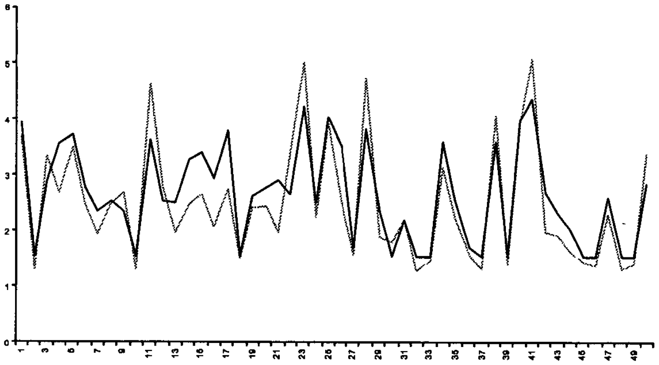

Fig. 8. Plot of the predicted results (black line) and the actual output (grey line).

Authorized licensed use limited to: Murdoch University. Downloaded on October 30, 2009 at 03:03 from IEEE Xplore. Restrictions apply.

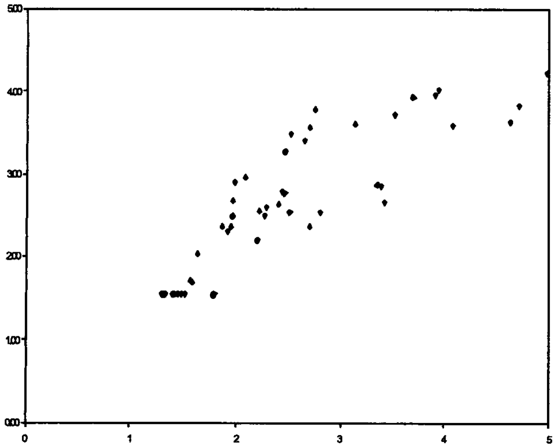

Fig. 9. Cross plot of the results.

I. Conclusion

This paper has shown a technique in extracting fuzzy rules from any available input-output data. It makes use of a self-organising neural network to cluster the output space. Then a data mining algorithm is used to extract association rules that describes the relationships between the input space and the output cluster. Fuzzy membership functions are constructed from the crisp cluster information. With the set of association rules and the fuzzy membership functions, fuzzy rules are then extracted. Our emphasis is put on the rule consequents and then finds an association concerning the premises. This has overcome the major problem of using fuzzy models, which is rule explosion. The size of the fuzzy rules in our technique is not increased exponentially with the increase of input variables and the membership functions. As the time taken by the self-organising neural network and the data-mining algorithm is fast, fuzzy rules can be extracted in a short period of time. Our illustration example has also shown that fuzzy rules extracted by this technique can be used to model nonlinear system with accurate prediction results.

Acknowledgement

This research is supported by the Australian Research Council.

References

[1] Wong, K.W., Gedeon, T.D., Fung, C.C. "Using Modified Alpha-cut Based Fuzzy Interpolation in Petrophysical Properties Prediction," *Proceedings of the 4-th Japan-Australia Joint Workshop on Intelligent and Evolutionary Systems*, October 2000, Japan, pp. 136-142, 2000.

[2] Zadeh, L.A. "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Transactions on System, Man and Cybernetics*, pp. 28-44, 1973.

[3] Mamdani, E.H. and Assilian, S. "An experiment in linguistic synthesis with a fuzzy logic contoller," *International Journal of Man-Machine Studies*, pp. 1-13, 1975.

[4] Wang, L.X. and Mendel, J.M. "Generating fuzzy rules by learning from examples," *IEEE Transactions on System, Man, and Cybernetics*, vol. 22, no. 6, pp. 1414-1427, 1992.

[5] Chiu, S.L. "Fuzzy model identification based on cluster estimation," *Journal of Intelligent Fuzzy System*, vol. 2, pp. 267-278, 1994.

[6] Abe, S. and Lan, M. S. "A method for fuzzy rules extraction directly from numerical data and its application to pattern classification," *IEEE Transactions of Fuzzy System*, vol. 3, pp. 18-28, 1995.

[7] Thawonmas, R. and Abe, S. "Function Approximation Based on Fuzzy Rules Extracted from Partitioned Numerical Data," *IEEE Transaction on Systems, Man, and Cybernetics – Part B: Cybernetics*, vol. 29, no. 3, pp. 525-534, 1999.

[8] Kohonen, T. *Self-Organization and Associative Memory*, 3rd Edition, Springer-Verlag, Berlin, 1989.

[9] Kohonen, T. "The Self-Organising Map," *Proceedings of the IEEE*, Vol. 78, No. 9, pp. 1464-1480, 1990.

[10] Fayyad, U. M., Piatetsky-Shapiro, G., and Smyth, P. "From Data Mining to Knowledge Discovery: An Overview," *Advances in Knowledge Discovery and Data Mining*, (ed.) Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R., The AAAI/MIT Press, Menlo Park, California/Cambridge, Massachusetts, pp. 1-34, 1996.

[11] Agrawal, R., Mannila, H., Srikant, R., Toivonen, H. and Verkamo, A. I. "Fast Discovery of Association Rules," *Advances in Knowledge Discovery and Data Mining*, (ed.) Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R., AAAI/MIT Press, pp. 307-328, 1996.

[12] Savasere, A., Omiecinski, E., and Navathe, S. (1995) "An Efficient Algorithm for Mining Association Rules in Large Databases," *Proceedings of the 21st International Conference on Very Large Data Bases (VLDB'95)*, pp. 432-444, 1995.

[13] Srikant, R., and Agrawal, R. (1996) "Mining Quantitative Association Rules in Large Relational Tables," *Proceedings of ACM SIGMOD Conference on Management of Data*, Montreal, Canada, pp. 1-12.

Authorized licensed use limited to: Murdoch University. Downloaded on October 30, 2009 at 03:03 from IEEE Xplore. Restrictions apply.