Finding Fuzzy Rules for IRIS by Neural Network with Weighted Fuzzy Membership Function

Joon Shik Lim
College of Software, Kyungwon University, Sungnam 461-701

Abstract

Fuzzy neural networks have been successfully applied to analyze/generate predictive rules for medical or diagnostic data. However, most approaches proposed so far have not considered the weights for the membership functions much. This paper presents a neural network with weighted fuzzy membership functions. In our approach, the membership functions can capture the concentrated and essential information that affects the classification of the input patterns. To verify the performance of the proposed model, well-known Iris data set is performed. According to the results, the weighted membership functions enhance the prediction accuracy. The architecture of the proposed neural network with weighted fuzzy membership functions and the details of experimental results for the data set is discussed in this paper.

Key Words: Fuzzy neural network, rule extraction, weighted membership function

1. INTRODUCTION

Neural network and fuzzy set theory can be effectively used for pattern classification and predictive rule generation tool. The goal of pattern classification is to partition the feature space into decision regions. Artificial neural networks have been successfully used in many pattern classification problems [3] [6]. Fuzzy set theory was introduced by Zadeh [23] as a means of representing and processing data by allowing partial set membership rather than crisp set membership or non-membership. As a combined approach of neural network and fuzzy set theory, fuzzy neural networks (FNN) have been proposed as an adaptive decision support tool [2] [8] [11] [12] [13] [14] [17] [20]. Various architectures of FNN have been introduced along with algorithms for learning, adaptation and rule extraction [7] [10] [15] [16]. In pattern classification and predictive rule generation, high accuracy of acceptable regular patterns with simple types of knowledge representation in linguistic terms is desirable. In addition, the way of obtaining the optimal structure for fast adaptation without sacrificing the accuracy and performance is needed. The capability of fuzzy if-then rule extraction is one of the advantages of FNN for medical decision-making. For this, various approaches has been proposed. Among those approaches, fuzzy neural networks with self-organizing systems were developed by [9] [19] [21] to extract knowledge from a given set of training data. Setnes [16] presented a compact and accurate fuzzy rule-based model using a genetic algorithm. However, most approaches proposed so far have not considered the weights for the membership functions much.

In this paper, we present a new neural network model with weighted fuzzy membership functions (NNWFM) to classify IRIS data [4]. In our approach, three membership functions, each of which small, medium, and large functions with weights, are initially defined and trained to generate a combined membership function using the bounded sum function. The weighted fuzzy membership functions can preserve the disjunctive fuzzy information and characteristic input patterns in the pattern space, which can result in reducing the number of hyperboxes and rules. We also present our neural network architecture and experimental results in the following sections. The effectiveness of NNWFM is first validated using the popularly used data sets, Iris [4] for benchmarking of pattern classification.

2. NEURAL NETWORK MODEL with WEIGHTED FUZZY MEMBERSHIP FUNCTION (NNWFM)

2.1 The Structure of NNW

The structure of NNWFM is illustrated in Fig. 1. The NNWFM comprises three layers, namely input, hyperbox, and class layer. The input layer contains n input nodes for n featured input patterns. The hyperbox layer consists of m hyperbox nodes. Each hyperbox node B_i to be connected to a class node contains n fuzzy sets for n input nodes. The output layer is composed of p class nodes. Each class node is connected to one or more hyperbox nodes. An nth pattern can be recorded as I_n = \{A_1; \cdots; A_n; \cdots; A_m\}, where diagnosis is the result of diagnosis and A_n is the pattern on n different features. 'Unknown' features are represented as NULL. The nth fuzzy set of B_n, denoted by B'_n, has three weighted membership functions as shown in Fig. 2. These three weighted fuzzy membership functions are adjusted by the learning algorithm, which will be discussed in Section 2.2. The hyperbox nodes in the hyperbox layer are controlled by the adjust operations. After learning, the rules for classification can be extracted by
the hyperbox layer. The detail of rule extraction processes is described in the Section 2.3.

![Diagram of NNWFM Structure](image)

**Fig. 1. Structure of NNWFM**

### 2.2 Definitions and Operations

1) \( w_{ij} \): The connection weight between a hyperbox node \( B_i \) and a class node \( C_j \) is represented by \( w_{ij} \), which is initially set to 0. If there is a connection from a hyperbox node \( B_i \) to a class node \( C_j \), the \( w_{ij} \) will be set to 1 from 0. \( C_j \) can have more than one connection from hyperbox nodes, whereas \( B_i \) is restricted to have one connection to a class node.

2) \( v_0, v_1, v_2, \) and \( v_3 \) represent the center vertices of the small, medium, and large membership functions respectively in Fig. 2. The center vertices can be adjusted during learning, while the other vertices \( v_0 \) and \( v_4 \) are fixed. It is assumed that the input feature value \( a_i \) ranges from \( v_{\min} \) to \( v_{\max} \) as shown in Fig. 2.

3) \( \mu_j \) and \( W_j \): \( \mu_j \) and \( W_j \) are the set of membership functions of a hyperbox node \( B_i \), for \( j = 1, 2, 3 \), representing small, medium, and large respectively. The shape of each membership function \( \mu_j \) is triangular, which is characterized by three vertices \((v_{j-1}, v_j, v_{j+1})\) with its membership function weight \( W_j \) (0 \( \leq W_j \leq 1 \), random weights in the range of 0.45 \( \leq W_j \leq 0.55 \) are initially set) that represents the strength of the membership function determined through learning. The shaded triangles in Fig. 2, called weighted fuzzy membership functions, can be formed by \((v_{j-1}, v_j, v_{j+1})\).

4) \( \text{Len}(\mu_j) \): To measure the size of a \( \mu_j \) length function, \( \text{Len}(\mu_j) \) is defined as follow:

\[
\text{Len}(\mu_j) = (v_{j+1} - v_{j-1}) / 2 W_j
\]

This function is similar to the length function, \( \text{Len}(\mu) \), defined by Lee et al. [13] except that the \( \text{Len}(\mu_j) \) is multiplied by the weights \( W_j \).

5) \( \text{Adjust}(B_i) \): This operation adjusts membership functions and their weights for \( n \) fuzzy sets in the hyperbox node \( B_i \) according to the input \( A_k = (a_1, a_2, \ldots, a_n) \). For each \( a_i \) set of membership functions with the input \( a_i \), the \( v_{\min} \) and \( W_j \) of the membership functions are adjusted by the value of \( \mu(a_i) \),

\[
\text{new}(v_i) = v_i + \Delta \text{Len}(\mu(a_i)) W_i
\]

\[
\text{new}(W_i) = W_i + \Delta \text{Len}(\mu(a_i)) - W_i
\]

where \( j = 1, 2, 3 \). As a result of \( \text{Adjust}(B_i) \) operation, the new vertices \( \text{new}(v_i) \)s and new weights \( \text{new}(W_i) \)s are set by the following expression:

![Diagram of Adjusted Membership Functions](image)

**Fig. 2. An \( i \)th Set of Weighted Membership Functions of \( B_i \)**

In these expressions, the \( \Delta \) and \( \text{Len}(\mu(a_i)) \) are the learning rate in the range from 0 to 1, and the variable \( \Delta \) is the difference between \( v_i \) and input \( a_i \). If \( \Delta \) is bigger than the adjacent \( \text{Len}(\mu(a_i)) \), the smaller one is selected. The detail process of the \( \text{Adjust}(B_i) \) operation is described in the learning algorithm in the next section. Fig. 3 shows the result of the \( \text{Adjust}(B_i) \) operations for the input \( a_i \) and the \( i \)th set of fuzzy weighted membership functions in \( B_i \).

![Diagram of Adjusted Membership Functions](image)

**Fig. 3. Example of Before And After Adjust(Bi) Operation**

As shown in Fig. 3, the weights and the center of each membership functions are adjusted by the \( \text{Adjust}(B_i) \) operation, e.g., \( W_1, W_2 \), and \( W_3 \) are moved down, \( v_1 \) and \( v_2 \) are moved
Finding Fuzzy Rules for IRIS by Neural Network with Weighted Fuzzy Membership Function

6) Random(B_i): This operation makes the hyperbox node B_i with I (number of inputs) sets of randomly distributed membership functions with their random weights ranging from 0.45 to 0.55. The random center vertices v_i should be in their ranges such that

\[ r_{i-1} \leq v_i \leq r_i, \quad \text{where } i = 1, 2, 3 \]

while \( v_0 \) and \( c_0 \) are fixed as in Fig. 2. Also, the connection weights \( w_j \) are set to 0.

7) Output(B_j): For the \( h \)th input \( A_h \) \((a_1, a_2, \ldots, a_d)\) with \( n \) features to the hyperbox \( B_j \) output of the \( B_i \) is calculated by

\[ \text{Output}(B_i) = \frac{1}{n} \sum_{j=1}^{n} B_j(\mu_i(a_j))W_j. \quad (2.4) \]

2.3 Learning Algorithm for NNWFM

This section describes the learning algorithm for NNWFM. The algorithm uses the Learning(B,C) procedure to adjust the locations of vertices and weights, and to connect the hyperbox nodes to class nodes.

Algorithm NNWFM

1. While (result is satisfied)
   1.1 for \( i = 1 \to m \) \( \quad // \) \( m \) is number of hyperboxes,
      // usually start from number input
      1.1.1 Random(B_i);
      1.1.2 for \( j = 1 \to p \) \( \quad // \) \( p \) is number of class nodes
      1.1.2.1 \( w_j = 0 \); \( \quad // \) initial connection weight
      // between \( B_i \) and \( C_j \)
      1.2 for \( k = 1 \to h \) \( \quad // \) \( h \) is number of input patterns
      1.2.1 find \( B_i \) that has the maximum value of \( \text{EnhOutput}(B_i) \) \( // \) see the formula 2.6 among \( m \) hyperbox nodes from the input \( A_h \)
      // input vector: \( A_h = (a_1, a_2, \ldots, a_d), \) \( \) diagnosis \( \)
      1.2.2 Learning(B_i,C); \( // \) \( C_j \) is a diagnosis in \( B_i \)

Procedure Learning(B,C);

// \( m \) is number of hyperboxes
1. Case 1: \( \forall m, \text{wmi} = 0 \), where \( m \neq i \)
   1.1 \( w_{si} = 1 \);
   1.2 Adjust(B);

2. Case 2: \( \exists m \) satisfying \( w_{si} = 1 \), where \( m \neq i \)
   2.1 \( w_{si} = 1 \);
   2.2 Adjust(B);

2.4 Fuzzy Rule Extraction

The learned NNWFM can be used for fuzzy rule extraction in if-then form to classify input patterns. After learning, each fuzzy set in hyperbox node \( B_i \) contains three weighted fuzzy membership functions (WFM, grey membership functions in Fig. 6).

Fig. 6. Example of Bounded Sum of the Three Weighted Fuzzy Membership Functions (BSWF, Bold Line)

The rules can be extracted directly from the WFM. We suggest a rule extraction strategy as described below.

1) The bounded sum (one of operations on fuzzy set) of WFM (BSWF) in the ith fuzzy set of \( B_i(x) \), denoted as \( \mu_i^e(x) \) (bold line in Fig. 6), is defined by

\[ \mu_i^e(x) = \sum_{j=1}^{3} \mu_j B_j(\mu_j(x)). \quad (2.5) \]

The bounded sum of WFM combines the fuzzy characteristics of three WFM in Fig. 6.

2) We use the Takagi and Sugeno[18] model for an inference mechanism that performs the reasoning process to derive an Output(B_i) (2.4). In addition, we suggest a heuristic of \( \text{Len}(\mu) \) to enhance the reasoning capability such that:

\[ \text{EnhOutput}(B_i) = \frac{1}{n} \sum_{j=1}^{n} \mu_j B_j(\mu_j) / \text{Len}(\mu). \quad (2.6) \]
The bold lines in Fig. 6 are enhanced BSWFM using (2.6). The heuristic of \( \text{Len}(\mu_i) \) is based on the concept that smaller \( \text{Len}(\mu_i) \) has more concentrated information that is essential for classification of the given pattern.

3. EXPERIMENTAL RESULTS

In this section, we present the experimental results for the Iris data set to evaluate the accuracy and rule extraction capability of the proposed NNWFMs.

The data set used to evaluate the performance of the proposed model is Fisher's Iris data [4], which is one of the popularly used data sets in pattern classification studies. The data set consists of 150 four-dimensional feature vectors (sepal length, sepal width, petal length, and petal width) in three classes (Setosa, Versicolor, and Virginica; 50 instances for each class). Nauck [15] uses a neuro-fuzzy method to learn fuzzy classifiers presenting seven rules with three misclassifications. Ishibuchi [7] obtains four fuzzy rules with four error classifications by weighted voting method. Setnes [16] proposes a genetic algorithm-based model, called GA-fuzzy modeling that produces two or three rules with only one error classification.

Fig. 7. BSWFMs and Enhanced BSWFMs (bold lines) for Iris Data Set

The three prediction rules in Table 1 for Iris data are extracted from the enhanced BSWFMs in Fig. 7. In Fig. 7, each graph represents trained membership functions for three classes: Setosa, Versicolor, Virginica, for each feature in the Iris data set. In this experiment, we used the membership functions shown by the bold lines, which are Enhanced BSWFMs based on BSWFMs. In each graph, the enhanced BSWFMs (the bold lines) are obtained by the division of \( \text{Len}(\mu_i) \) function defined in the section 2.4 (see the equation 2.6). The \( \text{Len}(\mu_i) \) function measures the width of a membership function. Heuristically, we can consider that the narrower the width of the membership function, the more concentrated information it may contain. The comparisons of their performance results on Iris data are shown in Table 2. The complete 150 cases in the data set are used as a training set. The results show that the proposed model, NNWFMs, has the best classification rate with three rules.
TABLE 1
Rules for Iris Data

<table>
<thead>
<tr>
<th>Rule</th>
<th>Sepal length</th>
<th>Sepal width</th>
<th>Petal length</th>
<th>Petal width</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(a)</td>
<td>(a)</td>
<td>(a)</td>
<td>(a)</td>
<td>Setosa</td>
</tr>
<tr>
<td>2</td>
<td>(b)</td>
<td>(b)</td>
<td>(b)</td>
<td>(b)</td>
<td>Versicolor</td>
</tr>
<tr>
<td>3</td>
<td>(c)</td>
<td>(c)</td>
<td>(c)</td>
<td>(c)</td>
<td>Virginica</td>
</tr>
</tbody>
</table>

In each graph in Fig. 7, line (a), (b), and (c) (Enhanced BSWFM, bold line) represents each rule for classification.

TABLE 2
Performance Results on Iris Data

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>No. of Misclassifications</th>
<th>No. of Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iribuchi [7]</td>
<td>4/150</td>
<td>4</td>
</tr>
<tr>
<td>Setnes [16]</td>
<td>1/150</td>
<td>2 - 3</td>
</tr>
<tr>
<td>NNWFN</td>
<td>0/150</td>
<td>3</td>
</tr>
</tbody>
</table>

4. CONCLUDING REMARKS

In the proposed neural network model, a fuzzy set in a hyperbox node implies information of small, medium, and large weighted fuzzy membership functions. After learning, the enhanced bounded sums of weighted fuzzy membership functions (enhanced BSWFM) in hyperbox nodes contain characteristics of input patterns. This feature gives the following advantages.

1) The enhanced BSWFM in a hyperbox can maintain complementary information which handles local maxima problems, ambiguous subsets of feature space, and disjunctive fuzzy information in the pattern space.
2) Rules can be extracted easily since enhanced BSWFM are direct interpretation of fuzzy rules.
3) The number of hyperbox nodes can be reduced, since the enhanced BSWFM represent the characteristics of input patterns by preserving the fuzziness.
4) The enhanced BSWFM graphs show visually important feature values.

By the above advantageous properties of NNWFN, we could extract human comprehensive three rules for IRIS data set to classify with 100% accuracy.

REFERENCES


Joon Shik Lim received the B.S. and M.S. degrees in computer science from Inha University, Korea, University of Alabama at Birmingham, and Ph.D. degree from Louisiana State University, Baton Rouge, Louisiana, in 1986, 1989, and 1994, respectively.

He is currently a Professor of College of Software at Kyungwon University, Korea. His research focuses on neuro-fuzzy systems, biomedical prediction systems, and human-centered systems. He has authored three textbooks Computer Practice and Applications (Jungik Press, 1998), Artificial Intelligence Programming (Green Press, 2000), and Javaquest (Green Press, 2003).

Phone : 031)750-5330
Fax : 031)750-5662
E-mail : jslim@kyungwon.ac.kr