International Journal of Applied Science and Engineering
2004. 2, 1: 90-104

A New Method for Handling Fuzzy Classification Problems Using Clustering Techniques

Shyi-Ming Chen[a*] and Cheng-Hao Yu[b]

[a] *Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan, R. O. C.*

[b] *Department of Electronic Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan, R. O. C.*

Abstract: It is obvious that fuzzy classification systems are important applications of the fuzzy set theory. Fuzzy classification systems can deal with perceptual uncertainties in classification problems. In recent years, many methods have been proposed to deal with fuzzy classification problems. In this paper, we present a new method to deal with the Iris data classification problem based on the concept of fuzzy compatibility relations for finding the cluster centers of training instances. The proposed method can get a higher average classification accuracy rate to deal with the Iris data classification problem than the existing methods.

Keywords: clustering techniques; fuzzy classification systems; fuzzy relations; fuzzy sets; Iris data.

1. Introduction

In recent years, many methods have been proposed to design fuzzy classification systems based on the fuzzy set theory [19] for dealing with fuzzy classification problems [1, 2, 4, 7-11, 13, 16-18]. In [1], Chang et al. presented a method for generating fuzzy rules from numerical data for handling fuzzy classification problems. In [2], Castro et al. presented a method to learn maximal structure rules in fuzzy logic for handling fuzzy classification problems. In [4], Chen et al. presented a method to generate fuzzy rules from training data by using genetic algorithms. In [7], Hong et al. presented a method for generating fuzzy rules from training instances based on finding relevant attributes and membership functions. In [8], Hong et al. presented a method for inducing fuzzy rules based on processing individual fuzzy attributes. In [9], Hong et al. presented a method for the induction of fuzzy rules and membership functions from training examples. In [10], Hong et al. investigated the effect of the merging order on performance of fuzzy induction. In [11], Kao et al. presented a method for constructing membership functions and generating fuzzy rules from training data containing noise. In [13], Lin et al. presented a method to generate weighted fuzzy rules from training instances using genetic algorithms. In [16], Wang et al. presented a fuzzy inductive strategy for modular rules. In [17],

* Corresponding author: e-mail: smchen@et.ntust.edu.tw *Accepted for Publication: Dec. 23, 2003*

Wu et al. presented a method for constructing membership functions and fuzzy rules from training examples.

In this paper, we present a new method to deal with the Iris data [5] classification problem based on the concept of fuzzy compatibility relations for finding the cluster centers of the training instances. The proposed method can discard useless input attributes, thus improving the average classification accuracy rate. It can get a higher average classification accuracy rate to deal with the Iris data classification problem than the existing methods.

The rest of this paper is organized as follows. In Section 2, we briefly review fuzzy sets and fuzzy relations from [12] and [19]. In Section 3, we present a new method for dealing with the Iris data classification problem based on clustering techniques. In Section 4, we use an example to illustrate the proposed method. In Section 5, we show the experimental results of the proposed method. The conclusions are discussed in Section 6.

2. Fuzzy sets and fuzzy relations

Roughly speaking, a fuzzy set [19] is a set with fuzzy boundaries. A fuzzy set A of the universe of discourse U, $U = \{u_1, u_2, \dots, u_n\}$, can be characterized by a membership function μ_A, $\mu_A: U \rightarrow [0, 1]$, where $\mu_A(u_i)$ indicates the grade of membership of the element u_i in the fuzzy set A, $\mu_A(u_i) \in [0, 1]$, and $1 \le i \le n$. A fuzzy set A of the universe of discourse U can be represented by

$$A = \mu_A(u_1)/u_1 + \mu_A(u_2)/u_2 + \dots + \mu_A(u_n)/u_n, \quad (1)$$

where the symbol "+" denotes the union operator and the symbol "/" denotes the separator.

Definition 2.1: Let A_1, A_2, … , and A_n be fuzzy sets and let $A_1 \times A_2 \times \dots \times A_n$ be their Cartesian product. Then, a *n*-ary fuzzy relation R is defined as follows [12]:

$$R(A_1, A_2, \dots, A_n) \subset A_1 \times A_2 \times \dots \times A_n, \quad (2)$$

where

$$A_1 \times A_2 \times \dots \times A_n = \{(x_1, x_2, \dots, x_n) \mid x_i \in A_i \text{ and } 1 \le i \le n\}. \quad (3)$$

The membership function of $R(A_1, A_2, \dots, A_n)$ is represented by $\mu_R(x_1, x_2, \dots, x_n)$, where $\mu_R(x_1, x_2, \dots, x_n) \in [0, 1]$ and $1 \le i \le n$.

Definition 2.2: Assume that A is a fuzzy set. A fuzzy relation R(A, A) is reflexive [12] if and only if

$$\mu_R(x_i, x_i) = 1, \quad \forall x_i \in A. \quad (4)$$

Definition 2.3: Assume that A is a fuzzy set. A fuzzy relation R(A, A) is symmetric [12] if and only if

$$\mu_R(x, y) = \mu_R(y, x), \quad \forall x, y \in A. \quad (5)$$

If a binary fuzzy relation R is reflexive, symmetric, and transitive, then it is called a fuzzy equivalence relation [12]. If a binary fuzzy relation R is reflexive and symmetric, then it is called a compatibility relation [12].

3. A new method for dealing with the Iris data classification problem based on clustering techniques

In this section, we present a new method to deal with the Iris data [5] classification problem based on the concept of fuzzy compatibility relations to find cluster centers from training instances, where we can choose n instances from the Iris data as the training data set, and let the other instances of the Iris data be the testing data set. There are three species of flowers in the Iris data (i.e., "Iris-Setosa", "Iris-Versicolor" and "Iris-Virginica") and there are 150 instances in the Iris data, with 50 instances for each species and each species has four input attributes (i.e., Sepal Length (SL), Sepal Width (SW), Petal Length (PL) and Petal Width (PW)). Assume that there are *n* training instances x_1, x_2, …, and x_n, and assume that the *i*th training instance x_i has *m* input attribute values $x_{i1}, x_{i2}, \ldots, x_{im}$, and one output classification y_i shown as follows:

$x_i = ((x_{i1}, x_{i2}, \ldots, x_{im}), y_i)$,

where $x_{i,j}$ is the value of the jth input attribute X_j of the ith training instance x_i and $x_{i,j}$ is a real value; y_i is the output classification of the output attribute Y of the ith training instance x_i, where $y_i \in$ {Iris-Setosa, Iris-Versicolor, Iris-Virginica}, $i = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, m$.

First, we find out the individual domain of each input attribute for each type of output classification. For each input attribute X_j, where $1 \leq j \leq 4$, we calculate the number of training instances for which the values of the input attribute X_j falls in the overlapping interval formed by the overlapping of the domain values of the input attribute X_j for each species of flowers. For example, assume that the training instances have three kinds of output classifications (i.e., three species of flowers), i.e., o_1, o_2 and o_3:

(1) When the classification is o_1: Assume that the minimal attribute value and the maximal attribute value of the input attribute X_j of the training instances are 2 cm and 3 cm, respectively.

(2) When the classification is o_2: Assume that the minimal attribute value and the maximal attribute value of the input attribute X_j of the training instances are 4 cm and 6 cm, respectively.

(3) When the classification is o_3: Assume that the minimal attribute value and the maximal attribute value of the input attribute X_j of the training instances are 5 cm and 7 cm, respectively.

Then, the domain of the input attribute X_j for the classification o_1 is [2 cm, 3 cm]; the domain of the input attribute X_j for the classification o_2 is [4 cm, 6 cm]; the domain of the input attribute X_j for the classification o_3 is [5 cm, 7 cm]. The distribution of the domain values of different output classifications of the input attribute X_j is shown in Figure 1.

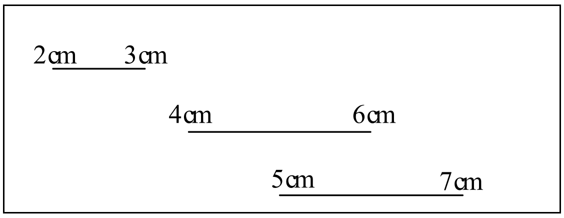

Figure 1. The distribution of the attribute values of different output classifications of the input attribute X_j

Then, we can see that

(i) Because the domain values of the input attribute X_j for the output classification o_1 and the domain values of the input attribute X_j for the output classification o_2 are not overlapping, the overlapping interval is an empty set. The number of training instances in which the values of the input attribute X_j fall in the overlapping interval is 0.

(ii) Because the domain values of the input attribute X_j for the output classification o_1 and the domain values of the input attribute X_j for the output classification o_3 are not overlapping, the overlapping interval is an empty set. The number of training instances in which the values of the input attribute X_j fall in the overlapping interval is 0.

(iii) Because the domain values of the input attribute X_j for the output classification o_2 and the domain values of the input attribute X_j for the output classification o_3 are overlapping, the overlapping interval is [5 cm, 6 cm]. Then, we can calculate the number of training instances for which the values of the input attribute X_j fall in the overlapping interval [5 cm, 6 cm] formed by the overlapping of the intervals [4 cm, 6 cm] and [5 cm, 7 cm].

Let n_1 denote the number of training instances whose values of the input attribute X_j fall in the overlapping interval formed by overlapping the domain values of the input attribute X_j of the species of flowers "Iris-Setosa" and "Iris-Versicolor", let n_2 denote the number of training instances whose values of the input attribute X_j fall in the overlapping interval formed by overlapping the domain values of the input attribute X_j of the species of flowers "Iris-Setosa" and "Iris-Virginica", let n_3 denote the number of training instances whose values of the input attribute X_j fall in the overlapping interval formed by overlapping the domain values of the input attribute X_j of the species of flowers "Iris-Versicolor" and "Iris-Virginica", let z_1 denote the number of training instances belonging to the species of flowers "Iris-Setosa" and "Iris-Versicolor", let z_2 denote the number of training instances belonging to the species of flowers "Iris-Setosa" and "Iris-Virginica", let z_3 denote the number of training instances belonging to the species of flowers "Iris-Versicolor" and "Iris-Virginica", and let λ be an overlapping threshold value given by the user, where $\lambda \in [0, 1]$. If $\frac{n_1 + n_2 + n_3}{z_1 + z_2 + z_3} > \lambda$, then the input attribute X_j can be discarded.

In the following, we assume that there is a data set X which contains n training instances, and each training instance has m input attribute values. Then, a fuzzy compatibility relation R on X is defined by

$$R(x_i,\ x_k) = 2^{D(x_i, x_k)} - 1, \tag{6}$$

where $x_i \in X$, $x_k \in X$, $R(x_i, x_k) \in [0, 1]$, and $D(x_i, x_k)$ is the distance between the training instances x_i and x_k defined as follows:

$$D(x_i, x_k) = \sum_{j=1}^{m} \mu_j \left[(1 - \frac{|x_{ij} - x_{kj}|}{|x_{\max j} - x_{\min j}|}) / \delta \right], \tag{7}$$

where $x_{i,j}$ denotes the value of the jth input attribute X_j of the ith training instance, $x_{k,j}$ denotes the value of the jth input attribute X_j of the kth training instance, $x_{max,j}$ denotes the maximum attribute value of the jth input attribute X_j of the training instances, $x_{min,j}$ denotes the minimum attribute value of the jth input attribute X_j of the training instances and m denotes the number of input attributes. For the Iris data [5], if an input attribute X_j was discarded, then we let $\mu_j = 0$; otherwise, we let $\mu_j = 1$, where $1 \le j \le 4$. The value of δ denotes the number of input attributes which are not discarded. We can use formulas (6) and (7) to derive the fuzzy compatibility relation R of the training instances. Let $R(x_i, x_j)$ denote the fuzzy compatibility value between the training instances x_i and x_j, where R is a fuzzy compatibility relation and $R(x_i, x_j) \in [0, 1]$. Let $TFCV(x_i)$ denote the total fuzzy compatibility value of the training instance x_i, where $TFCV(x_i) = \sum_{j=1}^{n} R(x_i, x_j)^*$, where if $R(x_i, x_j) \ge \alpha$, then let $R(x_i, x_j)^* = R(x_i, x_j)$; if $R(x_i, x_j) \le \beta$, then let $R(x_i, x_j)^* = 1 - R(x_i, x_j)$; if $\beta < R(x_i, x_j) < \alpha$, then let $R(x_i, x_j)^* = 0$, where α and β are the upper bound and the lower bound threshold values given by the user, and $0 \le \beta \le \alpha \le 1$. The proposed method for handling the Iris data classification problem is presented as follows:

Step 1: Find the maximum attribute value and the minimum attribute value for each input attribute of each species of flowers from the training data set.

Step 2: For each input attribute, calculate the number of training instances that falls in each interval formed by the overlapping of each pair of species of flowers.

Step 3: Let n_1 denote the number of training instances whose values of the input attribute X_j fall in the overlapping interval formed by overlapping the domain values of the input attribute X_j of the species of flowers "Iris-Setosa" and "Iris-Versicolor", let n_2 denote the number of training instances whose values of the input attribute X_j fall in the overlapping interval formed by overlapping the domain values of the input attribute X_j of the species of flowers "Iris-Setosa" and "Iris-Virginica", let n_3 denote the number of training instances whose values of the input attribute X_j fall in the overlapping interval formed by overlapping the domain values of the input attribute X_j of the species of flowers "Iris-Versicolor" and "Iris-Virginica", let z_1 denote the number of training instances belonging to the species of flowers "Iris-Setosa" and "Iris-Versicolor", let z_2 denote the number of training instances belonging to the species of flowers "Iris-Setosa" and "Iris-Virginica", let z_3 denote the number of training instances belonging to the species of flowers "Iris-Versicolor" and "Iris-Virginica", and let λ be an overlapping threshold value given by the user, where $\lambda \in [0, 1]$.

For each input attribute X_j, where $1 \le j \le 4$ **do**
  if $\frac{n_1 + n_2 + n_3}{z_1 + z_2 + z_3} > \lambda$, where λ is an overlapping threshold value given by the user
  then discard the input attribute X_j
end.

Step 4: Based on formula (7), calculate the distance $D(x_i, x_k)$ between each pair of training instances x_i and x_k in the training data set.

Step 5: Based on formula (6), derive the fuzzy compatibility relation R of the training instances.

Step 6: **For** i = 1 **to** n **do**
    $TFCV(x_i) = 0$;
  end;
  for i = 1 **to** n **do**
    for j = 1 **to** n **do**
      if $R(x_i, x_j) \ge \alpha$, where α is the upper bound threshold value given by the user and $\alpha \in [0, 1]$
      then let $TFCV(x_i) = TFCV(x_i) + R(x_i, x_j)$;
      if $R(x_i, x_j) \le \beta$, where β is the lower bound threshold value given by the user and $\beta \in [0, 1]$
      then let $TFCV(x_i) = TFCV(x_i) + (1 - R(x_i, x_j))$;
      if $\beta < R(x_i, x_j) < \alpha$ **then** let $TFCV(x_i) = TFCV(x_i) + 0$
    end
  end.

Step 7: **For** each species of flower **do**
  while there are training instances for the species of flower that have not been marked **do**
  begin
    let the training instance which has a maximum total fuzzy compatibility value be a cluster center for the species of flower;
    if the fuzzy compatibility value between this generated cluster center and any training instance x_k in the training data set is larger than the level threshold value γ given by the user, where $\gamma \in [0, 1]$
    then mark this training instance x_k from the training data set, where $1 \le k \le n$. In this situation, if the number of marked training instances in the training data set is less than 5 percent of the number of training instances, then this cluster center can not be used, and we free the marked training instances (i.e., unmark the marked training instances) except this cluster center
  end
end.

Then, based on the generated cluster centers of the training instances, we can classify the testing instances described as follows. Assume that x_a, x_b, … , and x_z are the cluster centers of the training instances, then

For each testing instance x_k in the testing data set **do**

if the testing instance x_k has the largest fuzzy compatibility value $R(x_k, x_p)$ regarding the cluster center x_p, where $p \in \{a, b, \dots, z\}$

then the testing instance x_k is classified into the same species of flower of the training instance x_p, where $p \in \{a, b, \dots, z\}$

end.

4. An example

In this section, we apply the proposed algorithm to deal with the Iris data [5] classification problem. For simple illustration, we randomly choose 15 instances from the Iris data as the training data set and the testing data set. Assume that the chosen instances are as shown in Table 1 and assume that the upper bound threshold value α, the lower bound threshold value β, the level threshold value γ and the overlapping threshold value λ given by the user are 0.5, 0.2, 0.7 and 0.2, respectively, i.e., $\alpha = 0.5$, $\beta = 0.2$, $\gamma = 0.7$ and $\lambda = 0.2$.

In the following, we use x_1, x_2, …, and x_{15} to represent the training instances shown in Table 1, where x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm), 1), x_2 = ((4.9 cm, 3.0 cm, 1.4 cm, 0.2 cm), 1), x_3 = ((4.7 cm, 3.2 cm, 1.3 cm, 0.2 cm), 1), x_4 = ((4.6 cm, 3.1 cm, 1.5 cm, 0.2 cm), 1), x_5 = ((5.0 cm, 3.6 cm, 1.4 cm, 0.2 cm), 1), x_6 = ((7.0 cm, 3.2 cm, 4.7 cm, 1.4 cm), 2), x_7 = ((6.4 cm, 3.2 cm, 4.5 cm, 1.5 cm), 2), x_8 = ((6.9 cm, 3.1 cm, 4.9 cm, 1.5 cm), 2), x_9 = ((5.5 cm, 2.3 cm, 4.0 cm, 1.3 cm), 2) x_{10} = ((6.5 cm, 2.8 cm, 4.6 cm, 1.5 cm), 2), x_{11} = ((6.3 cm, 3.3 cm, 6.0 cm, 2.5 cm), 3), x_{12} = ((5.8 cm, 2.7 cm, 5.1 cm, 2.5 cm), 3), x_{13} = ((7.1 cm, 3.0 cm, 5.9 cm, 2.1 cm), 3), x_{14} = ((6.3 cm, 2.9 cm, 5.6 cm, 1.8 cm), 3), x_{15} = ((6.5 cm, 3.0 cm, 5.8 cm, 2.2 cm), 3). The output attribute value "1" denotes that the species of flower is "Iris-Setosa", the output attribute value "2" denotes that the species of flower is "Iris-Versicolor", and the output attribute value "3" denotes that the species of flower is "Iris-Virginica".

[Step 1] Based on Table 1, we can find the maximum attribute value and the minimum attribute value for each input attribute of each species of flowers as shown in Table 2.

Table 1. A small training data set

Iris-Setosa					Iris-Versicolor					Iris-Virginica				
SL	SW	PL	PW	Output	SL	SW	PL	PW	Output	SL	SW	PL	PW	Output
5.1 cm	3.5 cm	1.4 cm	0.2 cm	1	7.0 cm	3.2 cm	4.7 cm	1.4 cm	2	6.3 cm	3.3 cm	6.0 cm	2.5 cm	3
4.9 cm	3.0 cm	1.4 cm	0.2 cm	1	6.4 cm	3.2 cm	4.5 cm	1.5 cm	2	5.8 cm	2.7 cm	5.1 cm	1.9 cm	3
4.7 cm	3.2 cm	1.3 cm	0.2 cm	1	6.9 cm	3.1 cm	4.9 cm	1.5 cm	2	7.1 cm	3.0 cm	5.9 cm	2.1 cm	3
4.6 cm	3.1 cm	1.5 cm	0.2 cm	1	5.5 cm	2.3 cm	4.0 cm	1.3 cm	2	6.3 cm	2.9 cm	5.6 cm	1.8 cm	3
5.0 cm	3.6 cm	1.4 cm	0.2 cm	1	6.5 cm	2.8 cm	4.6 cm	1.5 cm	2	6.5 cm	3.0 cm	5.8 cm	2.2 cm	3

[Step 2] For the input attribute "Sepal Length" shown in Table 2, we can see that

(1) When the species of flower is "Iris Setosa": The minimum attribute value of the input attribute "Sepal Length" is 4.6 cm and the maximum attribute value of the input attribute "Sepal Length" is 5.1 cm.

(2) When the species of flower is "Iris-Versicolor": The minimum attribute value of the input attribute "Sepal Length" is 5.5 cm and the maximum attribute value of the input attribute "Sepal Length" is 7.0 cm.

(3) When the species of flower is "Iris-Virginica": The minimum attribute value of the input attribute "Sepal Length" is 5.8 cm and the maximum attribute value of the input attribute "Sepal Length" is 7.1 cm.

(4) Because the domain values of the input attribute "Sepal Length" for the species of flower "Iris-Setosa" and the domain values of the input attribute "Sepal Length" for the species of flower "Iris-Versicolor" are not overlapping, the overlapping interval is an empty set. The number of training instances whose values of the input attribute "Sepal Length" fall in the overlapping interval is 0.

(5) Because the domain values of the input attribute "Sepal Length" for the species of flower "Iris-Setosa" and the domain values of the input attribute "Sepal Length" for the species of flower "Iris-Virginica" is not overlapping, the overlapping interval is an empty set. The number of training instances whose values of the input attribute "Sepal Length" fall in the overlapping interval is 0.

Table 2. Minimum attribute values and maximum attribute values for different species of flowers of the training data set

Species	Input attribute	Minimum attribute value	Maximum attribute value
Iris-Setosa	Sepal Length (SL)	4.6 cm	5.1 cm
	Sepal Width (SW)	3.0 cm	3.6 cm
	Petal Length (PL)	1.3 cm	1.5 cm
	Petal Width (PW)	0.2 cm	0.2 cm
Iris-Versicolor	Sepal Length (SL)	5.5 cm	7.0 cm
	Sepal Width (SW)	2.3 cm	3.1 cm
	Petal Length (PL)	4.0 cm	4.9 cm
	Petal Width (PW)	1.3 cm	1.5 cm
Iris-Virginica	Sepal Length (SL)	5.8 cm	7.1 cm
	Sepal Width (SW)	2.7 cm	3.3 cm
	Petal Length (PL)	5.1 cm	6.0 cm
	Petal Width (PW)	1.8 cm	2.5 cm

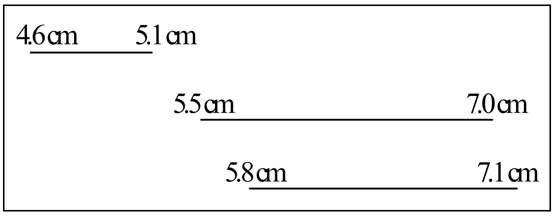

Figure 2. The distribution of the domain values of the different species of flowers for the attribute "Sepal Length"

(6) Because the domain values of the input attribute "Sepal Length" for the species of flower "Iris-Versicolor" and the domain values of the input attribute "Sepal Length" for the species of flower "Iris-Virginica" are overlapping, the overlapping interval is [5.8 cm, 7.0 cm]. Then, we can calculate the number of training instances whose values of the input attribute "Sepal Length" fall in the overlapping interval [5.8 cm, 7.0 cm]. From Table 1, we can see that the number of training instances whose values of the input attribute "Sepal Length" fall in the overlapping interval [5.8 cm, 7.0 cm] is 8.

In the same way, we can calculate the number of training instances whose values of each input attribute fall in the overlapping interval formed by each pair of species of flowers as shown in Table 3.

[Step 3] From Table 3, we can see that the number of training instances whose values of the input attribute "Sepal Length" fall in the overlapping interval formed by the values of the input attribute "Sepal Length" for the species of flower "Iris-Setosa" and the values of the input attribute "Sepal Length" for the species of flower "Iris-Versicolor" is 0; the number of training instances whose values of the input attribute "Sepal Length" fall in the overlapping interval formed by the values of the input attribute "Sepal Length" for the species of flower "Iris-Setosa" and the values of the input attribute "Sepal Length" for the species of flower "Iris-Virginica" is 0; the number of training instances whose values of the input attribute "Sepal Length" fall in the overlapping interval formed by the values of the input attribute "Sepal Length" for the species of flower "Iris-Versicolor" and the values of the input attribute "Sepal Length" for the species of flower "Iris-Virginica" is 8. From Table 1, we can see that 5 training instances belong to the species of flower "Iris-Setosa"; 5 training instances belong to the species of flower "Iris-Versicolor". Thus, the number of training instances belonging to the pair of species of flower "Iris-Setosa and Iris-Versicolor" is 10. Furthermore, we can see that 5 training instances belong to the species of flower "Iris-Setosa"; 5 training instances belong to the species of flower "Iris-Virginica". Thus, the number of training instances belonging to the pair of species of flower "Iris-Setosa and Iris-Virginica" is 10. Finally, we can see that 5 training instances belong to the species of flower "Iris-Versicolor"; 5 training instances belong to the species of flower "Iris-Virginica". Thus, the number of training instances belonging to the pair of species of flower "Iris-Versicolor and Iris-Virginica" is 10. Therefore, we can get (0 + 0 + 8) / (10 + 10 + 10) = 8/30 = 0.27. Because the overlapping threshold value given by the user is 0.2, the input attribute "Sepal Length" is discarded. In the same way, we can see that the input attribute "Sepal Width" is discarded; the input attribute "Petal Length" is used; the input attribute "Petal Width" is used.

[Step 4] Because $m = 4$, $\mu_1 = 0$, $\mu_2 = 0$, $\mu_3 = 1$, $\mu_4 = 1$, $\delta = 2$, $x_{max,1} = 7.1$, $x_{min,1} = 4.6$, $x_{max,2} = 3.6$, $x_{min,2} = 2.3$, $x_{max,3} = 6.0$, $x_{min,3} = 1.3$, $x_{max,4} = 2.5$, $x_{min,4} = 0.2$ and based on formula (7), we can calculate the distance $D(x_i, x_k)$ between each pair of training instances x_i and x_k in the training data set. For example, we can use formula (7) to calculate the distance $D(x_1, x_6)$ between the first training instance x_1 and the sixth training instance x_6 of Table 1, where x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm),

1) and x_6 = ((7.0 cm, 3.2 cm, 4.7 cm, 1.4 cm),2), shown as follows:

$$
\begin{aligned}
D(x_1, x_6) &= \sum_{j=1}^{4} \mu_j \left[(1 - \frac{|x_{1,j} - x_{6,j}|}{|x_{\max,j} - x_{\min,j}|}) / \delta \right] \\
&= 0 \times \left[(1 - \frac{|x_{1,1} - x_{6,1}|}{|x_{\max,1} - x_{\min,1}|}) / 2 \right] + 0 \times \left[(1 - \frac{|x_{1,2} - x_{6,2}|}{|x_{\max,2} - x_{\min,2}|}) / 2 \right] + \\
&\quad 1 \times \left[(1 - \frac{|x_{1,3} - x_{6,3}|}{|x_{\max,3} - x_{\min,3}|}) / 2 \right] + 1 \times \left[(1 - \frac{|x_{1,4} - x_{6,4}|}{|x_{\max,4} - x_{\min,4}|}) / 2 \right] \\
&= 0 + 0 + \left[(1 - \frac{|1.4 - 4.7|}{|6.0 - 1.3|}) / 2 \right] + \left[(1 - \frac{|0.2 - 1.4|}{|2.5 - 0.2|}) / 2 \right] \\
&= \left[(1 - \frac{3.3}{4.7}) / 2 \right] + \left[(1 - \frac{1.2}{2.3}) / 2 \right] \\
&= \left[(1 - 0.70) / 2 \right] + \left[(1 - 0.52) / 2 \right] \\
&= (0.30 / 2) + (0.48 / 2) \\
&= 0.15 + 0.24 \\
&= 0.39.
\end{aligned}
$$

Table 3. The number of training instances that overlap between each species of flowers

Input attribute	Pair of species of flowers	Number of training instances falls in the overlapping interval
Sepal Length (SL)	Iris-Setosa and Iris-Versicolor	0
	Iris-Setosa and Iris-Virginica	0
	Iris-Versicolor and Iris-Virginica	8
Sepal Width (SW)	Iris-Setosa and Iris-Versicolor	3
	Iris-Setosa and Iris-Virginica	6
	Iris-Versicolor and Iris-Virginica	6
Petal Length (PL)	Iris-Setosa and Iris-Versicolor	0
	Iris-Setosa and Iris-Virginica	0
	Iris-Versicolor and Iris-Virginica	0
Petal Width (PW)	Iris-Setosa and Iris-Versicolor	0
	Iris-Setosa and Iris-Virginica	0
	Iris-Versicolor and Iris-Virginica	0

In the same way, we can use formula (7) to calculate the distance $D(x_i, x_k)$ between each pair of training instances x_i and x_k, where $1 \le i \le 15$ and $1 \le k \le 15$, as shown in Table 4.

[Step 5] Based on formula (6), we can derive the fuzzy compatibility relation R from the training instances. For example, we can use formula (6) to calculate the fuzzy compatibility relation R (x_1, x_6) between the first training instance x_1 and the sixth training instance x_6, where x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm), 1) and x_6 = ((7.0 cm, 3.2 cm, 4.7 cm, 1.4 cm), 2), shown as follows:

$$\begin{aligned} R(x_1, x_6) &= 2^{D(x_1, x_6)} - 1 \\ &= 2^{0.39} - 1 \\ &= 0.31, \end{aligned}$$

where D (x_1, x_6) = 0.39 is obtained from Step 4.

In the same way, we can use formula (6) to calculate the fuzzy compatibility value R (x_i, x_k) between each pair of training instances x_i and x_k, where $1\le i \le 15$ and $1\le k \le 15$, as shown in Table 5.

[Step 6] Because the upper bound threshold value α and the lower bound threshold value β given by the user are 0.5 and 0.2, respectively, based on Table 5, the total fuzzy compatibility value TFCV(x_i) of each training instance x_i, where $1\le i \le 15$, can be obtained, shown as follows:

TFCV(x_1) = 9.50, TFCV(x_2) = 9.48,
TFCV(x_3) = 9.48, TFCV(x_4) = 9.40,
TFCV(x_5) = 9.48, TFCV(x_6) = 8.09,
TFCV(x_7) = 8.13, TFCV(x_8) = 8.29,
TFCV(x_9) = 6.82, TFCV(x_{10}) = 8.20,
TFCV(x_{11}) = 11.39, TFCV(x_{12}) = 12.32,
TFCV(x_{13}) = 12.33, TFCV(x_{14}) = 12.37,
TFCV(x_{15}) = 12.39.

Table 4. Distance between each pair of training instances

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	X_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}
x_1	1	1.00	0.99	0.99	1.00	0.39	0.39	0.35	0.48	0.38	0.01	0.24	0.11	0.21	0.10
x_2	1.00	1	0.99	0.99	1.00	0.39	0.39	0.35	0.49	0.38	0.01	0.24	0.11	0.21	0.10
x_3	0.99	0.99	1	0.98	0.99	0.38	0.38	0.33	0.47	0.37	0.00	0.28	0.10	0.19	0.09
x_4	0.99	0.99	0.98	1	0.99	0.40	0.40	0.36	0.49	0.39	0.02	0.25	0.12	0.22	0.11
x_5	1.00	1.00	0.99	0.99	1	0.39	0.39	0.35	0.48	0.38	0.01	0.24	0.11	0.21	0.10
x_6	0.39	0.39	0.38	0.40	0.39	1	0.96	0.96	0.90	0.97	0.62	0.85	0.72	0.82	0.71
x_7	0.39	0.39	0.38	0.40	0.39	0.96	1	0.96	0.90	0.99	0.62	0.85	0.72	0.82	0.71
x_8	0.35	0.35	0.33	0.36	0.35	0.96	0.96	1	0.86	0.97	0.67	0.89	0.76	0.86	0.75
x_9	0.48	0.48	0.47	0.49	0.48	0.90	0.90	0.86	1	0.89	0.53	0.75	0.62	0.72	0.61
x_{10}	0.38	0.38	0.37	0.39	0.38	0.97	0.99	0.97	0.89	1	0.63	0.86	0.73	0.83	0.72
x_{11}	0.01	0.01	0.00	0.02	0.01	0.62	0.62	0.67	0.53	0.63	1	0.77	0.90	0.81	0.91
x_{12}	0.24	0.24	0.28	0.25	0.24	0.85	0.85	0.89	0.75	0.86	0.77	1	0.87	0.93	0.86
x_{13}	0.11	0.11	0.10	0.12	0.11	0.72	0.72	0.76	0.62	0.73	0.90	0.87	1	0.90	0.97
x_{14}	0.21	0.21	0.19	0.22	0.21	0.82	0.82	0.86	0.72	0.83	0.81	0.93	0.90	1	0.89
x_{15}	0.10	0.10	0.09	0.11	0.10	0.71	0.71	0.75	0.61	0.72	0.91	0.86	0.97	0.89	1

[Step 7] For the species of flower "Iris-Setosa", because TFCV(x_1) has the largest value among the values of TFCV(x_1), TFCV(x_2), … , and TFCV(x_5), the system lets the training instance x_1 be a cluster center, where x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm), 1). Because the values of R(x_1, x_1), R(x_1, x_2), R(x_1, x_3), R(x_1, x_4) and R(x_1, x_5) are larger than the level threshold value 0.7 given by the user, the system marks the training instances x_1, x_2, x_3, x_4 and x_5 from the training data set. Because all of the training instances for the species of flower "Iris-Setosa" in the training data set have been marked, we can see that the training instance x_1 is the cluster center of the training instances for the species of flower "Iris-Setosa".

For the species of flower "Iris-Versicolor", because TFCV(x_8) has the largest value among the values of TFCV(x_6), TFCV(x_7), … , and TFCV(x_{10}), the system lets the training instance x_8 be a cluster center, where x_8 = ((6.9 cm, 3.1 cm, 4.9 cm, 1.5 cm), 2). Because the values of R(x_8, x_6), R(x_8, x_7), R(x_8, x_8), R(x_8, x_9) and R(x_8, x_{10}) are larger than the level threshold value 0.7 given by the user, the system marks the training instances x_6, x_7, x_8, x_9 and x_{10} from the training data set. Because all of the training instances for the species of flower "Iris-Versicolor" in the training data set have been marked, we can see that the training instance x_8 is the cluster center of the training instances for the species of flower "Iris-Versicolor".

For the species of flower "Iris-Virginica", because TFCV(x_{14}) has the largest value among the values of TFCV(x_{11}), TFCV(x_{12}), … , and TFCV(x_{15}), the system lets the training instance x_{14} be a cluster center, where x_{14} = ((6.3 cm, 2.9 cm, 5.6 cm, 1.8 cm), 3). Because the values of R(x_{14}, x_{11}), R(x_{14}, x_{12}), R(x_{14}, x_{13}), R(x_{14}, x_{14}) and R(x_{14}, x_{15}) are larger than the level threshold value 0.7 given by the user, the system marks the training instances x_{11}, x_{12}, x_{13}, x_{14} and x_{15} from the training data set. Because all of the training instances for the species of flower "Iris-Virginica" in the training data set have been marked, we can see that the training instance x_{14} is the cluster center of the training instances for the species of flower "Iris-Virginica".

In summary, because all of the training instances in the training data set have been marked, we can see that the training instances x_1, x_8 and x_{14} are the cluster centers of the training instances. It should be noted that if the number of training instances in a cluster is less than 5 percent of the number of training instances, the cluster center will not be used as the cluster centers of the training instances. Because there are 15 training instances, therefore, if the number of training instances in a cluster is less than 1 instance, the cluster center will not be used as the cluster center of the training instances. Therefore, each generated cluster center contains five training instances, the cluster centers x_1, x_8 and x_{14} will be used as the cluster centers of the training instances.

Then, based on the generated cluster centers x_1, x_8 and x_{14}, we can classify the testing instances shown in Table 1 as follows:

(i) Based on Table 5, we can find the largest fuzzy compatibility value between the testing instance x_1 and the cluster centers x_1, x_8, x_{14}, where x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm), y_1), x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm), 1), x_8 = ((6.9 cm, 3.1 cm, 4.9 cm, 1.5 cm), 2) and x_{14} = ((6.3 cm, 2.9 cm, 5.6 cm, 1.8 cm), 3). From Table 5, we can see that R(x_1, x_1) = 1, R(x_1, x_8) = 0.27, R(x_1, x_{14}) = 0.15, where R(x_1, x_1) is the largest fuzzy compatibility value among them. Thus, the testing instance x_1 is in the same cluster as the cluster center x_1. That is, the classification result of the testing instance x_1 is belonging to "Iris-Setosa".

(ii) Based on Table 5, we can find the largest fuzzy compatibility value between the testing instance x_2 and the cluster centers x_1, x_8, x_{14}, where x_2 = ((4.9 cm, 3.0 cm, 1.4 cm, 0.2 cm), y_2), x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm), 1),

x_8 = ((6.9 cm, 3.1 cm, 4.9 cm, 1.5 cm), 2) and x_{14} = ((6.3 cm, 2.9 cm, 5.6 cm, 1.8 cm), 3).

(iii) From Table 5, we can see that R(x_2, x_1) = 1, R(x_2, x_8) = 0.27 and R(x_2, x_{14}) = 0.15, where R(x_2, x_1) is the largest fuzzy compatibility value among them. Thus, the testing instance x_2 is in the same cluster as the cluster center x_1. That is, the classification result of the testing instance x_2 is belonging to "Iris-Setosa".

⋮

(xiii) Based on Table 5, we can find the largest fuzzy compatibility value between the testing instance x_{13} and the cluster centers x_1, x_8, x_{14}, where x_{13} = ((7.1 cm, 3.0 cm, 5.9 cm, 2.1 cm), y_{13}), x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm), 1), x_8 = ((6.9 cm, 3.1 cm, 4.9 cm, 1.5 cm), 2) and x_{14} = ((6.3 cm, 2.9 cm, 5.6 cm, 1.8 cm), 3). From Table 5, we can see that R(x_{13}, x_1) = 0.07, R(x_{13}, x_8) = 0.70 and R(x_{13}, x_{14}) = 0.87, where R(x_{13}, x_{14}) is the largest fuzzy compatibility value among them. Thus, the testing instance x_{13} is in the same cluster as cluster center x_{14}. That is, the classification result of the testing instance x_{13} is belonging to "Iris-Virginica".

(xiv) Based on Table 5, we can find the larg est fuzzy compatibility value between the testing instance x_{14} and the cluster centers x_1, x_8, x_{14}, where x_{14} = ((6.3 cm, 2.9 cm, 5.6 cm, 1.8 cm), y_{14}), x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm), 1), x_8 = ((6.9 cm, 3.1 cm, 4.9 cm, 1.5 cm), 2) and x_{14} = ((6.3 cm, 2.9 cm, 5.6 cm, 1.8 cm), 3). From Table 5, we can see that R(x_{14}, x_1) = 0.15, R(x_{14}, x_8) = 0.82 and R(x_{14}, x_{14}) = 1, where R(x_{14}, x_{14}) is the largest fuzzy compatibility value among them. Thus, the testing instance x_{14} is in the same cluster as the cluster center x_{14}. That is, the classification result of the testing instance x_{14} is belonging to "Iris-Virginica".

Table 5. Fuzzy compatibility relation between each pair of training instances

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}
x_1	1	1.00	0.99	0.99	1.00	0.31	0.31	0.27	0.40	0.30	0.01	0.18	0.07	0.15	0.07
x_2	1.00	1	0.99	0.99	1.00	0.31	0.31	0.27	0.40	0.30	0.01	0.18	0.07	0.15	0.07
x_3	0.99	0.99	1	0.97	0.99	0.30	0.30	0.26	0.39	0.29	0.00	0.17	0.07	0.14	0.06
x_4	0.99	0.99	0.97	1	0.99	0.32	0.32	0.28	0.41	0.31	0.02	0.19	0.09	0.14	0.08
x_5	1.00	1.00	0.99	0.99	1	0.31	0.31	0.27	0.40	0.30	0.01	0.18	0.08	0.15	0.07
x_6	0.31	0.31	0.30	0.32	0.31	1	0.94	0.94	0.87	0.96	0.54	0.80	0.65	0.76	0.63
x_7	0.31	0.31	0.30	0.32	0.31	0.94	1	0.94	0.87	0.99	0.54	0.80	0.65	0.76	0.64
x_8	0.27	0.27	0.26	0.28	0.27	0.94	0.94	1	0.82	0.96	0.59	0.86	0.70	0.82	0.68
x_9	0.40	0.40	0.39	0.41	0.40	0.87	0.87	0.82	1	0.86	0.44	0.68	0.54	0.65	0.53
x_{10}	0.30	0.30	0.29	0.31	0.30	0.96	0.99	0.96	0.86	1	0.55	0.81	0.66	0.78	0.65
x_{11}	0.01	0.01	0.00	0.02	0.01	0.54	0.54	0.59	0.44	0.55	1	0.71	0.87	0.75	0.88
x_{12}	0.18	0.18	0.17	0.19	0.18	0.80	0.80	0.86	0.68	0.81	0.71	1	0.83	0.90	0.82
x_{13}	0.07	0.07	0.07	0.09	0.08	0.65	0.65	0.70	0.54	0.66	0.87	0.83	1	0.87	0.96
x_{14}	0.15	0.15	0.14	0.16	0.15	0.76	0.76	0.82	0.65	0.78	0.75	0.90	0.87	1	0.86
x_{15}	0.07	0.07	0.06	0.08	0.07	0.63	0.64	0.68	0.53	0.65	0.88	0.82	0.96	0.86	1

(xv) Based on Table 5, we can find the largest fuzzy compatibility value between the testing instance x_{15} and the cluster centers x_1, x_8, x_{14}, where x_{15} = ((6.5 cm, 3.0 cm, 5.8 cm, 2.2 cm), y_{15}), x_1 = ((5.1 cm, 3.5 cm, 1.4 cm, 0.2 cm), 1), x_8 = ((6.9 cm, 3.1 cm, 4.9 cm, 1.5 cm), 2) and x_{14} = ((6.3 cm, 2.9 cm, 5.6 cm, 1.8 cm), 3).

From Table 5, we can see that $R(x_{15}, x_1) = 0.07$, $R(x_{15}, x_8) = 0.68$ and $R(x_{15}, x_{14}) = 0.86$, where $R(x_{15}, x_{14})$ is the largest fuzzy compatibility value among them. Thus, the testing instance x_{15} is in the same cluster as cluster center x_{14}. That is, the classification result of the testing instance x_{15} is belonging to "Iris-Virginica".

In summary, the classification results of this illustrate example are as follows:

Cluster 1(Iris-Setosa) = $\{x_1, x_2, x_3, x_4, x_5\}$, Cluster Center = x_1;

Cluster 2(Iris-Versicolor) = $\{x_6, x_7, x_8, x_9, x_{10}\}$, Cluster Center = x_8;

Cluster 3(Iris-Virginica) = $\{x_{11}, x_{12}, x_{13}, x_{14}, x_{15}\}$, Cluster Center = x_{14}.

From Table 1, we can see that the classification accuracy rate of this example is 100%.

5. Experimental results

Based on the proposed algorithm, we have Implemented a program on a Pentium 4 PC by using Visual Basic Version 6.0 to deal with the Iris data classification problem. Let us consider the following cases:

Case 1: The training data set contains 150 training instances (i.e., the whole Iris data set) and the testing data set is equal to the training data set containing the same 150 training instances. After executing the program 200 times, the average classification accuracy rate is 97.33%, where the upper bound threshold value $\alpha = 0.8$, the lower bound threshold value $\beta = 0.1$, the level threshold value $\gamma = 0.7$, and the overlapping threshold value $\lambda = 0.2$.

Case 2: The system randomly chooses 120 instances from the Iris data as the training data set and lets other 30 instances of the Iris data be the testing data set. After executing the program 200 times, the average classification accuracy rate is 96.65%, where the upper bound threshold value $\alpha = 0.6$, the lower bound threshold value $\beta = 0.1$, the level threshold value $\gamma = 0.9$, and the overlapping threshold value $\lambda = 0.2$.

Case 3: The system randomly chooses 75 instances from the Iris data as the training data set and lets the other 75 instances of the Iris data be the testing data set. After executing the program 200 times, the average classification accuracy rate is 96.24%, where the upper bound threshold value $\alpha = 0.5$, the lower bound threshold value $\beta = 0.2$, the level threshold value $\gamma = 0.7$, and the overlapping threshold value $\lambda = 0.2$.

A comparison of the average classification accuracy rate of the proposed method with that of Hong-and-Chen's method [8], Hong-and-Lee's method [9], Hong-and-Lee's method [10], Wang's method [16], Wu-and-Chen's method [17], Chang-and-Chen's method [1] and Castro's method [2] is shown in Table 6.

6. Conclusions

In this paper, we have presented a new method to deal with the Iris data [5] classification problem based on the concept of fuzzy compatibility relations to find the cluster centers of the training instances. From Table 6, we can see that the proposed method is better than the existing methods due to the fact that it can get a higher average classification accuracy rate to deal with the Iris data classification problem than the existing methods. The proposed method also can be generalized to deal with other classification problems. In this paper, the upper bound threshold value α, the lower bound threshold value β, the level threshold value γ and the overlapping threshold value λ are given by the user, where $\alpha \in$

[0, 1], β∈ [0, 1], γ ∈ [0, 1] and λ∈ [0, 1]. In the future, we will develop an automatic learning mechanism to automatically derive the optimal values of α, β, γ and λ to get a higher average classification accuracy rate for dealing with fuzzy classification problems.

Table 6. A Comparison of the average classification accuracy rate for different methods

Methods	Average classification accuracy rate
Hong-and-Lee's method [9] (Training Data Set: 75 Instances; Testing Data Set: 75 Instances; Executing 200 Times)	95.57%
Hong-and-Lee's method [10] (Training Data Set: 75 Instances; Testing Data Set: 75 Instances; Executing 200 Times)	95.57%
Chang-and-Chen’s method [1] (Training Data Set: 75 Instances; Testing Data Set: 75 Instances; Executing 200 Times)	96.07%
Wu-and-Chen's method [17] (Training Data set: 75 Instances; Testing Data Set: 75 Instances; Executing 200 Times)	96.21%
The Proposed method (Training Data Set: 75 Instances; Testing Data Set: 75 Instances; Executing 200 Times; The Upper Bound Threshold Value α = 0.5; The Lower Bound Threshold Value β = 0.2; The Level Threshold Value γ = 0.7; The Overlapping Threshold Value λ = 0.2)	96.24%
Castro's et al. method [2] (Training Data Set: 120 Instances; Testing Data Set: 30 Instances; Executing 10 Times)	96.60%
The Proposed method (Training Data set: 120 Instances; Testing Data set: 30 Instances; Executing 200 Times; The Upper Bound Threshold Value α = 0.6; The Lower Bound Threshold Value β = 0.1; The Level Threshold Value γ = 0.9; The Overlapping Threshold Value λ = 0.2)	96.65%
Hong-and-Chen's method [7] (Training Data Set: 150 Instances; Testing Data Set: 150 Instances)	96.67%
Hong-and-Chen's method [8] (Training Data Set: 150 Instances; Testing Data Set: 150 Instances)	97.33%
Wang’s method [16] (Training Data Set: 150 Instances; Testing Data Set: 150 Instances)	97.33%
The proposed method (Training Data Set: 150 Instances; Testing Data Set: 150 Instances; Executing 200 Times; The Upper Bound Threshold Value α = 0.8; The Lower Bound Threshold Value β = 0.1; The Level Threshold Value γ = 0.7; The Overlapping Threshold Value λ = 0.2)	97.33%

Acknowledgements

This work was supported in part by the National Science Council, Republic of China, under Grant NSC 91-2213-E-011-037.

References

[1] Chang, C. H. and Chen, S. M. 2001. Constructing membership functions and generating weighted fuzzy rules from training data. *Proceedings of the 2001 Ninth National Conference on Fuzzy Theory and Its Applications*, Chungli, Taoyuan, Taiwan, Republic of China: 708-713.

[2] Castro, J. L., Castro-Schez, J. J., and Zurita, J. M. 1999. Learning maximal

structure rules in fuzzy logic for knowledge acquisition in expert systems. *Fuzzy Sets and Systems,* 101: 331-342.

[3] Chen, S. M. and Yeh, M. S. 1998. Generating fuzzy rules from relational database systems for estimating null values. *Cybernetics and Systems: An International Journal*, 29: 363-376.

[4] Chen, Y. C. and Chen, S. M. 2001. Constructing membership functions and generating fuzzy rules using genetic algorithms. *Proceedings of the 2001 Ninth National Conference on Fuzzy Theory and Its Applications*, Chungli, Taoyuan, Taiwan, Republic of China: 195-200.

[5] Fisher, R. 1936. The use of multiple measurements in taxonomic problems. *Annales of Eugenics,* 7: 179-188.

[6] Giarratano, J. and Riley, G. 1994. "*Expert Systems: Principles and Programming*". PWS Publishing Company, Boston.

[7] Hong, T. P. and Chen, J. B. 1999. Finding relevant attributes and membership functions. *Fuzzy Sets and Systems*, 103: 389-404.

[8] Hong, T. P. and Chen, J. B. 2000. Processing individual fuzzy attributes for fuzzy rule induction. *Fuzzy Sets and Systems*, 112: 127-140.

[9] Hong, T. P. and Lee, C. Y. 1996. Induction of fuzzy rules and membership functions from training example. *Fuzzy Sets and Systems*, 84: 33-47.

[10] Hong, T. P. and Lee, C. Y. 1999. Effect of merging order on performance of fuzzy induction. Intelligent Data Analysis, 3: 139-151.

[11] Kao, C. M. and Chen, S. M. 2000. A new method to generate fuzzy rules from training data containing noise for handling classification problems. *Proceedings of the Fifth Conference on Artificial Intelligence and Applications*, Taipei, Taiwan, Republic of China: 324-332.

[12] Klir, G. J. and Yuan, B. 1995. "*Fuzzy Sets and Fuzzy Logic Theory and Applications*". Prentice-Hall, Englewood Cliffs.

[13] Lin, H. L. and Chen, S. M. 2001. A new method for generating weighted fuzzy rules from training instances using genetic algorithms. *Proceedings of the 6th Conference on Artificial Intelligence and Applications*, Kaohsiung, Taiwan, Republic of China: 628-633.

[14] Sugeno, M. and Kang, G. T. 1988. Structure identification of fuzzy model. *Fuzzy Sets and Systems*, 28: 15-33.

[15] Takagi, T. and Sugeno, M. 1985. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15: 116-132.

[16] Wang, C. H., Liu, J. F., Hong, T. P., and Tseng, S. S. 1999. A fuzzy inductive strategy for modular rules. *Fuzzy Sets and Systems*, 103: 91-105.

[17] Wu, T. P. and Chen, S. M. 1999. A new method for constructing membership functions and fuzzy rules from training examples. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 29: 25-40.

[18] Yu, C. H. and Chen, S. M. 2002. A new method for handling fuzzy classification problems based on clustering techniques. *Proceedings of the Seventh Conference on Artificial Intelligence and Applications*, Taichung, Taiwan, Republic of China: 107-112.

[19] Zadeh, L. A. 1965. Fuzzy sets. *Information and Control*, 8: 338-353.

[20] Zadeh, L. A. 1988. Fuzzy logic. *IEEE Computer*, 21: 83-91.

[21] Zadeh, L. A. 1975. The concepts of a linguistic variable and its application to approximate reasoning – I. *Information Sciences*, 8: 199-249.