A fuzzy classifier with ellipsoidal regions for diagnosis problems

Abe, Shigeo

Thawonmas, Ruck

Kayama, Masahiro

(Citation)
IEEE transactions on systems, man, and cybernetics. Part C, Applications and reviews,29(1):140-149

(Issue Date)
1999-02

(Resource Type)
journal article

(Version)
Version of Record

(URL)
https://hdl.handle.net/20.500.14094/90000217

Correspondence

A Fuzzy Classifier with Ellipsoidal Regions for Diagnosis Problems

Shigeo Abe, Ruck Thawonmas, and Masahiro Kayama

***Abstract*—In our previous work, we developed a fuzzy classifier with ellipsoidal regions that has a training capability. In this paper, we extend the fuzzy classifier to diagnosis problems, in which the training data belonging to abnormal classes are difficult to obtain while the training data belonging to normal classes are easily obtained. Assuming that there are no data belonging to abnormal classes, we first train the fuzzy classifier with only the data belonging to normal classes. We then introduce the threshold of the minimum-weighted distance from the centers of the clusters for the data belonging to normal classes. If the unknown datum is within the threshold, we classify the datum into normal classes and, if not, abnormal classes. The operator checks whether the diagnosis is correct. If the incoming datum is classified into the same normal class both by the classifier and the operator, nothing is done. But if the input datum is classified into the different normal classes by the classifier and the operator, or if the incoming datum is classified into an abnormal class, but the operator classified it into a normal class, the slopes of the membership functions of the fuzzy rules are tuned. If the operator classifies the datum into an abnormal class, the classifier is retrained adding the newly obtained datum irrespective of the classifier's classification result. The online training is continued until a sufficient number of the data belonging to abnormal classes are obtained. Then the threshold is optimized using the data belonging to both normal and abnormal classes. We evaluate our method using the Fisher iris data, blood cell data, and thyroid data, assuming some of the classes are abnormal. We show that, for the Fisher iris data and blood cell data, more than a 90% recognition rate is obtained, even if there are no training data belonging to abnormal classes.**

***Index Terms*— Blood cell data, diagnosis problems, Fisher iris data, fuzzy classifiers, membership functions, neural networks, rule extraction, thyroid data, tuning.**

I. Introduction

To overcome the problem of intractability in multilayered neural network classifiers, several fuzzy classifiers with a training capability have been proposed [1]–[6]. They are categorized by the shapes of their fuzzy regions as follows:

1) classifiers with ellipsoidal regions [2], [3];
2) those with hyperbox regions [4], [5];
3) those with polyhedron regions [6].

The advantage of these fuzzy classifiers, in addition to tractability, over multilayered neural network classifiers is a fast training ability.

The generalization ability of multilayered neural network classifiers depends on the training method used and the initial values of weights. But, in general, multilayered neural network classifiers have a robust generalization ability over a wide range of applications. Some of the fuzzy classifiers lack robustness, but in [3] the generalization ability of the fuzzy classifier with ellipsoidal regions was shown to be comparable to the maximum generalization ability of neural network classifiers when the training data did not include discrete input variables.

A diagnosis problem in which a system state is diagnosed as normal or abnormal is one type of classification problem, and diagnosis is one of the fields to which multilayered neural network classifiers and fuzzy classifiers are applied. At the initial stage of classifier development, usually the data belonging to abnormal classes are difficult to obtain, although the data belonging to normal classes are readily available. But without a sufficient number of the data belonging to abnormal classes, we cannot obtain a neural network classifier or a fuzzy classifier with a sufficient generalization ability. To overcome this problem, we usually generate training data that belong to abnormal classes, adding noises to the data belonging to normal classes [7] or using experts' knowledge.

In this paper, we take a different approach using the fuzzy classifier with ellipsoidal regions: we use online training. Assuming that there are no data belonging to abnormal classes, we first train the fuzzy classifier using only the data belonging to normal classes. We then introduce the threshold of the minimum-weighted distance from the centers of the clusters for the data belonging to normal classes. If the unknown datum is within the threshold, we classify the datum into a normal class and, if not, into an abnormal class. The operator checks whether the diagnosis is correct. If the incoming datum is classified into the same normal class both by the classifier and the operator, nothing is done. But if the input datum is classified into the different normal classes by the classifier and the operator, or if the incoming datum is classified into an abnormal class but the operator classifies it into a normal class, the slopes of the membership functions of the fuzzy rules are tuned. If the operator classifies the datum into an abnormal class, the classifier is retrained adding the newly obtained datum, irrespective of the classifier's classification result. The online training is continued until a sufficient number of the data belonging to abnormal classes are obtained. Then the threshold is optimized using the data belonging to normal and abnormal classes so that the recognition rate is maximized.

In Section II, we introduce the threshold to the fuzzy classifier with ellipsoidal regions. In Sections III–V, we discuss fuzzy rule generation, fuzzy rule tuning, and threshold optimization. In Section VI, we describe online training of the classifier dividing the training process into two stages. In Section VII, using the Fisher iris data, blood cell data, and thyroid data, we evaluate the proposed fuzzy classifier and compare its performance with that of the neural network classifier and the fuzzy classifier with hyperbox regions [5] when a sufficient number of the data belonging to abnormal classes are available.

II. Fuzzy Classifier with Threshold

In a typical diagnosis problem, we may classify the system state into one normal class and one abnormal class. But in other types of problems, there may be several normal and abnormal classes. Since the data belonging to normal classes are easily obtained, it is relatively easy to determine how many normal classes we should consider. But since the data belonging to abnormal classes are difficult to obtain, it is not easy to determine the number of abnormal classes without obtaining a sufficient number of the data belonging to abnormal classes. The most important thing in diagnosis problems is whether

Manuscript received January 19, 1996; revised December 3, 1996 and April 10, 1998.

S. Abe is with the Department of Electrical and Electronics Engineering, Kobe University, Kobe, 657 Japan (e-mail: abe@eedept.kobe-u.ac.jp).

R. Thawonmas is with the University of Aizu, Aizu-Wakamatsu, 965-80 Japan.

M. Kayama is with Omika Works, Hitachi, Ltd., Hitachi, 319-12 Japan.

Publisher Item Identifier S 1094-6977(99)00104-2.

the system in consideration is in the normal or abnormal state. Therefore, here we assume that we know the number of normal classes, but do not know the number of abnormal classes. Thus, in the following, we assume that there is only one abnormal class.

We consider classification of an M-dimensional input vector $\mathbf{x}$ into $n-1$ normal classes and one abnormal class. Assume that class i ($i = 1, \ldots, n$) is divided into several clusters ij ($j = 1, \ldots$), in which cluster ij denotes the jth cluster for class i. For each cluster ij, we define the following fuzzy rule:

$$R_{ij}: \quad \text{If } \mathbf{x} \text{ is } \mathbf{c}_{ij} \text{ then } \mathbf{x} \text{ belongs to class } i \tag{1}$$

where $\mathbf{c}_{ij}$ is the center of cluster ij. The membership function $m_{ij}(\mathbf{x})$ of (1) for input $\mathbf{x}$ is given by

$$m_{ij}(\mathbf{x}) = \exp\left(-h_{ij}^2(\mathbf{x})\right) \tag{2}$$

$$h_{ij}^2(\mathbf{x}) = \frac{d_{ij}^2(\mathbf{x})}{\alpha_{ij}} \tag{3}$$

$$d_{ij}^2(\mathbf{x}) = (\mathbf{x} - \mathbf{c}_{ij})^t Q_{ij}^{-1} (\mathbf{x} - \mathbf{c}_{ij}) \tag{4}$$

where $d_{ij}(\mathbf{x})$ is the weighted distance between $\mathbf{x}$ and $\mathbf{c}_{ij}$, $h_{ij}(\mathbf{x})$ is the tuned distance, α_{ij} (> 0) is the tuning parameter for cluster ij, $\mathbf{c}_{ij} = (c_{ij,1}, \ldots, c_{ij,M})^t$, Q_{ij} is the $M \times M$ covariance matrix of cluster ij, t denotes the transpose of a matrix, and -1 denotes the inverse of a matrix.

The center $\mathbf{c}_{ij}$ and the covariance matrix Q_{ij} are calculated using the data belonging to cluster ij, assuming that clustering is 100% correct. Namely, we do not assume a weight for each datum. After we calculate $\mathbf{c}_{ij}$ and Q_{ij}, using the membership function $m_{ij}(\mathbf{x})$ given by (2), we know the weight of each datum. The center $\mathbf{c}_{ij}$ is calculated by calculating the average values of the data belonging to cluster ij

$$c_{ij,k} = \frac{1}{N_{ij}} \sum_{\mathbf{x} \in \text{cluster } ij} x_k \tag{5}$$

where N_{ij} is the number of the data belonging to cluster ij.

The covariance matrix Q_{ij} is calculated by

$$Q_{ij} = \frac{1}{N_{ij}} \sum_{\mathbf{x} \in \text{cluster } ij} (\mathbf{x} - \mathbf{c}_{ij})(\mathbf{x} - \mathbf{c}_{ij})^t. \tag{6}$$

If the covariance matrix Q_{ij} is singular, we set all the off-diagonal elements of Q_{ij} to zero so that Q_{ij} becomes regular. By this setting, the axes of the ellipsoidal regions are parallel to the input variables. But since the kth diagonal element of Q_{ij} is the variance of x_k belonging to cluster ij, the diagonalized Q_{ij} is still a good estimate of the covariance matrix. In the extreme case, if only one datum belongs to cluster ij, Q_{ij} is a zero matrix. In this case, we set a small value to the diagonal elements of Q_{ij}.

Assuming that Q_{ij} is regular, Q_{ij} is a positive definite matrix. Then it is easy to see that the mean square weighted distance is M

$$\frac{1}{N_{ij}} \sum_{\mathbf{x} \in \text{cluster } ij} d_{ij}^2(\mathbf{x}) = M. \tag{7}$$

Now we introduce the threshold t_h (> 0) for the normal classes. If for $\mathbf{x}$, the membership function for cluster ij ($1 \geq i \geq n-1$), i.e., $m_{ij}(\mathbf{x})$, is the largest among n classes and

$$\frac{d_{ij}^2(\mathbf{x})}{M} \leq t_h \tag{8}$$

is satisfied, $\mathbf{x}$ is classified into the normal class i. If $m_{ij}(\mathbf{x})$ ($1 \leq i \leq n-1$) is the largest but (8) is not satisfied, $\mathbf{x}$ is classified into the abnormal class n. If $m_{nj}(\mathbf{x})$ is the largest, $\mathbf{x}$ is classified into the abnormal class n irrespective of the value of the weighted distance $d_{ij}(\mathbf{x})$. Note that to exploit (7), we use the weighted distance, not the tuned distance, in (8). Otherwise, we need to set different thresholds to different clusters for normal classes.

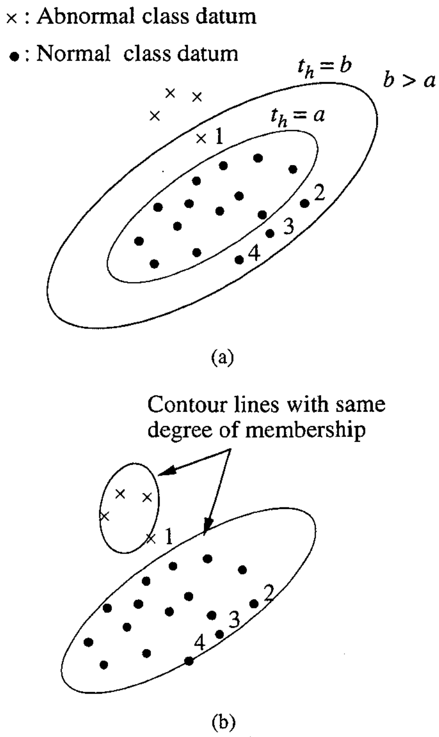

Fig. 1. Classification by a fuzzy classifier with ellipsoidal regions. (a) With a threshold but without fuzzy rules for the abnormal class. (b) With a fuzzy rule for the abnormal class.

Introducing the threshold, we can classify the data belonging to the abnormal class, even if there are no fuzzy rules for the abnormal class. But to improve the recognition rate, we need to generate fuzzy rules for the abnormal class when we get the data belonging to the abnormal class. Consider the two-dimensional (2-D) case shown in Fig. 1(a). When the threshold is a, the datum 1 belonging to the abnormal class is correctly classified into the abnormal class but data 2–4 that belong to the normal class are misclassified into the abnormal class. If we increase the threshold to b, data 2–4 belonging to the normal class are correctly classified, but datum 1 belonging to the abnormal class is now misclassified into the normal class. Thus, in this case, we cannot correctly classify all the data that belong to the normal and abnormal classes. This means that, since by the threshold the class boundary between normal and abnormal classes is defined by an ellipsoid, the data belonging to the normal class that are outside of the boundary but are far away from the abnormal class region are classified into the abnormal class. This can be avoided, as shown in Fig. 1(b), if we define a fuzzy rule for the data belonging to the abnormal class and adjust the tuning parameters α_{ij} so that the data belonging to the normal and abnormal classes are correctly classified. In this case, we may further introduce a threshold with a large value to improve reliability of classification.

III. Fuzzy Rule Generation

Here we discuss adaptive fuzzy rule generation. There are many clustering techniques; most of them are iterative [2], [8]–[11]. In the following, we use a clustering technique that is similar to the method discussed in [2]. The major difference is that we do not consider the conflict between different classes since overlapping between different classes is resolved by tuning α_{ij} after fuzzy rule generation. The method discussed here is that, if a new datum is within a specified

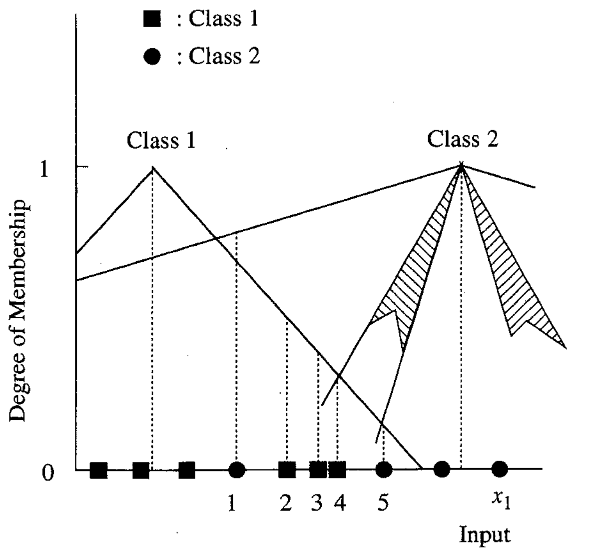

Fig. 2. Concept of tuning. If the slope of the membership function for class 2 is increased so that the resulting function lies between the shaded regions, datum 1 is misclassified, but data 2–4 are correctly classified.

distance from the center of the cluster to which the datum belongs, that datum is included in the cluster. If there is no such cluster, we generate a new cluster.

Suppose we have the fuzzy rules R_{ij} $(j = 1, \dots, n_i)$ for class i and a new datum $\mathbf{x}$ belonging to class i, where n_i is the number of fuzzy rules for class i already generated. If

$$\min_{i,j} \|\mathbf{x} - \mathbf{c}_{ij}\| \le \sigma_M \tag{9}$$

is satisfied for cluster k $(k \in \{1, \dots, n_i\})$, we recalculate $\mathbf{c}_{ik}$ and Q_{ik} using $\mathbf{x}$ and the data belonging to cluster k, where σ_M is the maximum radius of the cluster and $\|\cdot\|$ is the Euclidean distance. The reason why we do not use the weighted distance is that, when there is an insufficient number of data, the associated covariance matrix calculated by (6) may not approximate the true covariance matrix.

If (9) is not satisfied, we generate a fuzzy rule R_{i,n_i+1} with

$$\mathbf{c}_{i,n_i+1} = \mathbf{x} \tag{10}$$

$$Q_{i,n_i+1} = \begin{bmatrix} \varepsilon & & 0 \\ & \ddots & \\ 0 & & \varepsilon \end{bmatrix} \tag{11}$$

where ε is a small positive value. The covariance matrix Q_{i,n_i+1} given by (11) is a rough estimate. Thus, in tuning α_{ij}, as discussed in the following section, Q_{i,n_i+1} is first tuned. Then all α_{ij} are tuned.

IV. Fuzzy Rule Tuning

Tuning of the tuning parameters α_{ij} is discussed in [3]. Here, we summarize the procedure of tuning, and we discuss the detailed procedure in the Appendix.

To explain the concept of tuning, we consider a two-class case with one rule for each class, as shown in Fig. 2. (In the figure, instead of the Gaussian function, we use the triangular function as the membership function.) Datum 1 is correctly classified into class 2, while data 2–4 are misclassified into class 2. If we increase α_{11} or decrease α_{21}, datum 1 is first misclassified, but if we allow datum 1 to be misclassified, we can make data 2–4 be correctly classified. Fig. 2 shows this when α_{21} is decreased so that the degree of membership for class 2 lies between the shaded regions. Then, by allowing one datum to be misclassified, three data are correctly classified, i.e., the recognition rate is improved by two data.

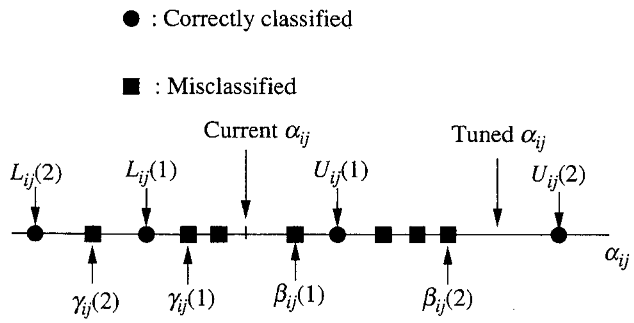

Fig. 3. Determination of tuned α_{ij}. If the current α_{ij} is modified to the tuned α_{ij} in $(\beta_{ij}(2), U_{ij}(2))$, one correctly classified datum is misclassified, but four misclassified data are correctly classified.

Now, suppose we tune the tuning parameter α_{ij}. Up to some value, we can increase or decrease α_{ij} without making the correctly classified data belonging to class i be misclassified. Now let $U_{ij}(1)$ and $L_{ij}(1)$ denote the upper and lower bounds that do not make the correctly classified data be misclassified, respectively. Likewise, $U_{ij}(l)$ and $L_{ij}(l)$ denote the upper and lower bounds in which $l-1$ correctly classified data are misclassified, respectively. Then, for instance, if we set a value in the interval $[U_{ij}(1), U_{ij}(2))$ to α_{ij}, one correctly classified datum belonging to class i is misclassified, where $[a, b]$ and (a, b) denote the closed and open intervals, respectively.

Similarly, if we increase or decrease α_{ij}, misclassified data may be correctly classified. Let $\beta_{ij}(l)$ denote the upper bound of α_{ij} that is smaller than $U_{ij}(l)$, and that makes the previously misclassified data be correctly classified. And $\gamma_{ij}(l)$ denotes the lower bound of α_{ij} that is larger than $L_{ij}(l)$, and that makes the previously misclassified data be correctly classified. Fig. 3 shows an example. If we change the current α_{ij} to the tuned α_{ij} in $(\beta_{ij}(2), U_{ij}(2))$, one correctly classified datum is misclassified but four misclassified data are correctly classified.

Then the next task is to find which interval among $(L_{ij}(l), \gamma_{ij}(l))$ and $(\beta_{ij}(l), U_{ij}(l))$ $(l = 1, \dots)$ gives the maximum recognition rate. To limit the search space, we introduce the maximum l, i.e., l_M. Let $(L_{ij}(l), \gamma_{ij}(l))$ be the interval that gives the maximum recognition rate for the training data among $(L_{ij}(k), \gamma_{ij}(k))$ and $(\beta_{ij}(k), U_{ij}(k))$ for $k = 1, \dots, l_M$. Then even if we set any value in the interval to α_{ij}, the recognition rate for the training data does not change but the recognition rate for the test data may change. To control the generalization ability, we set α_{ij} as follows:

$$\alpha_{ij} = \beta_{ij}(l) + \delta(U_{ij}(l) - \beta_{ij}(l)) \tag{12}$$

for $(\beta_{ij}(l), U_{ij}(l))$, where δ satisfies $0 < \delta < 1$ and

$$\alpha_{ij} = \gamma_{ij}(l) - \delta(\gamma_{ij}(l) - L_{ij}(l)) \tag{13}$$

for $(L_{ij}(l), \gamma_{ij}(l))$.

According to the above discussion, the tuning algorithm becomes as follows.

1) Set a positive number to parameter l_M, where $l_M - 1$ is the maximum number of misclassifications allowed for tuning α_{ij}, a value in $(0, 1)$ to δ in (12) and (13), and the same positive initial value (usually one) to α_{ij}.
2) For α_{ij} $(i = 1, \dots, n, j = 1, \dots)$, calculate $L_{ij}(l)$, $U_{ij}(l)$, $\beta_{ij}(l)$, and $\gamma_{ij}(l)$ for $l = 1, \dots, l_M$. Find the interval $(L_{ij}(l), \gamma_{ij}(l))$ or $(\beta_{ij}(l), U_{ij}(l))$ that realizes the maximum recognition rate of the training data, and change α_{ij} using (12) or (13).
3) Iterate step 2) until there is no improvement in the recognition rate.

Usually $l_M = 10$ is sufficient. According to our experiments [3], the value of δ did not affect the recognition rate of the test data significantly, but a small value of δ sometimes gave a better recognition rate of the test data. Thus, in the experiments in Section VII, we use 0.1. The detailed tuning algorithm is discussed in the Appendix.

We call the update of all α_{ij} $(i = 1, \ldots, n, j = 1, \ldots)$ one iteration of tuning, and if there is no improvement in the recognition rate for the two consecutive iterations, or the recognition rate of the training data reaches 100%, we stop tuning. Our tuning algorithm determines, for each fuzzy rule R_{ij}, the optimum tuning parameter α_{ij}, allowing the data that are correctly classified before tuning R_{ij} to become misclassified after tuning R_{ij} as long as the recognition rate of the training data is improved. To allow the data that are correctly classified before tuning some fuzzy rule to be misclassified after tuning that fuzzy rule is, so to speak, to prevent the tuning process from leading to convergence to a local minimum. But of course, since the tuning process is nonlinear, we cannot guarantee that this method always gives the optimal solution.

V. Threshold Optimization

When a sufficient number of the data belonging to the abnormal class are obtained, we can determine the threshold t_h so that the recognition rate is maximized. Letting $N(a)$ be the number of the correctly classified data with threshold a, then we give $N(a)$ by

$$N(a) = N_{nn}(a) + N_{aa}(a) \quad (14)$$

where $N_{nn}(a)$ is the number of the data that are correctly classified into the normal classes with $t_h = a$ and $N_{aa}(a)$ is the number of the data that are correctly classified into the abnormal class with $t_h = a$. By introducing the threshold the regions for the normal classes are bounded. Thus, the following relations hold:

$$N_{nn}(a) \le N_{nn}(\infty), \quad N_{aa}(a) \ge N_{aa}(\infty) \text{ for positive } a. \quad (15)$$

Therefore, there is an optimal a that maximizes $N(a) - N(\infty)$. Since

$$\begin{aligned} N(a) - N(\infty) &= N_{aa}(a) - N_{aa}(\infty) + N_{nn}(a) - N_{nn}(\infty) \\ &= \Delta N_{aa}(a) + \Delta N_{nn}(a) \end{aligned} \quad (16)$$

where $\Delta N_{aa}(a)$ (≥ 0) is the increase of the data correctly classified into the abnormal class by introducing $t_h = a$ and $\Delta N_{nn}(a)$ (≤ 0) is the decrease of the data correctly classified into the normal classes by introducing $t_h = a$.

To simplify the maximization of (16), we set the interval $[t_{h,\min}, t_{h,\max}]$, where $t_{h,\min} > 1$, and in the interval, we calculate (16) for $t_{h,\min} + k\Delta$, where $t_{h,\min} + k\Delta \le t_{h,\max}$ for $k = 0, 1, \ldots$, and we obtain $t_{h,\min} + k\Delta$ that maximizes (16).

VI. Online Training

The parameters of the fuzzy rule R_{ij} are the center $\mathbf{c}_{ij}$, the covariance matrix Q_{ij}, and the tuning parameter α_{ij}. The first two parameters are determined by the training data. Thus, if a sufficient number of the training data for that rule are not available, $\mathbf{c}_{ij}$ and Q_{ij} need to be recalculated when the associated data are obtained. The tuning parameter α_{ij} is determined so that the recognition rate of the training data is maximized. Thus, if the fuzzy rule R_{ij} is determined using a sufficient number of the training data, we only need to tune α_{ij} for that rule using the training data.

Now for the diagnosis problem we assume that at the initial development stage of the fuzzy classifier we have a sufficient number of the data belonging to the normal classes, but no data belonging to the abnormal class. Thus, the parameters $\mathbf{c}_{ij}$ and Q_{ij} of the fuzzy rules for the normal classes are calculated only once at the initial generation and are fixed thereafter. By contrast the parameters $\mathbf{c}_{ij}$ and Q_{ij} of the fuzzy rules for the abnormal class are recalculated or new fuzzy rules are generated whenever new data belonging to the abnormal class are obtained until there are a sufficient number of the data belonging to the abnormal class.

TABLE I
Processing After Classification

Classifier's classification	Operator's classification	Processing
the same normal class		none
different normal classes		tune α_{ij}
normal	abnormal	modify rules and tune α_{ij}
abnormal	normal	tune α_{ij}
abnormal	abnormal	modify rules and tune α_{ij}

The development of the fuzzy classifier is divided into two stages: the training stage with an insufficient number of the data belonging to the abnormal class and the tuning stage with a sufficient number of the data belonging to the abnormal class, as described next. Training stage: in this stage, there are no rules for the data belonging to the abnormal class or, even if they exist, their reliability is low. Therefore, we set the small threshold t_h to avoid misclassification of the data belonging to the abnormal class. But to ensure a sufficient recognition rate of the data belonging to the normal classes, t_h needs to be larger than one. After classification of data by the fuzzy classifier, the operator checks the result and feeds back the operator's decision to the classifier. Processing thereafter is divided into five cases, as shown in Table I. If the input datum is classified into the same normal class both by the classifier and the operator, no processing is done. But if the input datum is classified into the different normal classes by the classifier and the operator, all α_{ij} are tuned using the previously obtained data and the newly obtained datum. If the input datum is classified into one of the normal classes by the classifier, but classified into the abnormal class by the operator, or classified into the abnormal class both by the classifier and the operator, the rules for the abnormal class are generated or modified and then all α_{ij} are tuned using the previously obtained data and the newly obtained datum. If the input datum is classified into the abnormal class by the classifier, but it is classified into one of the normal classes by the operator, all α_{ij} are tuned using the previously obtained data and the newly obtained datum. The input datum is stored for tuning or rule generation or modification thereafter.

Tuning stage: in this stage, a sufficient number of the data belonging to the abnormal class are obtained, and hence, modification or generation of fuzzy rules for the abnormal class is not necessary. Then, if the classification of the classifier and that of the operator are different, α_{ij} are tuned and the threshold t_h is optimized.

VII. Performance Evaluation

We evaluated the performance of the fuzzy classifier with ellipsoidal regions using iris data [12], blood cell data [13], and thyroid data [14] and compared the performance with that of the multilayered neural network classifier and the fuzzy classifier with hyperbox regions [5] when a sufficient number of the data belonging to the abnormal class were available. By combining several classes into one, classification usually becomes difficult. Thus, in our study, we combined several classes into one class and considered it as the abnormal class. When we clustered classes, we only clustered the data belonging to the abnormal class. Then to test the feasibility of online training, we first generated a fuzzy classifier using all data

TABLE II
PERFORMANCE OF THE FUZZY CLASSIFIER WITH ELLIPSOIDAL REGIONS FOR IRIS DATA BY CHANGING σ_M (WITHOUT THRESHOLD)

σ_M	No. Clusters	Init.	Final	Iterations	Time (s)
0.5	1	2	2	2	2
0.4	2	1	2	1	2
0.3	2	1	2	1	2
0.2	3	2	2	1	2
0.1	7	2	2	1	2

belonging to the normal classes included in the training data. Then we trained the fuzzy classifier online, successively feeding the data belonging to the abnormal class selected from the top to the bottom of the training data file. At each online training step, we evaluated the recognition rate using the test data. The recognition rates in the following tables are all those of the test data.

Unless otherwise stated, for α_{ij} tuning, we set $\alpha_{ij} = 1$ as initial values, $\delta = 0.1$, and $l_M = 10$ and, for threshold tuning, we set $t_{h,\min} = 2$, $t_{h,\max} = 10$, and $\Delta = 0.2$. For evaluation of the fuzzy classifier with hyperbox regions, we used a 16-MIPS workstation. Except for that, we used a 60-MIPS mainframe computer, and the calculation times listed in the following tables are the CPU times. We trained the three-layered neural network by the backpropagation algorithm [1]. The performance of the fuzzy classifier with hyperbox regions was evaluated for the expansion parameters of 0.01, 0.1, 0.15, 0.2, 0.25, and 0.3, where the expansion parameter is the parameter for expanding the overlapping hyperbox regions.

A. Iris Data

The Fisher iris data [12] consist of 150 data with four input features and three classes. In our study, the training data set was composed of the first 25 data of each class, while the test data set was composed of the remaining 25 data of each class. We assumed that the first and the second classes were abnormal classes and combined them into one class. Table II shows the results of the fuzzy classifier with ellipsoidal regions when all training data were used. We changed σ_M from 0.5 to 0.1 and the threshold was not used. In the table, the number of clusters is for the data belonging to the abnormal class. The numbers in the "Init." and "Final" columns are the initial and final numbers of misclassified data, respectively. The number of iterations shows the number of tunings of α_{ij}. For $\sigma_M = 0.1$ to 0.4, the number of iterations was one. This was either because the recognition rate of the training data was 100% for the initial values or because it reached 100% by one iteration of tuning. The number of misclassified data was not changed even if we determined the optimal threshold using the test data. The minimum number of misclassifications was one with two clusters for the abnormal class and without tuning. Table III shows the performance of the test data using the neural network classifier and the fuzzy classifiers trained with all training data. The three-layered neural network classifier with two hidden units was trained ten times using different initial weights distributed in $[-0.1, 0.1]$; for each training, the number of epochs was 1000 and the learning rate was set to one with zero momentum. The minimum number of misclassified data was one, and the average number of misclassified data was 1.7. The minimum number of misclassifications was one using the fuzzy classifier with hyperbox regions, and the corresponding number of rules was 17. The results of the fuzzy classifier with ellipsoidal regions were summarized from Table II. The number of fuzzy rules was the number of clusters for the abnormal class plus one for the normal class. The performance, including the computation time, was comparable for the three classifiers.

TABLE III
PERFORMANCE FOR IRIS DATA

Classifier	No. Wrong	No. Rules	Time(s)
N.N.	1.7 (1 - 3)	2 units	2
Hyperbox	1 - 5	17 - 5	1
Ellipsoid	1 - 2	2 - 8	2

(): Minimum and maximum numbers of misclassified data

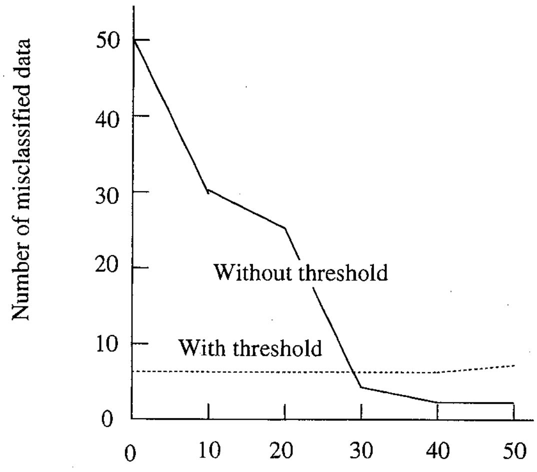

Fig. 4. Number of misclassified iris data in online training.

Fig. 4 shows the results of online training of the fuzzy classifier with ellipsoidal regions. The data belonging to the abnormal class were not clustered. The numbers of misclassified data were counted without the threshold and with the threshold of two. When there were no abnormal training data or in the early stage of training, the fuzzy classifier with the threshold worked better than that without the threshold. When no data belonging to the abnormal class were used for training, the data belonging to the abnormal class were classified into the normal class without the threshold. Thus, without the threshold, 50 test data were misclassified into the normal class. But with the threshold, only six data were misclassified; the numbers of misclassified data belonging to the abnormal class were almost constant for the increase of the training data belonging to the abnormal class. Since the first 25 data were the first class in the original iris data, the recognition rates of the classifier without the threshold were poor when the numbers of the training data belonging to the abnormal class were ten and 20.

B. Blood Cell Data

The blood cell data consist of 3097 training data and 3100 test data. The blood cell classification involves classifying optically screened white blood cells into five normal and seven abnormal classes, using 13 features. This is a very difficult problem; class boundaries for some classes are ambiguous because the classes are defined according to the growth stages of blood white cells. In this study, we combined seven abnormal classes into one abnormal class. Thus, the classes consisted of five normal classes and one abnormal class.

Table IV shows the results of the fuzzy classifier with ellipsoidal regions when all training data were used. We changed σ_M from one to 0.3. In the table, "∞" and "Optimal" columns denote the recognition rates, after tuning of α_{ij}, without the threshold and with the optimal threshold determined by the test data, respectively. When the optimal

TABLE IV
PERFORMANCE OF THE FUZZY CLASSIFIER WITH ELLIPSOIDAL REGIONS FOR BLOOD CELL DATA BY CHANGING σ_M (WITHOUT THRESHOLD OR WITH OPTIMAL THRESHOLD)

σ_M	No. Clusters	Init.	Final		Iterations	Time (s)
			∞	Optimal		
1.0	1	91.87	94.52	95.06 (3.0)	3	19
0.9	3	93.71	95.03	95.26 (3.0)	3	23
0.8	3	93.71	95.03	95.26 (3.0)	3	23
0.7	5	94.58	95.16	95.39 (3.0)	3	29
0.6	7	95.16	94.58	94.87 (3.2)	3	30
0.5	17	95.06	95.68	95.77 (7.6)	3	51
0.4	28	95.16	95.06	95.13 (7.6)	3	69
0.3	59	93.19	94.71	94.91 (7.6)	3	114

(): threshold t_h

TABLE V
PERFORMANCE FOR BLOOD CELL DATA

Classifier	Rate	No. Rules	Time (s)
N.N.	94.24*	30 units	95 min.
	95.68 - 93.42		
Hyperbox	93.65 - 92.74	88 -61	2
Ellipsoid	95.68 - 94.52	22 - 6	51 - 19

*: Average recognition rate

threshold was used, the recognition rate was slightly improved. When the threshold was determined using the training data, the recognition rates were also improved for $\sigma_M = 1.0$ to 0.6. For example, with $\sigma_M = 1.0$ the recognition rate was 94.61% for $t_h = 7.6$.

Table V shows the results by the neural network classifier and the fuzzy classifiers trained with all training data. The neural network classifier with 30 hidden units was trained five times with different initial weights; the number of epochs was 10 000, which required 95 min of CPU time. The recognition rate of the fuzzy classifier with hyperbox regions was 1–2% lower, but the fuzzy rule extraction was extremely fast. The results of the fuzzy classifier with ellipsoidal regions were summarized from Table IV for $\sigma_M = 1.0$ to 0.5. The number of fuzzy rules was the number of clusters for the abnormal class plus five for the normal classes. The maximum recognition rate of the fuzzy classifier with ellipsoidal regions was the same as that of the neural network classifier, and the recognition rate with one abnormal cluster exceeded the average performance of the neural network classifier. Although the training time was longer than that of the fuzzy classifier with hyperbox regions, it was much shorter than that of the neural network classifier. Thus, the training time does not hinder online training.

Fig. 5 shows the results of online training of the fuzzy classifier with ellipsoidal regions when data belonging to the abnormal class were not clustered. The recognition rates were calculated without the threshold and with the threshold of two. The fuzzy classifier with the threshold worked better than that without the threshold when there were no data belonging to the abnormal class or the data belonging to the abnormal class were not sufficient. When no data belonging to the abnormal class were used for training, with the threshold of two, the recognition rate of 90.58% was achieved. Fig. 6 shows the distributions of the normal and abnormal test data against the square of the weighted distance from the cluster center when no data belonging to the abnormal class were used for training. The data belonging to the normal classes whose square of the weighted distance were larger than two occupied 7.3% of the total data belonging to the normal classes, while the data belonging to the abnormal class whose square of the weighted distance were larger than two occupied

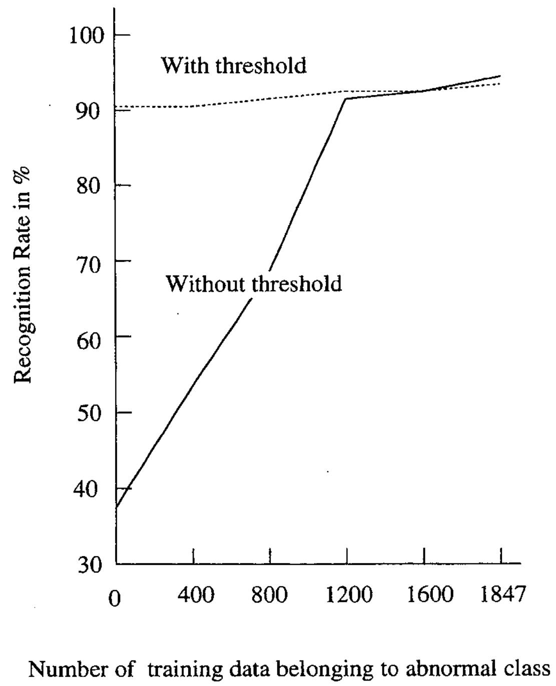

Fig. 5. Recognition rate of the blood cell data for online training.

89.8%. Namely, since the data sets belonging to normal and abnormal classes were distributed in the quasi-Gaussian distributions, the fuzzy classifier with the threshold worked.

C. Thyroid Data

The thyroid data classify input data consisting of 21 features into three classes. The training data and the test data consist of 3772 and 3428 data, respectively. The characteristics of the data are that the input features include 15 discrete features and more than 92% of the data belong to one class. We assumed the first two classes were abnormal and combined them into one abnormal class.

Table VI shows the results of the fuzzy classifier with ellipsoidal regions when all training data were used. We changed σ_M from 1.9 to 1.2. The recognition rates were calculated without the threshold. The initial recognition rates were extremely low. This meant that the covariance matrices did not approximate the distributions of the training data; namely, their distributions were not Gaussian.

Table VII shows the results by the neural network classifier and the fuzzy classifiers trained with all training data. The neural network classifier with three hidden units was trained five times with different initial weights; the number of epochs was 10 000, which required 60 min of CPU time. When six and nine hidden units were used, the average recognition rates were 97.72% and 97.87%, respectively, and there was not much difference for the change of the number of hidden units. The fuzzy classifier with hyperbox regions performed best. The minimum recognition rate was the maximum recognition rate of the neural network classifier, and the training time was negligible. The results of the fuzzy classifier with ellipsoidal regions were summarized from Table VI for $\sigma_M = 1.9$ to 1.3. The number of fuzzy rules was the number of clusters for the abnormal class plus one for the normal class. The recognition rate of the fuzzy classifier with the ellipsoidal regions was the worst. Thus, for the thyroid data, we cannot use the fuzzy classifier with the ellipsoidal regions. Fig. 7 shows the distributions of the normal and abnormal test data against the square of the weighted distance from the cluster center when only the data belonging to the normal classes were used

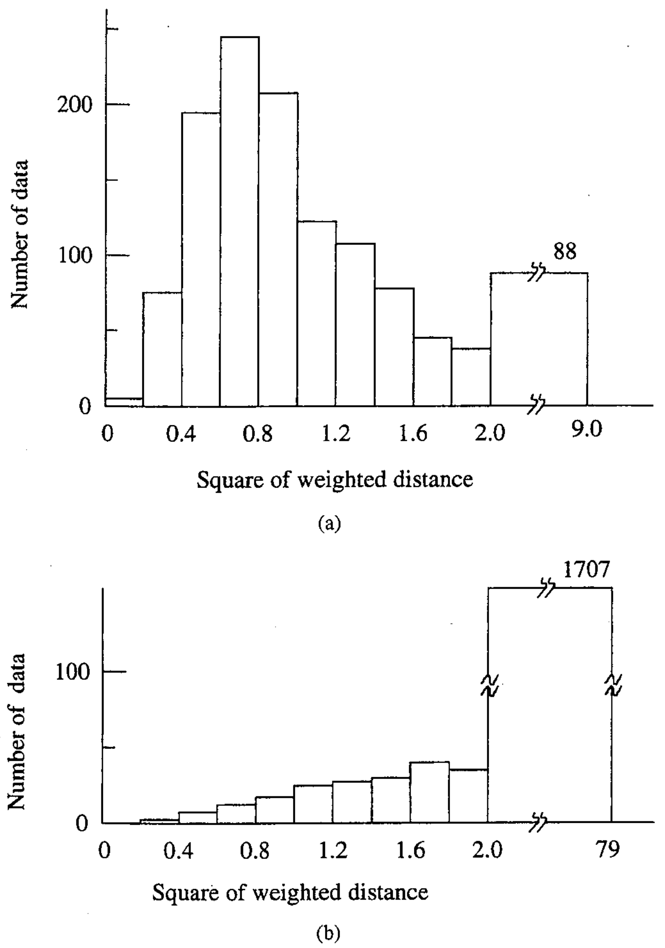

Fig. 6. Distributions of blood cell test data. (a) Distribution of normal class data. (b) Distribution of abnormal class data.

TABLE VI
PERFORMANCE OF THE FUZZY CLASSIFIER WITH ELLIPSOIDAL REGIONS FOR THYROID DATA BY CHANGING σ_M (WITHOUT THRESHOLD $l_M = 50$)

σ_M	No. Clusters	Init.	Final	Iterations	Time (s)
1.9	1	10.39	93.41	4	13
1.8	2	9.16	93.41	4	17
1.7	4	8.87	93.41	4	27
1.6	5	8.90	93.49	4	31
1.5	7	9.25	93.49	5	54
1.4	10	9.31	93.43	4	69
1.3	12	9.63	93.67	4	86
1.2	15	9.63	93.58	4	132

for training. The data belonging to the normal classes whose square of the weighted distance was larger than two occupied 14.7% of the total data belonging to the normal classes, which was twice as large as that of the blood cell data, and the data belonging to the abnormal class whose square of the weighted distance was larger than two occupied 40.9%, which was half as small as that of the blood cell data. Thus, we could not expect a high recognition rate using the threshold.

To investigate the effect of discrete input variables to classification, we used only the six continuous input variables of the thyroid data, i.e., the first and the seventeenth to twenty-first input variables. Table VIII shows the results of the fuzzy classifier with ellipsoidal regions. We changed σ_M from 0.7 to 0.3. The recognition rates were calculated without the threshold. The initial recognition rates were low but better than those when all the input variables were used (see Table VI). And the final recognition rates were better than when all the input variables were used. Thus, the 15 discrete input variables did not contribute to improving the recognition rate.

Table IX shows the results by the neural network classifier and the fuzzy classifiers. The neural network classifier with three hidden units was trained ten times with different initial weights; the number of epochs was 10 000. When six and nine hidden units were used,

TABLE VII
PERFORMANCE FOR THYROID DATA

Classifier	Rate	No. Rules	Time (s)
N.N.	97.65* 97.87 - 97.26	3 units	60 min.
Hyperbox	98.19 - 97.87	21 - 25	2
Ellipsoid	93.67 - 93.41	13 - 2	13 - 86

*: Average recognition rates

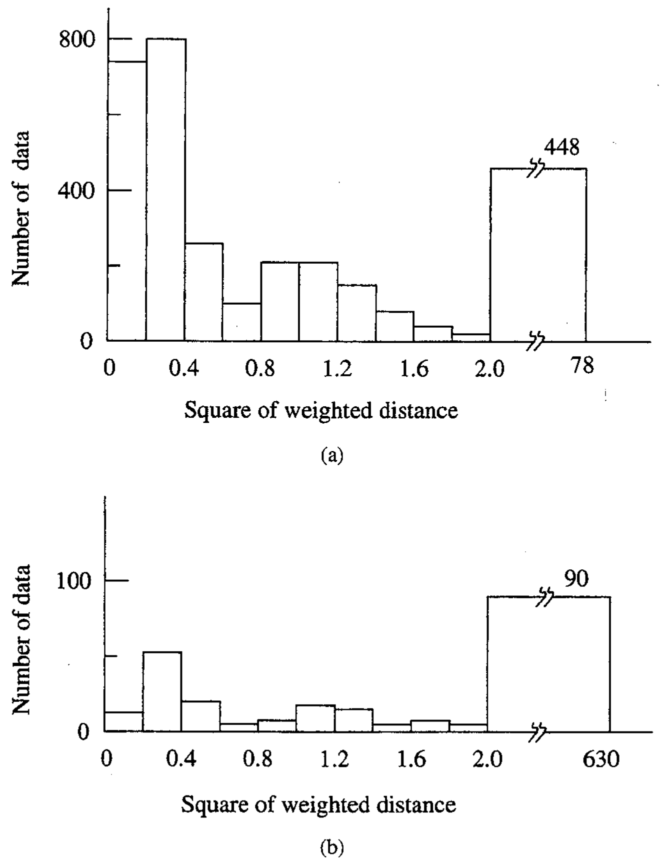

Fig. 7. Distribution of thyroid test data. (a) Distribution of normal class data. (b) Distribution of abnormal class data.

TABLE VIII
PERFORMANCE OF THE FUZZY CLASSIFIER WITH ELLIPSOIDAL REGIONS FOR THYROID DATA WITH SIX INPUT VARIABLES BY CHANGING σ_M (WITHOUT THRESHOLD)

σ_M	No. Clusters	Init.	Final	Iterations	Time (s)
0.7	1	53.03	95.13	8	6
0.6	2	50.79	95.22	8	7
0.5	3	49.42	95.10	8	9
0.4	9	45.62	95.16	8	17
0.3	14	45.01	95.04	7	27

TABLE IX
PERFORMANCE FOR THYROID DATA WITH SIX INPUT VARIABLES

Classifier	Rate	No. Rules	Time (s)
N.N.	96.72*	3 units	21 min.
	96.82 - 96.65		
Hyperbox	97.02 - 96.91	40 - 36	2
Ellipsoid	95.22 - 95.04	13 - 2	6 - 27

*: Average recognition rates

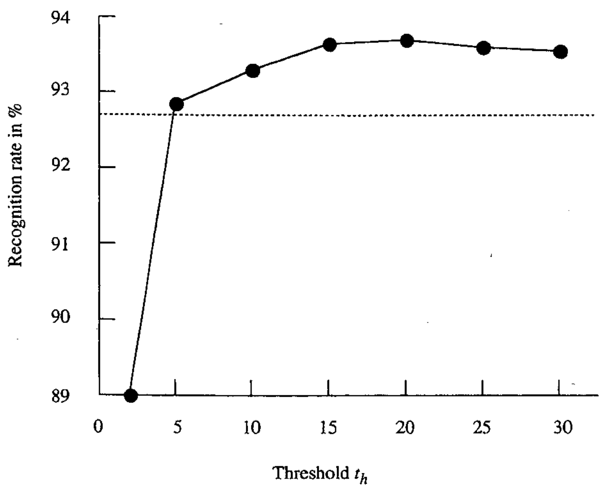

Fig. 8. Classification of thyroid data with six input variables by the fuzzy classifier with ellipsoidal regions trained by only the normal class data.

the average recognition rates were 96.82% and 96.83%, respectively. Thus, there was not much difference in the change of the number of hidden units. The fuzzy classifier with hyperbox regions performed best. The number of fuzzy rules was the number of clusters for the abnormal class plus one for the normal class. The recognition rate of the fuzzy classifier with the ellipsoidal regions was the worst. But the performance gap was decreased. Fig. 8 shows the recognition rate of the fuzzy classifier with ellipsoidal regions with the threshold for the thyroid test data, when only the data belonging to the normal class were used. Since the test data belonging to the normal class occupied 92.71% of the total test data, the recognition rate with the threshold lower than this value was meaningless. For $t_h = 2$, the recognition rate was 88.96%, but when t_h was larger than or equal to five, the recognition rate exceeded 92.71%. Thus, if we set a large value to the threshold, we could obtain an improved recognition rate by the fuzzy classifier with ellipsoidal regions.

VIII. DISCUSSION

The fuzzy classifier proposed in this paper performed well even when there were no training data belonging to the abnormal class, if the training data did not include discrete input variables. This is one of the advantages of the fuzzy classifier with ellipsoidal regions over other classifiers since, without training data belonging to the abnormal class, training of the neural network classifier or fuzzy classifier with hyperbox regions is impossible. The training time was short enough for implementing the online system. At the training stage, we need a sufficient number of training data, but after we generate fuzzy rules, i.e., at the tuning stage we can select training data for tuning α_{ij} and threshold optimization. Namely, to tune α_{ij} we need only the data that exist at the boundary of classes, i.e., away from the centers of the clusters. In addition, to optimize the threshold, we only need the data around the threshold. Thus, by deleting data that are near the centers of clusters from the training data, we can reduce the number of training data.

IX. CONCLUSION

In this paper, we extended a fuzzy classifier with ellipsoidal regions to make it applicable to diagnosis problems, in which the data belonging to the abnormal class are difficult to obtain. Assuming that there were no data belonging to the abnormal class, we first trained the fuzzy classifier with only the data belonging to the normal classes. We then introduced the threshold of the weighted distance from the center of the cluster for the data belonging to the normal classes. Based on agreement or disagreement of the classification by the classifier and the operator, the classifier was adapted until there were a sufficient number of the data belonging to the abnormal class. We evaluated our method using the Fisher iris data, blood cell data, and thyroid data and obtained good performance when the training data did not contain discrete input variables.

APPENDIX

In Appendix A, we calculate the upper bound and the lower bound of α_{ij} that allow the $l-1$ (≥ 0) data that are correctly classified to become misclassified. And in Appendix B, we check how many data that are misclassified are correctly classified if α_{ij} is changed within the bounds calculated in Appendix A. Then in Appendix C, α_{ij} is determined so that the recognition rate of the training data is maximized.

A. Upper and Lower Bounds of α_{ij}

We calculate the upper bound $U_{ij}(l)$ and the lower bound $L_{ij}(l)$ of α_{ij} allowing the $l-1$ (≥ 0) data that are correctly classified to be misclassified. We divide a set of input data into X and Y, where X consists of the data correctly classified using the set of fuzzy rules $\{R_{ij}\}$ and Y consists of the misclassified data. Then, we choose $\mathbf{x}$ ($\in X$) that belongs to class i, and that satisfies

$$h_{ij}(\mathbf{x}) \leq \min_{k \neq j} h_{ik}(\mathbf{x}). \tag{17}$$

If (17) does not hold, $\mathbf{x}$ remains to be correctly classified even if we change α_{ij}. If $\mathbf{x}$ further satisfies

$$h_{ij}^2(\mathbf{x}) = \frac{d_{ij}^2(\mathbf{x})}{\alpha_{ij}} < \min_{o \neq i, p=1,\ldots} h_{op}^2(\mathbf{x}) < \min_{k \neq j} h_{ik}^2(\mathbf{x}) \tag{18}$$

there is a lower bound $L_{ij}(\mathbf{x})$ to keep $\mathbf{x}$ correctly classified

$$L_{ij}(\mathbf{x}) = \frac{d_{ij}^2(\mathbf{x})}{\min_{o \neq i, p=1,\ldots} h_{op}^2(\mathbf{x})} < \alpha_{ij}. \tag{19}$$

If (18) is not satisfied, namely

$$h_{ij}^2(\mathbf{x}) = \frac{d_{ij}^2(\mathbf{x})}{\alpha_{ij}} < \min_{k \neq j} h_{ik}^2(\mathbf{x}) < \min_{o \neq i, p=1,\ldots} h_{op}^2(\mathbf{x}) \tag{20}$$

α_{ij} can be decreased without making $\mathbf{x}$ become misclassified.

Now the lower bound $L_{ij}(1)$, which is defined as the lower bound that does not make any correctly classified data become misclassified, is

$$L_{ij}(1) = \max_{\mathbf{x} \in X} L_{ij}(\mathbf{x}). \tag{21}$$

To clarify the discussion, we assume that $L_{ij}(\mathbf{x})$ is different for different $\mathbf{x}$. Then (21) is satisfied by one $\mathbf{x}$. Similarly, $L_{ij}(2)$, which is defined as the lower bound that allows one correctly classified datum to be misclassified, is the second maximum among $L_{ij}(\mathbf{x})$ and is given by

$$L_{ij}(2) = \max_{\mathbf{x} \in X, L_{ij}(\mathbf{x}) \neq L_{ij}(1)} L_{ij}(\mathbf{x}). \tag{22}$$

In general

$$L_{ij}(l) = \max_{\mathbf{x}\in X, L_{ij}(\mathbf{x})\neq L_{ij}(1),\ldots,L_{ij}(l-1)} L_{ij}(\mathbf{x}). \tag{23}$$

In the similar manner that we determined the lower bound $L_{ij}(l)$, we can determine the upper bound $U_{ij}(l)$. We choose $\mathbf{x}$ ($\in X$), which belongs to class o ($\neq i$). Let cluster op have the minimum tuned distance $h_{op}(\mathbf{x})$

$$h_{op} = \min_q h_{op}(\mathbf{x}). \tag{24}$$

Since the tuned distance $h_{ij}(\mathbf{x})$ is larger than $h_{op}(\mathbf{x})$, the upper bound $U_{ij}(\mathbf{x})$ of α_{ij}, in which $\mathbf{x}$ remains correctly classified, is

$$U_{ij}(\mathbf{x}) = \frac{d_{ij}^2(\mathbf{x})}{\min_q h_{oq}^2(\mathbf{x})}. \tag{25}$$

Now the upper bound $U_{ij}(1)$, which is defined as the upper bound that does not make any correctly classified data be misclassified, is

$$U_{ij}(1) = \min_{\mathbf{x}\in X} U_{ij}(\mathbf{x}). \tag{26}$$

Here we also assume that $U_{ij}(\mathbf{x})$ is different for different $\mathbf{x}$. Then (26) is satisfied by one $\mathbf{x}$. Similarly, $U_{ij}(2)$, which is defined as the upper bound that allows one correctly classified datum to be misclassified, is the second minimum among $U_{ij}(\mathbf{x})$ and is given by

$$U_{ij}(2) = \min_{\mathbf{x}\in X, U_{ij}(\mathbf{x})\neq U_{ij}(1)} U_{ij}(\mathbf{x}). \tag{27}$$

In general

$$U_{ij}(l) = \min_{\mathbf{x}\in X, U_{ij}(\mathbf{x})\neq U_{ij}(1),\ldots,U_{ij}(l-1)} U_{ij}(\mathbf{x}). \tag{28}$$

Thus, α_{ij} is bounded by

$$\cdots < L_{ij}(l) < L_{ij}(l-1) < \cdots < L_{ij}(1) < \alpha_{ij} < U_{ij}(1) < \cdots < U_{ij}(l-1) < U_{ij}(l) < \cdots \tag{29}$$

If we change α_{ij} in the range of $(L_{ij}(1), U_{ij}(1))$, the correctly classified data remain to be correctly classified, where (a, b) denotes the open interval. And if we change α_{ij} in the range of $[U_{ij}(l-1), U_{ij}(l))$, or $(L_{ij}(l), L_{ij}(l-1)]$, the $l-1$ correctly classified data are misclassified, where $[a, b]$ denotes the closed interval.

B. Resolution of Misclassification by Changing α_{ij}

For $\mathbf{x}$ ($\in Y$) that is misclassified into class i or that belongs to class i but is misclassified into class o ($\neq i$), we check whether it can be correctly classified by changing α_{ij}. First we consider increasing α_{ij}. Let $\mathbf{x}$, which belongs to class i, be misclassified into class o. This datum can be correctly classified if

$$\alpha_{ij} > V_{ij}(\mathbf{x}) = \frac{d_{ij}^2(\mathbf{x})}{\min_p h_{op}^2(\mathbf{x})} \tag{30}$$

irrespective of the values of $h_{ik}(\mathbf{x})$ ($k \neq i$), where $V_{ij}(\mathbf{x})$ is the lower bound of α_{ij} that makes the misclassified $\mathbf{x}$ correctly classified.

Let $\mathrm{Inc}(l)$ denote the number of the misclassified data that are correctly classified if we set the value of α_{ij} in $[U_{ij}(l-1), U_{ij}(l))$. We increase $\mathrm{Inc}(l)$ by one if $V_{ij}(\mathbf{x})$ is included in $(\alpha_{ij}, U_{ij}(l))$, and we define

$$\beta_{ij} = \max_{V_{ij(\mathbf{x})} < U_{ij}(l)} V_{ij}(\mathbf{x}). \tag{31}$$

If α_{ij} is set to be larger than $\max(\beta_{ij}(l), U_{ij}(l-1))$, $\mathrm{Inc}(l)$ data are correctly classified although the $l-1$ correctly classified data are misclassified.

Let $\mathbf{x}$, which belongs to class o, be misclassified into class i. Then similar to the above discussions, we check whether $\mathbf{x}$ can be correctly classified by decreasing α_{ij}. First, the minimum tuned distance for class o should be the second minimum among n classes, namely, q in the following needs to be o:

$$\min_k h_{ik}(\mathbf{x}) < \min_{q\neq i, r=1,\ldots} h_{qr}(\mathbf{x}). \tag{32}$$

Second, $h_{ij}(\mathbf{x})$ needs to be the minimum in class i, and the second minimum in class i is larger than the minimum tuned distance in class o

$$h_{ij}(\mathbf{x}) < \min_p h_{op}(\mathbf{x}) < \min_{k\neq j} h_{ik}(\mathbf{x}). \tag{33}$$

Then, the datum can be correctly classified if

$$\alpha_{ij} < K_{ij}(\mathbf{x}) = \frac{d_{ij}^2(\mathbf{x})}{\min_p h_{op}^2(\mathbf{x})} \tag{34}$$

where $K_{ij}(\mathbf{x})$ is the upper bound of α_{ij} that makes misclassified $\mathbf{x}$ become correctly classified.

Let $\mathrm{Dec}(l)$ denote the number of the misclassified data that are correctly classified if we set the value of α_{ij} in $(L_{ij}(l), L_{ij}(l-1)]$. We increase $\mathrm{Dec}(l)$ by one if $K_{ij}(\mathbf{x})$ is included in $(L_{ij}(l), \alpha_{ij})$. We define

$$\gamma_{ij}(l) = \min_{K_{ij}(\mathbf{x}) > L_{ij}(l)} K_{ij}(\mathbf{x}). \tag{35}$$

If α_{ij} is set to be smaller than $\min(\gamma_{ij}(l), L_{ij}(l-1))$, $\mathrm{Dec}(l)$ data are correctly classified although the $l-1$ correctly classified data are misclassified.

C. Modification of α_{ij}

For $\mathrm{Inc}(l), l = 1, \ldots, l_M$, where l_M is a positive integer, we find l that satisfies

$$\max_l(\mathrm{Inc}(l) - l + 1). \tag{36}$$

Similarly, for $\mathrm{Dec}(l), l = 1, \ldots, l_M$, we find l that satisfies

$$\max_l(\mathrm{Dec}(l) - l + 1). \tag{37}$$

If there are plural l's that satisfy (36) or (37), we chose the smallest l. First we consider the case in which (36) is larger than or equal to (37). If we increase α_{ij} so that it is larger than $\beta_{ij}(l)$ in $(\alpha_{ij}, U_{ij}(l))$, the net increase of the correctly classified data is $\mathrm{Inc}(l) - l + 1$. Thus, we set α_{ij} in $[\beta_{ij}(l), U_{ij}(l))$ as follows:

$$\alpha_{ij} = \beta_{ij}(l) + \delta(U_{ij}(l) - \beta_{ij}(l)) \tag{38}$$

where δ satisfies $0 < \delta < 1$. Here, $\beta_{ij}(l) \geq U_{ij}(l-1)$ holds, otherwise l cannot satisfy (36).

Likewise, if (36) is smaller than (37), we decrease α_{ij} so that it is smaller than $\gamma_{ij}(l)$ in $(L_{ij}(l), \gamma_{ij}(l)]$ as follows:

$$\alpha_{ij} = \gamma_{ij}(l) - \delta(\gamma_{ij}(l) - L_{ij}(l)). \tag{39}$$

Equations (38) and (39) are the same as (12) and (13), respectively. The parameter δ is used to control the recognition rate of the test data (the recognition rate of the training data is the same irrespective of the value of δ).

Acknowledgment

The authors are grateful to Prof. N. Matsuda of the Kawasaki Medical School for providing the blood cell data and to P. Murphy and D. Aha of the University of California at Irvine for organizing the databases, including the thyroid data (ics.uci.edu: pub/machine-learning-databases). Thanks are also due to the anonymous reviewers for their constructive comments.

REFERENCES

[1] S. Abe, *Neural Networks and Fuzzy Systems: Theory and Applications.* Boston, MA: Kluwer, 1996.

[2] M. T. Musavi, W. Ahmed, K. H. Chan, K. B. Faris, and D. M. Hummels, "On the training of radial basis function classifiers," *Neural Networks*, vol. 5, no. 4, pp. 595–603, 1992.

[3] S. Abe and R. Thawonmas, "A fuzzy classifier with ellipsoidal regions," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 358–368, Mar. 1997.

[4] P. K. Simpson, "Fuzzy min-max neural networks—Part 1: Classification," *IEEE Trans. Neural Networks*, vol. 3, pp. 776–786, Sept. 1992.

[5] S. Abe and M.-S. Lan, "A method for fuzzy rules extraction directly from numerical data and its application to pattern classification," *IEEE Trans. Fuzzy Syst.*, vol. 3, pp. 18–28, Jan. 1995.

[6] F. Uebele, S. Abe, and M.-S. Lan, "A neural network-based fuzzy classifier," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, pp. 353–361, Mar. 1995.

[7] A. Kawato and S. Hayashi, "A method of ball-bearing diagnosis by neural network using the normal mode data," *Trans. Inst. Elect. Eng. Japan*, vol. 115-C, no. 11, pp. 1362–1368, 1995 (in Japanese).

[8] M. P. Windham, "Cluster validity for the fuzzy *c*-means clustering algorithm," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-4, pp. 357–363, Apr. 1982.

[9] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 98–110, Feb. 1993.

[10] J. C. Bezdek and R. J. Hathaway, "Convergence theory for fuzzy *c*-means: Counterexamples and repairs," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-17, pp. 873–877, Sept. 1987.

[11] T. Kohonen, *Self-Organization and Associative Memory*, 2nd ed. Berlin, Germany: Springer-Verlag, 1987.

[12] R. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, part II, pp. 179–188, 1936.

[13] A. Hashizume, J. Motoike, and R. Yabe, "Fully automated blood cell differential system and its application," in *Proc. IUPAC 3rd Int. Congress Automat. New Technol. Clinical Lab.*, Sept. 1988, pp. 297–302.

[14] S. M. Weiss and I. Kapouleas, "An empirical comparison of pattern recognition, neural nets, and machine learning classification methods," in *Proc. Int. Joint Conf. Neural Networks*, pp. 781–787, 1989.

Neural-Network-Based Fuzzy Model and Its Application to Transient Stability Prediction in Power Systems

Mu-Chun Su, Chih-Wen Liu, and Shuenn-Shing Tsay

***Abstract*—This paper presents a general approach to deriving a new type of neural-network-based fuzzy model for a complex system from numerical and/or linguistic information. To efficiently identify the structure and the parameters of the new fuzzy model, we first partition the output space instead of the input space. As a result, the input space itself induces corresponding partitions within each of which inputs would have similar outputs. Then we use a set of hyperrectangles to fit the partitions of the input space. Consequently, the premise of an implication in the new type of fuzzy rule is represented by a hyperrectangle and the consequence is represented by a fuzzy singleton. A novel two-layer fuzzy hyperrectangular composite neural network (FHRCNN) can be shown to be computationally equivalent to such a special fuzzy model. The process of presenting input data to each hidden node in a FHRCNN is equivalent to firing a fuzzy rule. An efficient learning algorithm was developed to adjust the weights of an FHRCNN. Finally, we apply FHRCNN's to provide real-time transient stability prediction for use with high-speed control in power systems. From simulation tests on the IEEE 39-bus system, it reveals that the proposed novel FHRCNN can yield a much better performance than that of conventional multilayer perceptrons (MLP's) in terms of computational burden and classification rate.**

***Index Terms*—Fuzzy systems, neural networks, transient stability prediction.**

I. INTRODUCTION

Neural networks and fuzzy systems have attracted the growing interest of researchers in various disciplines of engineering and science. Their applications range widely from consumer products to decision analysis. Basically, a neural network is a massively parallel-distributed processor. Among the many appealing properties of a neural network, the property that is of primary significance is the ability of the neural network to inductively learn concepts from given numerical data. A neural network improves its performance by adjusting its synaptic weights. Feedforward neural networks [e.g., multilayer perceptrons (MLP's)] have been proven to be able to approximate any real continuous function on a compact set to arbitrary accuracy [1]–[3]. Therefore, a feedforward neural network is an efficient tool for system modeling and identification, however, there are three major disadvantages in a feedforward neural network. The first one is that there is no systematic way to set up the topology of a neural network. The second one is that it usually takes a lot of time to train a neural network. The third and the most apparent one is that a trained neural network is unable to explain its response (i.e., the inference process cannot be stated explicitly). Therefore, even if we can finally model a complex system by a trained neural network, the knowledge encoded in the values of the parameters of the trained neural network is not physically meaningful to humans when they depend on appropriate and understandable information to make decisions. Accordingly, how to acquire a relevant and meaningful system description from observed data or experience is very much demanded.

Manuscipt received January 6, 1997; revised December 18, 1997 and June 2, 1998.

M.-C. Su is with the Department of Electrical Engineering, Tamkang University, Tamsui, 25137 Taiwan, R.O.C. (e-mail: muchun@ee.tku.edu.tw).

C.-W. Liu and S.-S. Tsay are with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C.

Publisher Item Identifier S 1094-6977(99)00095-4.