Feature selection by analyzing class regions approximated by ellipsoids

Abe, Shigeo

Thawonmas, Ruck

Kobayashi, Yoshiki

(Citation)
IEEE transactions on systems, man, and cybernetics. Part C, Applications and reviews,28(2):282-287

(Issue Date)
1998-05

(Resource Type)
journal article

(Version)
Version of Record

(URL)
https://hdl.handle.net/20.500.14094/90000220

Feature Selection by Analyzing Class Regions Approximated by Ellipsoids

Shigeo Abe, Ruck Thawonmas, and Yoshiki Kobayashi

Abstract—**In our previous work, we have developed a method for selecting features based on the analysis of class regions approximated by hyperboxes. In this paper, we select features analyzing class regions approximated by ellipsoids. First, for a given set of features, each class region is approximated by an ellipsoid with the center and the covariance matrix calculated by the data belonging to the class. Then, similar to our previous work, the exception ratio is defined to represent the degree of overlaps in the class regions approximated by ellipsoids. From the given set of features, we temporally delete each feature, one at a time, and calculate the exception ratio. Then, the feature whose associated exception ratio is the minimum is deleted permanently. We iterate this procedure while the exception ratio or its increase is within a specified value by feature deletion. The simulation results show that our current method is better than the principal component analysis (PCA) and performs better than our previous method, especially when the distributions of class data are not parallel to the feature axes.**

Index Terms—**Feature selection, membership function, pattern classification, principal component analysis.**

I. Introduction

In developing a pattern classification system for a given problem, we need to realize a high recognition rate for the unknown data, i.e., high generalization ability. The type of classifier used influences the generalization ability, but the most influencing factor is the set of features used.

There are two approaches to determining the set of features: feature extraction [1]–[4] and feature selection [5]–[11]. Feature extraction, linearly or nonlinearly, transforms the original set of features into a reduced one. Principal component analysis (PCA) [1], [2] is a well-known feature extraction method in which input axes are rotated around the coordinate origin of the original features in the directions of the eigenvectors of the feature covariance matrix and some of the transformed features are selected from the most significant axes in order. Discriminant analysis [1] finds the set of transformed features that gives the greatest class separation. In [3] and [4], class regions were analyzed to retain useful features and to eliminate redundant features.

Feature selection selects relevant features from the original features. In [5], various measures, such as the Bhattacharyya probabilistic distance, were discussed to select the set of features that maximizes class separability. In [6], some fuzzy parameters to measure class separability were used to select features. In [7], features were selected based on the mutual information criterion.

In [8], we proposed a feature selection method based on the analysis of class regions that are generated by a fuzzy classifier with hyperbox regions [9], [12]. The degree of overlaps in the class regions is defined as the exception ratio and is used as a measure for feature evaluation. Given a set of remaining features, the proposed algorithm eliminates the next feature, the elimination of which minimizes the exception ratio. The simulations for four benchmark data (iris data, numeral data, thyroid data, and blood-cell data) showed that the proposed method could successfully delete irrelevant features. We evaluated the recognition rates of the test data by the fuzzy classifier and the multilayered neural network classifier. Employing the reduced features obtained by our method resulted in an inferior recognition rate in comparison to the same number of features by the PCA, only when the recognition rate was evaluated by the neural network classifier for the blood-cell data. This inferior recognition rate was considered to be caused by the analysis of overlapping regions of hyperboxes whose surfaces were parallel to features [13]. Thus, when the distributions of the class data are not parallel to the features, analysis of the hyperboxes may include errors.

In this paper, to overcome this difficulty, we approximate class regions by ellipsoids, with the centers and the covariance matrixes calculated by the data belonging to the classes. Then, similar to [8], the exception ratio is defined to represent the degree of overlaps in the class regions approximated by ellipsoids. From the given set of features, we temporally delete each feature, one at a time, and calculate the exception ratio. Then, the feature whose associated exception ratio is the minimum is deleted permanently. We iterate this procedure while the exception ratio or its increase is within a specified value by feature deletion.

In Section II, we first approximate the class regions by ellipsoids. Then we define the exception ratio and propose the feature elimination algorithm based on the exception ratio. In Section III, we compare our method with that discussed in [8] and the PCA by the fuzzy classifier with hyperbox regions, the fuzzy classifier with ellipsoidal regions [13], and the multilayered neural network classifier for the same four benchmark data sets used in [8].

II. Feature Selection

A. Approximation of Class Regions by Ellipsoids

We represent m features for classifying n classes by an m-dimensional input vector $\boldsymbol{x}$. We assume that we have the data belonging to each class for feature selection. Then, we approximate the region for class i by one ellipsoid with the center $\boldsymbol{c}_i = (c_{i1}, \cdots, c_{im})^t$

$$c_{ik} = \frac{1}{N_i} \sum_{\boldsymbol{x} \in \text{ class } i} x_k \tag{1}$$

where t denotes the transpose of a matrix and N_i is the number of data belonging to class i and the $m \times m$ sample covariance matrix

$$Q_i = \frac{1}{N_i} \sum_{\boldsymbol{x} \in \text{ class } i} (\boldsymbol{x} - \boldsymbol{c}_i)(\boldsymbol{x} - \boldsymbol{c}_i)^t. \tag{2}$$

If the sample covariance matrix Q_i is singular, we set all of the off-diagonal elements of Q_i to zero so that $Q_i$becomes regular.

Now we can calculate the weighted distance of the input vector $\boldsymbol{x}$ from the class center $\boldsymbol{c}_i$, $d_i(\boldsymbol{x})$ by

$$d_i^2(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{c}_i)^t Q_i^{-1} (\boldsymbol{x} - \boldsymbol{c}_i) \tag{3}$$

where the superscript -1 denotes a matrix inversion.

Assuming that Q_i is regular, Q_i is a positive definite matrix. Then the mean-squared weighted distance is m (see the Appendix)

$$\frac{1}{N_i} \sum_{\boldsymbol{x} \in \text{ class } i} d_i^2(\boldsymbol{x}) = m. \tag{4}$$

Manuscript received June 7, 1996; revised October 22, 1997.

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

R. Thawonmas is with the Multimedia Devices Laboratory, Department of Computer Hardware, University of Aizu, Aizu, Wakamatsu 965-80, Japan.

Y. Kobayashi is with Hitachi Research Laboratory, Hitachi, Ltd., Hitachi, Ibaraki 319-12, Japan.

Publisher Item Identifier S 1094-6977(98)02519-X.

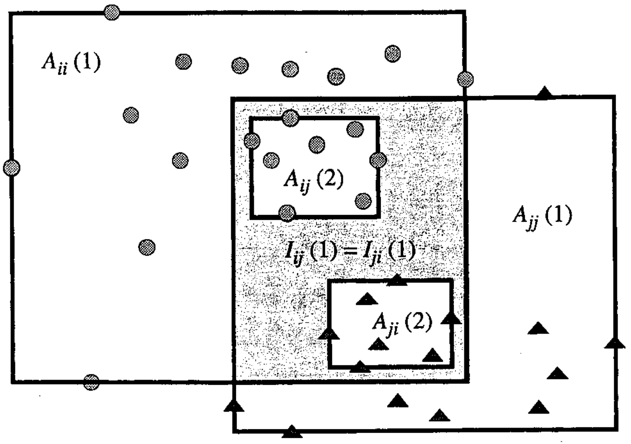

Fig. 1. Recursive definition of activation and inhibition hyperboxes.

Now we define the degree of membership of $\boldsymbol{x}$ belonging to class i by

$$m_i(\boldsymbol{x}) = \exp\left(-\frac{1}{m} d_i^2(\boldsymbol{x})\right). \tag{5}$$

Here, the reason why we divide the squared weighted distance by m is that, from (4), the mean value of $d_i^2(x)/m$ is one for any m, and this makes the membership functions for different numbers of features comparable.

If we classify $\boldsymbol{x}$ into class i when

$$m_i(\boldsymbol{x}) > m_j(\boldsymbol{x}), \qquad \text{for } j = 1, \cdots, n, j \neq i \tag{6}$$

the classification result coincides with that of the classifier discussed in [13] when the data of each class are not divided, i.e., one cluster per class, and when the membership function is calculated by

$$m_i(\boldsymbol{x}) = \exp\left(-\frac{1}{\alpha_i} d_i^2(\boldsymbol{x})\right) \tag{7}$$

where α_i is the tuning parameter and $\alpha_i = 1$. [We note that the classification result is the same even if the argument of the exponential function in (7) is divided by m.] In [13], the efficient training algorithm to determine α_i is discussed.

B. Exception Ratio

In [8], the degree of overlap between two classes was defined by two types of hyperboxes: activation and inhibition, which were recursively generated from the training data. Fig. 1 shows the recursive definition of hyperboxes for the two classes i and j. First, by calculating the minimum and maximum values of the respective class data, activation hyperboxes of level 1 A_{ii} (1) and A_{jj} (1) are defined. Then the overlapping region is defined as the inhibition hyperbox of level 1 I_{ij} (1). Since there are data in the inhibition hyperbox I_{ij} (1), we further define the activation hyperboxes $A_{ij}(2)$ and $A_{ji}(2)$. Since there is no overlap between $A_{ij}(2)$ and $A_{ji}(2)$, we stop defining hyperboxes. The degree of overlap of class i, with respect to class j at level 1, is defined by

$$o_{ij}^{(1)}(F) = \frac{\text{hypervolume of } I_{ij}(1)}{\text{hypervolume of } A_{ii}(1)} \tag{8}$$

where F is the set of features. Since there is no overlap between classes i and j at level 2

$$o_{ij}^{(2)}(F) = 0. \tag{9}$$

Similarly we may define the overlap of ellipsoidal class regions as follows. Assuming a positive value $c(1 \geq c \geq 0)$, we consider the region $\{\boldsymbol{x} | m_i(\boldsymbol{x}) \geq c \text{ and } m_j(\boldsymbol{x}) \geq c\}$ as the overlap between classes i and j. Then we define the degree of overlap of the class i region, with respect to the class j region, by (see Fig. 2)

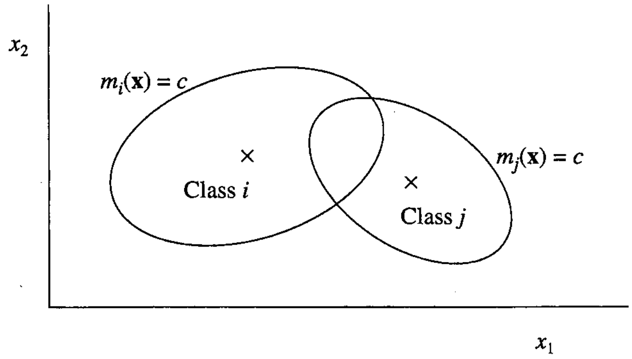

Fig. 2. Approximation of an overlap of class regions.

$$o_{ij}^1(F) = \frac{\displaystyle\int_{(m_i(\boldsymbol{x}) \geq c) \cap (m_j(\boldsymbol{x}) \geq c)} d\boldsymbol{x}}{\displaystyle\int_{(m_i(\boldsymbol{x}) \geq c)} d\boldsymbol{x}}. \tag{10}$$

The degree of overlap given by (10) is c dependent, and it is difficult to determine the proper value of c. Even if we can determine the value of c, it is by no means easy to calculate the integral.

To overcome this problem, we take a probabilistic approach. Let $p_i(\boldsymbol{x})$ be the probability that $\boldsymbol{x}$ belongs to class i. Then we define the degree of overlap of class i, with respect to class j, by

$$o_{ij}^2(F) = \frac{\displaystyle\int_{\boldsymbol{x} \in \text{ class } i} dp_j(\boldsymbol{x})}{\displaystyle\int_{\boldsymbol{x} \in \text{ class } i} dp_i(\boldsymbol{x})}. \tag{11}$$

By integrating $p_j(\boldsymbol{x})$ over $\boldsymbol{x}$ belonging to class i, we obtain the accumulated probability of class j for the class i data. The denominator of (11) normalizes the numerator, and it is one if the probability $p_i(\boldsymbol{x})$ is normalized. To approximate $p_i(\boldsymbol{x})$, we use $m_i(\boldsymbol{x})$ and calculate (11) using the training data. Namely

$$o_{ij}(F) = \frac{\displaystyle\sum_{\boldsymbol{x} \in \text{ class } i} m_j(\boldsymbol{x})}{\displaystyle\sum_{\boldsymbol{x} \in \text{ class } i} m_i(\boldsymbol{x})}. \tag{12}$$

Equation (12) gives the measure in which the class i region overlaps with the class j region. But if any of the class i data are not misclassified into class j, i.e., $m_i(\boldsymbol{x}) > m_j(\boldsymbol{x})$ for $\boldsymbol{x}$ belonging to class i, the overlap given by (12) does not make the classification of class i data difficult. To reflect this, we define the exception ratio by [8]

$$O(F) = \sum_{i,j=1}^{n} p_{ij}\, o_{ij}(F) \tag{13}$$

where

$$p_{ij} = \frac{\text{the number of the class } i \text{ data that are misclassified into class } j}{\text{the total number of the training data}}.$$

The exception ratio given by (13) has the form similar to that defined in [8]. The major difference is that, in the former, the class region is approximated by one ellipsoid, while in the latter, the class region is approximated by nested hyperboxes.

C. Feature Elimination Based on Exception Ratio

The feature elimination discussed here is similar to the one discussed in [8]. The major difference is the stopping criteria. We select features by backward selection search [1], which begins with all of the features and eliminates the most irrelevant feature, as follows. First, each of the features is temporarily eliminated and the exception ratio after each temporary elimination is computed. Then, the feature whose elimination minimizes the exception ratios is deleted permanently. We iterate this procedure and delete features until the stopping criteria discussed below are satisfied.

Let F_{org} denote the set of the original M features where $M \geq 2$; let F^m denote the set of m remaining features; and let F_i^m be the set of m features obtained by temporarily eliminating f_i^{m+1} from F^{m+1}, i.e., $F_i^m = F^{m+1} - \{f_i^{m+1}\}$, where f_i^{m+1} is the ith element in F^{m+1}. Let F_j^{m-1} satisfy

$$O(F_j^{m-1}) = \min_i (O(F_i^{m-1})). \tag{14}$$

Since monotonicity of the exception ratio is not always guaranteed, i.e., the exception ratio may be lower for the deletion of the features (see Figs. 4–6 in Section III), we introduce two criteria for terminating the feature elimination. The first criterion terminates the feature elimination when the exception ratio exceeds that of the original exception ratio

$$\frac{O(F_j^{m-1}) - O(F\text{org})}{O(F\text{org})} \geq \beta \tag{15}$$

where β is a small positive parameter. The second criterion terminated the feature elimination when the exception ratio begins to increase

$$\frac{O(F_j^{m-1}) - O(F^m)}{O(F\text{org})} \geq \delta \tag{16}$$

where δ is a small positive parameter. The stopping criterion (15) is the same as that in [8], while (16) is now added to prevent the deletion algorithm from deleting too many features. In [8], β was set to 0.5. So we use the same value for β. The characteristics of the exception ratio discussed in this paper differ from those in [8], and to avoid deleting too many features, we use $\delta = 0.01$ in the following simulations.

The feature elimination algorithm based on the exception ratio is as follows.

Step 1) Initialize F^m by setting $F^m \leftarrow F_{org}$, hence, $m = M$.
Step 2) Compute $O(F_i^{m-1})$ for $i = 1, \cdots, m$.
Step 3) Find the feature f_j^m that satisfies (14).
Step 4) If (15) or (16) holds, terminate; otherwise, go to Step 5).
Step 5) Set $F^{m-1} \leftarrow F_j^{m-1}$. ($f_j^m$ is permanently eliminated from F^m.)
Step 6) Set $m = m - 1$. If $m = 1$, terminate; otherwise, go to Step 2).

We call the above feature elimination algorithm, based on the exception ratio ERFE with ellipsoids, ERFEE and that in [8], based on ERFE with hyperboxes, ERFEH.

III. Performance Evaluation

We compare the ERFEE with the ERFEH and PCA by using the same data used in [8]: 1) iris data [14], 2) thyroid data [14], 3) numeral data [16], [17], and 4) blood-cell data [18]. The first two data sets are well-known benchmark data for classification, the numeral data are for license plate recognition, and the blood-cell data are for white blood-cell classification, which is a very hard problem since each class represents some stage of blood-cell growth, and thus, the boundaries of some of the classes are very vague. The specifications of the data are listed in the upper part of Table I. For each data set, all of the available data are divided into training data and test data. The training data are used both for eliminating features and for training classifiers. The test data are used for evaluating the recognition rate of the classifiers. In [8], both the discriminant analysis (DA) and the feature selection method that performs backward selection search using interclass Euclidean distance as the class separability measure (EDFE) were shown to be inferior to the PCA for the above four benchmark data sets. Therefore, we do not include their comparison here.

TABLE I
Benchmark Data Specifications and Training Conditions of the Three-Layered Neural Network Classifier

	Iris	Thyroid	Numeral	Blood Cell
No. Inputs	4	21	12	13
No. Classes	3	3	10	12
No. Training Data	75	3772	810	3097
No. Test Data	75	3428	820	3100
No. Hidden Units	3	3	6	18
No. Epochs	1000	10000	4000	15000
No. Runs	10	10	10	3

Three classifiers are used, namely, the fuzzy classifier with hyperbox regions [9], the fuzzy classifier with ellipsoidal regions [13], and a three-layered neural network classifier [15]. Unless explicitly specified, the following sets of parameters are used for the fuzzy classifiers and the neural network classifier, respectively.

1) Fuzzy classifier with hyperbox regions:
expansion parameter (which controls the expansion size of the inhibition hyperbox) = 0.001 and sensitivity parameter (which controls the slope of the membership function) = 1.
2) Fuzzy classifier with ellipsoidal regions:
one cluster per class and the maximum number of misclassifications allowed for tuning one cluster parameter is ten.
3) The neural network classifier:
learning rate = 1 and momentum = 0.

The training conditions of this classifier for the four data sets are listed in the lower part of Table I. Since the recognition rate of the neural network classifier varies according to the initial weights, we use the average recognition rate for a set of three runs for the blood-cell data or ten runs for the other data sets, each run having initial weights randomly assigned between -0.1 and 0.1. The numbers of hidden units and training epochs used in our experiments are the same as those in [8], except for the number of training epochs for the blood-cell data. (We use 15 000 epochs instead of 6000 to guarantee stable recognition rates when features are deleted. But we make only three runs instead of ten because Hitachi's 30 MIPS mainframe computer M-680 takes 14 h for the former.)

The parameters used for the ERFEH are the same as those used in [8], i.e., $\varepsilon = 0.001$ (which specifies the minimum edge length) and $\beta = 0.5$. And the parameters for the ERFEE are $\beta = 0.5$ and $\delta = 0.01$.

Table II lists the numbers of features selected by the ERFEE and ERFEH as well as the associated accumulation of eigenvalues by the PCA when the number of features determined by the ERFEH is used. In [8], for the iris data, the number of selected features was two. This shows the robustness of the selection method even when

TABLE II
NUMBER OF FEATURES SELECTED BY ERFEH AND ERFEE AND THE ASSOCIATED ACCUMULATION OF EIGENVALUES (ACC. EV.) BY THE PRINCIPAL COMPONENT ANALYSIS (PCA)

Data Set	Number of Features		Acc. Ev. (%)
	ERFEH	ERFEE	
Iris	3	3	99.65
Thyroid	5	7	73.33
Numeral	7	8	93.45
Blood Cell	10	10	99.53

TABLE III
RECOGNITION RATE OF THE FUZZY CLASSIFIER WITH HYPERBOX REGIONS (IN %)

Data Set	Original Features	Reduced Features		
		ERFEH	ERFEE	PCA
Iris	92	93.33	93.33	90.67
Thyroid	99.15	99.01	99.21	85.82
Numeral	99.63	99.51	99.51	98.90
Blood	85.16	85.45	84.71	83.23

TABLE IV
RECOGNITION RATE OF THE FUZZY CLASSIFIER WITH ELLIPSOIDAL REGIONS (IN %)

Data Set	Original Features	Reduced Features		
		ERFEH	ERFEE	PCA
Iris	97.33	98.67	98.67	94.67*
Thyroid	95.60	96.82	96.65	92.42
Numeral	99.39	99.15	99.39	99.39
Blood	91.65	90.61	91.39	89.87

*: Without tuning the recognition rate is 98.67%.

the exception ratio exceeded the stopping criterion (15). Here we use the numbers of features given by the stopping criterion (15) for the ERFEH and the stopping criteria (15), (16) for the ERFEE. For the iris and the blood-cell data, the numbers of selected features are the same for the ERFEH and ERFEE. For the iris data, the same feature (the second feature) is deleted. For the blood-cell data, one feature out of the three that are deleted is the same. For the thyroid and numeral data, the ERFEH eliminates more features than the ERFEE. Among the first five features that are deleted from the numeral data, three features are the same for both methods, and among the five features that remain in the thyroid data, four features are the same for both methods.

Tables III–V list the recognition rates of one classifier for each of the test data sets when the original features, features selected by the ERFEH, features selected by the ERFEE, and features transformed by the PCA are all used. For the ERFEE and the PCA, the same number of features as the ERFEH listed in Table II are used. Table III lists performance of the fuzzy classifier with hyperbox regions. In the following, we abbreviate the recognition rate using the features deleted by the ERFEH or ERFEE as the recognition rate with hyperboxes or ellipsoids and the recognition rate using the features transformed by the PCA as the recognition rate with the PCA. The recognition rates with the PCA are worse than those with the others for the four data sets, while the recognition rates with hyperboxes and ellipsoids are comparable.

TABLE V
RECOGNITION RATE OF THE MULTILAYERED NEURAL NETWORK (IN %)

Data Set	Original Features	Reduced Features		
		ERFEH	ERFEE	PCA
Iris	97.47	97.20	97.20	96.00
Thyroid	98.23	98.61	98.31	92.63
Numeral	99.48	99.40	99.51	99.28
Blood	89.18	88.62	88.38	89.83

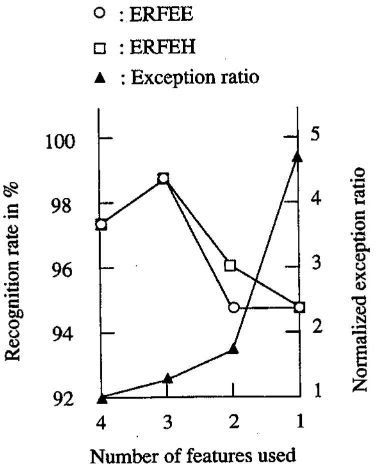

Fig. 3. Comparison of feature elimination for iris data. The recognition rates for test data are evaluated by the fuzzy classifier with ellipsoidal regions.

Table IV lists the recognition rates of the fuzzy classifier with ellipsoidal regions. Except for the numeral data, the recognition rates with the PCA are the worst. For the numeral data, the recognition rate with hyperboxes is 0.28% lower than that of the other two. Table V lists the recognition rates of the multilayered neural network classifier. Except for the blood-cell data, the recognition rates with the PCA are the worst.

In summary, the recognition rates with the PCA are the worst for the iris data and the thyroid data for the three classifiers, and thus, feature elimination by the PCA is not as robust as that by the ERFEE and ERFEH; the last two methods are comparable. As for the classifier performance, the fuzzy classifier with hyperbox regions performs best for the thyroid data, the fuzzy classifier with ellipsoidal regions performs best for the iris data and the blood-cell data, and the neural network classifier performs best for the numeral data. Although the recognition rate of the fuzzy classifier with ellipsoidal regions for the thyroid data is the poorest, this does not affect feature elimination. The ERFEE works for the thyroid data.

Now we compare the robustness of the ERFEE with that of the ERFEH for the four data sets. Fig. 3 shows, on the left-side ordinate, the recognition rates of the fuzzy classifier with ellipsoidal regions for the iris data when the features are deleted by the ERFEE and ERFEH. The right-side ordinate plots the exception ratio with ellipsoids normalized by that with the original features. The normalized exception ratio increases as the features are deleted. When two features are deleted, the exception ratio satisfies (15). The first eliminated features by both methods are the same, but the second eliminated features are different. When three data are deleted, the remaining feature is the same for both methods. Thus, for the iris data, the ERFEH performs better than ERFEE, when the features are deleted while satisfying (15).

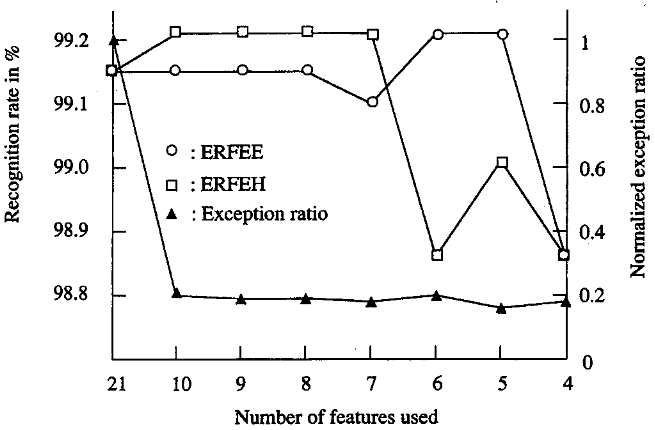

Fig. 4. Comparison of feature elimination for thyroid data. The recognition rates for test data are evaluated by the fuzzy classifier with hyperbox regions.

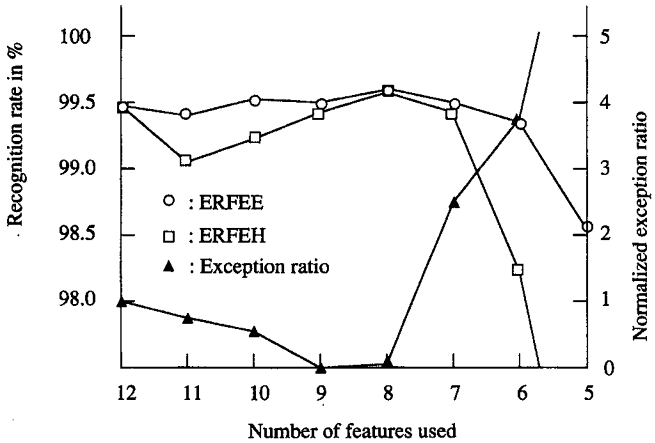

Fig. 5. Comparison of feature elimination for numerical data. The recognition rates for test data are evaluated by the multilayered neural network classifier.

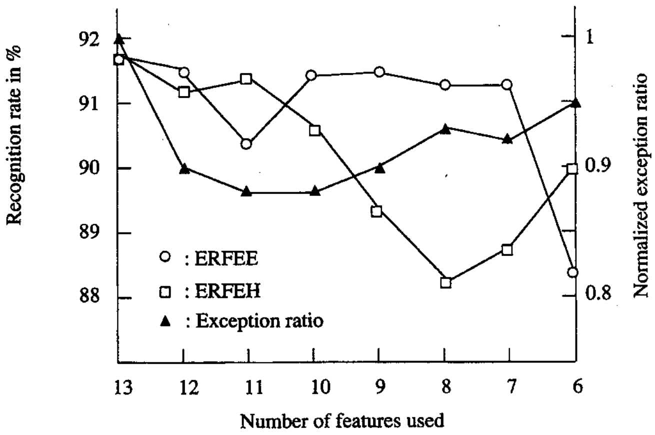

Fig. 6. Comparison of feature elimination for blood-cell data. The recognition rates for test data are evaluated by the fuzzy classifier with ellipsoidal regions.

Fig. 4 shows the recognition rates of the fuzzy classifier with hyperbox regions for the thyroid data and the normalized exception ratio. The normalized exception ratio does not exceed 1.0 until only one feature remains. Thus, if we use (15), the algorithm does not stop. But if we use (16) as well as (15), the normalized exception ratio increases when one feature is eliminated from the remaining seven features and the exception ratio satisfies (16). Therefore, we can select seven features. This elimination is rather conservative, as seen from the figure. The recognition rates by the ERFEH are better than those by the ERFEE when seven to ten features are used, but this is reversed when five or six features are used. Although the recognition rate of the fuzzy classifier with ellipsoidal regions is the poorest among the three classifiers, as listed in Tables III and IV, the exception ratio with ellipsoids well reflects the complexity of the class regions of the thyroid data.

Fig. 5 shows the recognition rates of the multilayered neural network for the numeral data and the normalized exception ratio. The normalized exception ratio starts to increase when one feature is eliminated from the remaining nine features and the exception ratio satisfies (15) when one more feature is deleted. Thus, eight features are selected. By the ERFEH, seven features are selected, as listed in Table II. Although the number of features selected by the ERFEE is larger than that by the ERFEH, the recognition rates are higher for five to 11 features.

Fig. 6 shows the recognition rates of the fuzzy classifier with ellipsoidal regions for the blood-cell data and the normalized exception ratio. The normalized exception ratio starts to increase when one feature is eliminated from the remaining ten features and the exception ratio satisfies (16). Thus, ten features are selected. The recognition rates drop for seven to nine features using the ERFEH. The high recognition rates are maintained for seven to nine features using the ERFEE. According to our analysis, the distribution of blood-cell data is not parallel to the feature axes. Thus, the ERFEH does not fit this type of data, while the ERFEE does not have this weakness.

IV. Discussion

Both ERFEE and ERFEH are local optimization methods. They delete each feature, one at a time, which minimizes the exception ratio. Thus, global optimality of the features selected by those methods is not guaranteed. But according to the simulations, both methods showed better performance, in most cases, than the PCA. Comparison of the ERFEE with the ERFEH pointed out that the former method did delete more features than the latter, except for the iris data; this was especially evident for the blood-cell data where the distribution of class data was not parallel to the feature axes; since the ERFEH is based on hyperboxes, which are parallel to the feature axes, it is unsuitable for this type of data. The weakness of the ERFEE is the stopping criteria given by (15) and (16); they usually give conservative elimination. Their improvement should be the subject of a future study.

The features were eliminated according to analysis of the ellipsoidal regions approximated using the training data. Thus, the ellipsoidal regions may not be a good approximation of the class regions for the test data. Now we need to consider why the ERFEE gave an inferior performance to that of the ERFEH for the iris data. As for the iris data, the second, fourth, and first features were successively deleted by the ERFEE, while the second, first, and fourth features were successively deleted by the ERFEH. The normalized exception ratio was 1.70 when the second and fourth features were deleted, and the normalized exception ratio was 1.94 when the second and first features were deleted. Thus, the fourth feature was deleted in addition to the second feature. But if we compare the normalized exception ratio of the test data, the normalized exception ratio was 2.80 when the second and first features were deleted, and the normalized exception ratio was 6.11 when the second and fourth features were deleted; this clearly indicates that the recognition rate of the test data eliminating the second and fourth features might be worse than that eliminating the second and first features.

The fuzzy classifier with ellipsoidal regions does not perform well when the distribution of data deviates from the Gaussian distribution. This happens when features include discrete inputs, such as those of the thyroid data. It is interesting to note that, although performance of the fuzzy classifier with ellipsoidal regions for the thyroid data is not good, the ERFEE successfully deletes the features.

In general, selecting features by the backward selection search is inefficient, especially when the number of initial features is large. But since the calculation of the exception ratio is not complicated, both ERFEE and ERFEH are relatively efficient for a medium to large number of initial features. For example, for the thyroid data with 21 features, ERFEH selected features in 2 min (in turnaround time) by using a workstation (Sun 4/20 model 71) under a multiuser environment.

V. Conclusions

In this paper, we proposed a method of selecting features based on the analysis of class regions approximated by ellipsoids. First, for a given set of features, each class region was approximated by an ellipsoid with the center and the covariance matrix calculated by the data belonging to the class. Then, the exception ratio was defined to represent the degree of overlap in the class regions approximated by ellipsoids. From the given set of features, we temporally deleted each feature, one at a time, and calculated the exception ratio. Then, the feature whose associated exception ratio was the minimum was deleted permanently. We iterated this procedure while the exception ratio or its increase was within a specified value by feature deletion. The simulation results showed that our method was better than the PCA, and its performance was better than our previous method, especially when the distributions of class data were not parallel to feature axes.

Appendix

We show that (4) holds, assuming that Q_i is nonsingular. Let P_i be the orthogonal matrix that diagonalizes Q_i. Namely

$$P_i^t Q_i P_i = \mathrm{diag}(\lambda_1, \cdots, \lambda_m) \tag{a1}$$

where $P_i P_i^t = E$, E is the unit matrix, diag denotes the diagonal matrix, and $\lambda_1, \cdots, \lambda_m$ are the eigenvalues of Q_i. From (a1)

$$Q_i = P_i\, \mathrm{diag}(\lambda_1, \cdots, \lambda_m) P_i^t \tag{a2}$$

$$Q_i^{-1} = P_i\, \mathrm{diag}(\lambda_1^{-1}, \cdots, \lambda_m^{-1}) P_i^t. \tag{a3}$$

Let

$$\tilde{\boldsymbol{x}}_i = P_i^t\,(\boldsymbol{x} - \boldsymbol{c}_i). \tag{a4}$$

Then from (2) and (a4), (a1) becomes

$$\frac{1}{N_i} \sum_{\boldsymbol{x} \in \text{ class } i} \tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_i^t = \mathrm{diag}\ (\lambda_1, \cdots, \lambda_m). \tag{a5}$$

Thus, for the diagonal elements of (a5)

$$\frac{1}{N_i} \sum_{\boldsymbol{x} \in \text{ class } i} \tilde{x}_{ik}^2 = \lambda_k, \qquad k = 1, \cdots, m. \tag{a6}$$

From (3), (a3), and (a4), the left-hand side of (4) becomes

$$\begin{aligned} \frac{1}{N_i} \sum_{\boldsymbol{x} \in \text{ class } i} d_i^2(\boldsymbol{x}) &= \frac{1}{N_i} \sum_{\boldsymbol{x} \in \text{ class } i} \tilde{\boldsymbol{x}}_i^t \mathrm{diag}\ (\lambda_l^{-1}, \cdots, \lambda_m^{-1}) \tilde{\boldsymbol{x}}_i \\ &= \frac{1}{N_i} \sum_{\boldsymbol{x} \in \text{ class } i} \sum_{k=1}^{m} \lambda_k^{-1} \tilde{x}_{ik}^2. \end{aligned} \tag{a7}$$

Thus, from (a6) and (a7), (4) holds.

Acknowledgment

The authors are grateful to Dr. N. Matsuda of Kawasaki Medical School for providing the blood-cell data. The thyroid data set was obtained from the machine learning databases at the University of California at Irvine (http://ics.uci.edu:pub/machine-learning-databases). The authors also thank P. Murphy and D. Aha who organized the databases.

References

[1] K. Fukunaga, *Introduction to Statistical Pattern Recognition,* 2nd ed. San Diego, CA: Academic, 1990.

[2] H. A. Malki and A. Moghaddamjoo, "Using the Karhunen–Loe've transformation in the back-propagation training algorithm," *IEEE Trans. Neural Networks,* vol. 2, pp. 162–165, Jan. 1991.

[3] C. Lee and D. A. Landgrebe, "Feature extraction based on decision boundaries," *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 15, no. 4, pp. 388–400, 1993.

[4] ——, "Decision boundary feature extraction for nonparametric classification," *IEEE Trans. Syst., Man, Cybern.,* vol. 23, pp. 433–444, Feb. 1993.

[5] J. Kittler, "Feature selection and extraction," in *Handbook of Pattern Recognition and Image Processing,* T. Y. Young and K. S. Fu, Eds. San Diego, CA: Academic, 1986, pp. 59–83.

[6] S. K. Pal and B. Chakraborty, "Fuzzy set theoretic measure for automatic feature evaluation," *IEEE Trans. Syst., Man, Cybern.,* vol. SMC-16, pp. 754–760, May 1986.

[7] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Trans. Neural Networks,* vol. 5, pp. 537–550, July 1994.

[8] R. Thawonmas and S. Abe, "A novel approach to feature selection based on analysis of fuzzy regions," *IEEE Trans. Syst., Man, Cybern. B,* vol. 27, pp. 196–207, Apr. 1997.

[9] 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, Feb. 1995.

[10] P. Pudil, J. Novovicová, and J. Kittler, "Floating search methods in feature selection," *Pattern Recognit. Lett.,* vol. 15, no. 11, pp. 1119–1125, 1994.

[11] D. Zongker and A. Jain, "Algorithms for feature selection: An evaluation," in *Proc. 13th Int. Conf. Pattern Recognit.,* Vienna, Austria, 1996, pp. 18–32.

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

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

[14] S. M. Weiss and I. Kapouleas, "An empirical comparison of pattern recognition, neural nets, and machine learning classification methods," in *Proc. IJCAI-89,* pp. 781–787.

[15] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructures of Cognition, Vol. 1: Foundations,* D. E. Rumelhart and J. L. McClelland, Eds. Cambridge, MA: MIT Press, 1986, pp. 318–362.

[16] M. Takatoo *et al.,* "Gray scale image processing technology applied to vehicle license number recognition system," in *Proc. Int. Workshop Ind. Applicat. Machine Vision Machine Intell.,* Feb. 1987, pp. 76–79.

[17] H. Takenaga *et al.,* "Input layer optimization of neural networks by sensitivity analysis and its application to recognition of numerals," *Trans. Inst. Elec. Eng. Japan,* vol. 111-D, no. 1, pp. 36–44, 1991, in Japanese. (Translated into English by Scripta Technica, Inc., *Elec. Eng. Japan,* vol. 111, no. 4, pp. 130–138, 1991.)

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