Discriminative Learning for Minimum Error Classification

Biing-Hwang Juang, *Fellow, IEEE,* and Shigeru Katagiri, *Member, IEEE*

Abstract—**Recently, due to the advent of artificial neural networks and learning vector quantizers, there is a resurgent interest in reexamining the classical techniques of discriminant analysis to suit the new classifier structures. One of the particular problems of interest is minimum error classification in which the misclassification probability is to be minimized based on a given set of training samples. In this paper, we propose a new formulation for the minimum error classification problem, together with a fundamental technique for designing a classifier that approaches the objective of minimum classification error in a more direct manner than traditional methods. We contrast the new method to several traditional classifier designs in typical experiments to demonstrate the superiority of the new learning formulation. The method can be applied to other classifier structures as well. Experimental results pertaining to a speech recognition task are also provided to show the effectiveness of the new technique.**

I. Introduction

PATTERN classification, particularly in the area of linear discriminant analysis, is a very well-studied topic with most of the original developments completed in the 1960's (see [1]–[5]). Recently, due to the advent of artificial neural networks (ANN) [6] and learning vector quantizers (LVQ) [7], there is a resurgent interest in reexamining the classical techniques to suit the new classifier structures. In this paper, therefore, we address specifically the problem of minimum error classification, propose a fundamental technique for designing a classifier that achieves minimum classification error, and contrast it to popular classifier structures, such as the perceptron [8], so that the new classifiers can be better utilized.

Consider a set of observations $\mathcal{L} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \cdots, \boldsymbol{x}_N\}$, where each $\boldsymbol{x}_i$ is a K-dimensional vector and is known to belong to one of M classes C_i, $i = 1, 2, \cdots, M$. A classifier normally consists of a set of parameters and a decision rule. The task of minimum error classifier design is to find the classifier parameter set, denoted by Λ, and the accompanying decision rule, based on the given sample set $\mathcal{L}$, such that the probability of misclassifying any $\boldsymbol{x}$ is minimized. Probability of misclassification is often empirically approximated by the recognition error rate, defined as the number of recognition errors incurred in classifying $\mathcal{L}$, normalized by the size of $\mathcal{L}$. When there is a penalty or cost associated with a misclassification, the objective is then to minimize the expected cost accordingly.

Bayes decision theory [1] is a fundamental statistical approach to the classification problem and is often the basis of many pattern classification techniques. Suppose we have full knowledge of the *a posteriori* probability $P_\Lambda(C_i|\boldsymbol{x})$, which is defined by the parameter set Λ as denoted by the subscript. (Note that this assumption also implies that the true *a posteriori* probability can be parametrized by Λ). The Bayes decision rule

$$C(\boldsymbol{x}) = C_i \quad \text{if } P_\Lambda(C_i|\boldsymbol{x}) = \max_j P_\Lambda(C_j|\boldsymbol{x}) \tag{1}$$

where $C(\cdot)$ denotes a classification operation, is known to lead to minimum misclassification probability. The rule is often written in terms of the *a priori* and the conditional probabilities as

$$C(\boldsymbol{x}) = C_i \quad \text{if } p_\Lambda(\boldsymbol{x}|C_i)P_\Lambda(C_i) = \max_j p_\Lambda(\boldsymbol{x}|C_j)P_\Lambda(C_j). \tag{2}$$

Since the exact probability measure is rarely known in real situations, the problem of optimal classifier design thus becomes that of estimating the *a priori* and the conditional probabilities, defined by the parameter set Λ, using the design samples $\mathcal{L}$. This empirical approach has been widely followed in the past because the subject of distribution estimation is a well-treated topic in statistics. The fundamental assumption in this approach is that the form of the distributions as functions of the parameter set Λ is known and that given a sufficient design sample set there is a good method to estimate correctly the unknown parameters Λ.

An alternative to the Bayes decision approach is to use discriminant functions in lieu of the probabilities. This requires a set of discriminant functions, $g_i(\boldsymbol{x}; \Lambda)$, $i = 1, 2, \cdots, M$, defined by the parameter set Λ, instead of explicit knowledge of the probability distributions. Unlike the Bayes approach, the problem of "optimal" classifier design becomes that of finding the right parameter set for the discriminant functions to minimize the "sample risk" [1], which is defined as the average cost incurred in classifying the design sample $\mathcal{L}$. The cost is usually defined on a pair of class indices (i, j) where i and j are the correct class index and the identified/recognized class index, respectively, indicating the penalty in mis-

Manuscript received August 21, 1991; revised February 6, 1992.

B.-H. Juang is with AT&T Bell Laboratories, Murray Hill, NJ 07974.
S. Katagiri is with Advanced Telecommunications Research, Kyoto, Japan 61902.

IEEE Log Number 9203223.

classifying a C_i class observation as a C_j class observation. Note that, as mentioned previously, the sample risk is not the expected cost (or cost expectation, see (24)) because the size of the design sample is usually finite; it can be considered an empirical estimate of the expected cost, however. While suboptimality may still occur because of improper choice of the discriminant functions, as in the case of incorrect distribution assumption in the Bayes approach, the discriminant function based method usually offers implementational simplicity [1] and with the advent of new classifier structure, it may be possible to circumvent the data consistency issue (see Section V). In this paper, we shall primarily focus on the design algorithms.

The difficulty associated with the discriminant approach lies in the derivation of a minimum-cost discriminant. A proper discriminant needs to be suitable for incorporation in an objective function for optimization. The sample risk, of which the number of classification errors is one of the simplest cases with a zero-one function as the misclassification cost [1], is obviously a piecewise constant function of the classifier parameter Λ and thus a poor candidate for optimization by a numerical search method. Traditionally, the discriminant based classifier design is formulated as an optimization problem aiming at minimization of some criterion functions that are analytically more tractable than the sample risk. Popular choices of these criterion functions include the perceptron criterion function and sum of squared errors (or minimum squared error (MSE)), for example. These criterion functions, as will be elaborated in Section II, do not generally lead to a minimum error probability classifier, although one can vigorously discuss the convergence properties of the solution as obtained by numerical search algorithms.

In this paper, we propose a new way of deriving the discriminant such that the result of the optimization procedure will be controllably consistent with the minimum sample risk objective. The new discriminant makes proper use of the L_p norm and is a continuous function of the classifier parameters, suitable for gradient-type numerical search. We shall also propose a descent search algorithm for optimizing the minimum error objective. This combination of a new discriminant, directly related to the minimum error objective, and the descent algorithm would then allow us to circumvent the difficulties encountered in most of the traditional techniques and address the optimal classifier design problem in a straightforward manner. We shall further point out that the algorithm can be shown to produce asymptotically a solution consistent with the minimum error result, one important step beyond the minimum sample risk objective. However, the proof of this asymptotic result will be provided in a separate paper [9] for clarity of presentation. In addition, we shall discuss how the new discriminant can be incorporated in new classifier structures, in particular, a multilayer perceptron, for an expanded application prospect.

The paper is organized as follows. In the next section, to provide a necessary analytical background, we summarize conventional criterion functions, particularly those related to linear discriminant, that have been extensively studied in the past. We then propose a new discriminant and formulate the minimum classification error problem in a manner suitable for optimization in Section III. In Section IV, we present a gradient search algorithm for solving the optimization problem, generalize the algorithm so that expected cost minimization, rather than sample risk minimization, can be addressed, and further discuss optimality as well as consistency issues associated with the algorithm. We then suggest in Section V how the error back-propagation technique can be revised for multilayer perceptron (MLP) training in order to accomplish the minimum classification error objective. In Section VI, we compare experimentally the new discriminant and the cost functions with traditional criterion functions and the associated classifier solutions in typical pattern recognition tasks. We show in particular the effect of minimum classification error criterion in contrast to the usual minimum squared error (MSE) and perceptron criteria. These comparisons are helpful in gaining insights on how classifiers can be better constructed for minimum classification error performance. In Section VII, we report a speech recognition experiment in which perceptrons with nonlinearity are compared with the new discriminative learning technique in real applications. We finally conclude the paper in Section VIII.

II. Discriminant Functions

We provide a general analytic background in this section using linear discriminant functions for simplicity. A linear discriminant function of a K-dimensional feature vector $\boldsymbol{x}$ has the form $\boldsymbol{w}^*\boldsymbol{x} + w_0$ where * denotes matrix transpose. The weight vector and the threshold, $\boldsymbol{w}$ and w_0, respectively, are defined for each class, resulting in M discriminant functions and a parameter set $\Lambda = \{\boldsymbol{w}_1, w_{01}, \boldsymbol{w}_2, w_{02}, \cdots, \boldsymbol{w}_M, w_{0M}\}$ which constitute the classifier. Each discriminant function can be written as

$$g_i(\boldsymbol{x}; \Lambda) = \boldsymbol{w}_i^*\boldsymbol{x} + w_{0i} = \lambda_i^*\boldsymbol{y} \tag{3}$$

where $\lambda_i^*, = [\boldsymbol{w}_i^*, w_{0i}]$ and $\boldsymbol{y}^* = [\boldsymbol{x}^* \; 1]$. The classifier uses the following decision rule:

$$C(\boldsymbol{x}) = C_i \quad \text{if } g_i(\boldsymbol{x}; \Lambda) = \max_j g_j(\boldsymbol{x}; \Lambda). \tag{4}$$

Since the discriminant functions are linear, the decision boundaries are hyperplanes. The linear discriminant function of (3) can be generalized by augmenting the feature vector $\boldsymbol{x}$ with higher order nonlinear terms such that $g_i(\boldsymbol{x}; \Lambda)$ becomes a polynomial in terms of the elements of $\boldsymbol{x}$. However, this generalization does not change the basic structure of the discriminant function. We shall stay with the expression of (3) in the following without loss of generality.

The classifier parameters are to be determined based on a given sample set $\mathcal{L}$ of N observations. The correct class labeling/association for each observation in the set is as-

sumed to be known. If there exist a set of weight vectors and thresholds such that classification based on the above discriminant functions and the decision rule produces no error at all, the sample set is called linearly separable. Otherwise, it is linearly nonseparable. For the simplest two-category case, by recognizing that $\lambda^* y_i < 0$ is identical to $\lambda^*(-y_i) > 0$, we are equivalently seeking a vector λ, normal to the separating plane, such that $\lambda^* y_i' > 0$ for all i where $y_i' = y_i$ if $x_i \in C_1$, and $y_i' = -y_i$ if $x_i \in C_2$. Linear separability thus means the existence of such a separating plane.

Determination of the classifier parameters is usually formulated as a problem of minimizing some analytically tractable scalar criterion functions, instead of the sample risk, such that the linear inequalities $\lambda^* y > 0$ can be readily solved by a gradient descent procedure. In the following, we summarize three essential criterion functions and discuss properties of the corresponding solutions as obtained by appropriate descent algorithms. Also, for brevity, we shall limit our discussion to two-category cases in this section. For multicategory considerations, Kesler's construction of the equivalent problem [4], of course, is advisable. Details can be found in [1].

A. The Perceptron Criterion Function

The perceptron criterion function is defined as

$$J_p(\lambda) = \sum_{y \in \mathcal{Y}} (-\lambda^* y) \tag{5}$$

where the summation is over the set $\mathcal{Y}$ of observations that are misclassified by Λ. The function is proportional to the sum of the distances from the misclassified observations to the separating plane. Note that the gradient of J_p is not continuous.

B. Selective Squared Distance Criterion

The squared distance criterion is very similar to the perceptron criterion and is defined as

$$J_q(\lambda) = \sum_{y \in \mathcal{Y}} (\lambda^* y)^2 \tag{6}$$

or

$$J_q'(\lambda) = \tfrac{1}{2} \sum_{y \in \mathcal{Y}'} \frac{(\lambda^* y - b)^2}{\| y \|^2} \tag{7}$$

where the summation is also over the set of misclassified observations. (For J_q' of (7), $\mathcal{Y}'$ is the set of observations for which $\lambda^* y \leq b$.) This criterion function leads to a descent algorithm known as the relaxation rule [11].

C. The Minimum Squared Error Criterion

Unlike the above two criterion functions which consider only the misclassified observations, the minimum squared error (MSE) criterion takes into account the entire design sample and is defined as

$$J_s(\lambda) = \sum_{i=1}^{N} (\lambda^* y_i - b_i)^2 \tag{8}$$

where the margin b_i is an arbitrarily specified vector with positive elements (and may be irrelevant to the observation index i). Many solution procedures are available for this well-studied criterion. In terms of classification, however, the solution depends on the choice of the margin vector b.

D. Properties of the Solution

Without making explicit all the related solution procedures, we shall attempt to discuss key properties of the solution to the optimization problem for the above three essential criterion functions. These properties are convergence, nonseparable behavior, and consistency with the minimum error objective.

It can be shown [1] that there exist gradient search procedures that converge to the right solution for the perceptron and the selective squared distance criterion functions when the design sample set is linearly separable. It is intuitively clear that the solution procedures for these criteria aim at correcting the errors, since the summation in (5)–(7) is over misclassified observations. If the design sample set is linearly separable, a relentless search would reach an error-free solution. Note that the error free solution here is in reference to the design sample only, but not to an independent test data set. When the design sample set is not linearly separable, no vector can perfectly separate the design sample and these procedures can never stop, yielding only a sequence of weight parameters, any of which may or may not be a useful solution to the classification problem.

Minimization of the squared error criterion is a better understood problem and many well-known procedures will lead to a solution that minimizes J_S, regardless of the linear separability of the design sample. The problem with the MSE procedures is that minimization of MSE does not necessarily lead to minimum classification error. Even if the design sample set is linearly separable, there is no guarantee that the solution corresponds to a separating plane for error-free classification, unless the margin vector b is carefully chosen. The celebrated Ho–Kashyap procedure [12] that includes adjustment rules for the margin vector b has been shown to be able to converge to a solution corresponding to a separating plane when the design sample set is linearly separable. For nonseparable cases, the inconsistency between the MSE solution and a minimum error classifier remains.

One interesting insight pertaining to the MSE solution is that with a fixed, properly chosen margin vector b, the discriminant approaches a minimum mean-squared-error approximation to the Bayes discriminant function

$$g_0(x) = P(C_1|x) - P(C_2|x) \tag{9}$$

asymptotically as the number of design observations grows [1], [13]. The quality of the approximation depends on the form of the generalized linear discriminant function which is a polynomial in the elements of the feature vector x. While this property has a certain theoretical

appeal, the fact remains that the discriminant function of a prescribed form that best approximates the Bayes discriminant of (9) at a finite set of sample points does not necessarily minimize the misclassification rate or error probability. For multiclass cases where $M > 2$, as will be pointed out shortly, there was a clear difficulty in defining a reasonable "Bayes" discriminant that combines the *a posteriori* probabilities into a well-behaved function and thus the above approximation interpretation of the MSE solution is of little significance. Since our objective is to find a classifier that achieves minimum error probability, the inadequacy of these traditional methods and criterion functions thus becomes clear.

III. Minimum Classification Error Discriminant

The traditional discriminant formulation above can be stated in two steps: definition of the discriminant function and incorporation of the discriminant function in a scalar criterion suitable for a gradient-type search procedure to find a solution, if the procedure does converge. The inadequacy of this traditional formulation lies in the fact that the decision rule does not appear in a functional form in the overall criterion function for easy optimization and that there is an inconsistency between the chosen scalar criterion function and the desired minimum error probability objective. Here, we propose a new way of deriving the objective criterion for a discriminant based approach to mend the above shortcomings.

We use a three-step procedure to derive the objective criterion. As with the traditional approaches, the form of the discriminant functions $g_i(\boldsymbol{x}; \Lambda)$ are first prescribed. The classifier makes its decision for each input $\boldsymbol{x}$ by choosing the largest of the discriminants evaluated on $\boldsymbol{x}$. This decision process needs to be expressed in a functional form such that further optimization can be easily accomplished. We thus in the second step introduce a misclassification measure which allows us to embed the decision process in the overall minimum classification error formulation.

The simplest form of a misclassification measure appears to be the Bayes discriminant defined for the two-category classification case:

$$d(\boldsymbol{x}) = P(C_2|\boldsymbol{x}) - P(C_1|\boldsymbol{x}) \tag{10}$$

where $P(C_i|\boldsymbol{x})$'s are the *a posteriori* probabilities and are assumed to be known. Intuitively, this enumerates how likely a class 1 observation is misclassified as a class 2 observation and the optimal decision boundary is accomplished by a solution to the equation $d(\boldsymbol{x}) = 0$. For multicategory cases ($M > 2$) with unknown distributions, it is not as straightforward to define a misclassification measure as the above two-category Bayes discriminant. One proposal by Amari [14] is to define the misclassification measure by

$$d_k(\boldsymbol{x}) = \sum_{i \in S_k} \frac{1}{m_k} [g_i(\boldsymbol{x}; \Lambda) - g_k(\boldsymbol{x}; \Lambda)] \tag{11}$$

where $S_k = \{i \,|\, g_i(\boldsymbol{x}; \Lambda) > g_k(\boldsymbol{x}; \Lambda)\}$, the set of "confusing classes," and m_k is the number of confusing classes in S_k. This misclassification measure apparently is motivated by the Bayes discriminant of (10). However, since S_k is not a fixed set, i.e., it varies with the parameter set Λ and $\boldsymbol{x}$, the misclassification measure of (11) is discontinuous in Λ and is not differentiable. For gradient algorithms, this is not very desirable.

There are many ways to define a misclassification measure that is continuous with respect to the classifier parameters. One reasonable possibility is as follows:

$$d_x(\boldsymbol{x}) = -g_k(\boldsymbol{x}; \Lambda) + \left[\frac{1}{M-1} \sum_{j, j \neq k} g_j(\boldsymbol{x}; \Lambda)^\eta\right]^{1/\eta} \tag{12}$$

where η is a positive number. (In most applications, g_j's are assumed to be positive.) This misclassification measure resembles the measure of (11) in that the decision rule is being enumerated. The measure of (12), however, is continuous and offers a fair amount of flexibility. By varying the value of η, one can take all the potential classes into consideration, to a various degree, in the search of the classifier parameter Λ. One extreme case is when η approaches ∞, the misclassification measure becomes

$$d_k(\boldsymbol{x}) = -g_k(\boldsymbol{x}; \Lambda) + g_i(\boldsymbol{x}; \Lambda) \tag{13}$$

where C_i is the class with the largest discriminant value among those classes other than C_k, because $(M-1)^{1/\infty} \cong 1$. Obviously in this case, $d_k(\boldsymbol{x}) > 0$ implies misclassification and $d_k(\boldsymbol{x}) \leq 0$ means correct decision. In this way, the decision rule becomes a judgement on a scalar value.

To complete the definition of the objective criterion, the above misclassification measure is used in the third step where the minimum error objective is formulated. A general form of the cost function can be defined as

$$\ell_k(\boldsymbol{x}; \Lambda) = \ell_k(d_k(\boldsymbol{x})) \tag{14}$$

which is expressed as a function of the misclassification measure. (The formulation of (14) was also introduced by Amari [14].) Note that the cost function ℓ_k and the misclassification measure d_k can be defined individually for each class k for generality. For minimum error classification, the following cost functions are merely two of several possibilities:

a) Exponential [14]:

$$\ell_k(d_k) = \begin{cases} (d_k)^\zeta, & d_k > 0 \\ 0, & d_k \leq 0 \end{cases} \tag{15}$$

where $\zeta > 0$ and $\zeta \to 0$;

b) Translated sigmoid:

$$\ell_k(d_k) = \frac{1}{1 + e^{-\xi(d_k + \alpha)}}, \qquad \xi > 0. \tag{16}$$

Both functions are smoothed zero-one cost functions suitable for gradient algorithms. Clearly, when $d_k(\boldsymbol{x}) < 0$

which implies correct classification, virtually no cost is incurred. On the other hand, a positive $d_k(\boldsymbol{x})$ leads to a penalty which becomes essentially a count of classification error if a zero-one cost function or any of the above smoothed zero-one functions is used. Finally, for any unknown $\boldsymbol{x}$, the classifier performance is measured by

$$\ell(\boldsymbol{x};\ \Lambda) = \sum_{k=1}^{M} \ell_k(\boldsymbol{x};\ \Lambda)\, 1(\boldsymbol{x} \in C_k) \tag{17}$$

where $1(\ )$ is an indicator function:

$$1(\mathfrak{A}) = \begin{cases} 1, & \text{if } \mathfrak{A} \text{ is true} \\ 0, & \text{otherwise} \end{cases} \tag{18}$$

and C_k is used to denote both the class and the data set.

This three-step definition emulates the classification operation and approximates the performance evaluation in terms of classification errors in a smooth functional form. To see how this formulation relates to the minimum classification error, let us assume the discriminant function is properly chosen to have the correct form as the true *a posteriori* probability $p_\Lambda(C_i|\boldsymbol{x})$, where the subscript Λ denotes the fact that the probabilities are defined by the parameter set Λ. The Bayes minimum risk (minimum classification error) resulting from the maximum *a posteriori* (MAP) rule can be written as

$$\mathcal{E} = \sum_{k=1}^{M} \int_{\mathfrak{X}_k} p_\Lambda(\boldsymbol{x},\ C_k)\, 1(\boldsymbol{x} \in C_k) d\boldsymbol{x}. \tag{19}$$

The integration in (19) is over part of the entire observation space $\mathfrak{X}$ that causes classification error according to the MAP rule, i.e.,

$$\mathfrak{X}_k = \{\boldsymbol{x} \in \mathfrak{X} \,|\, p_\Lambda(C_k|\boldsymbol{x}) \neq \max_i p_\Lambda(C_i|\boldsymbol{x})\}. \tag{20}$$

The classification error can be rewritten as

$$\begin{aligned} \mathcal{E} &= \sum_{k=1}^{M} \int_{\mathfrak{X}} p_\Lambda(\boldsymbol{x},\ C_k)\, 1(\boldsymbol{x} \in C_k) \\ &\qquad 1[p_\Lambda(C_k|\boldsymbol{x}) \neq \max_i p_\Lambda(C_i|\boldsymbol{x})]\, d\boldsymbol{x} \\ &\simeq \sum_{k=1}^{M} \int_{\mathfrak{X}} p_\Lambda(\boldsymbol{x},\ C_k)\, 1(\boldsymbol{x} \in C_k)\, \ell_k(d_k(\boldsymbol{x}))\, d\boldsymbol{x}. \end{aligned} \tag{21}$$

The approximation in (21) can be made arbitrarily close by varying the values of η and ζ or ξ. Note that even if the discriminant function differs from the true *a posteriori* probability, $\mathcal{E}$ of (21) still represents the classification error criterion conditioned on the choice of g, suitable for minimization via descent algorithms. The advantage of the formulation is immediately clear when the solution procedure employs gradient-type descent methods.

IV. Descent Methods

The cost function of (17) is defined for each input pattern $\boldsymbol{x}_i$. This cost function is the basis of the objective that we shall optimize with descent methods. Given a set of labeled training patterns $\mathfrak{L} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\}$, there are two ways of defining the performance objective; one is the empirical average cost and the other the expected cost. Although algorithmic difference between the two is minimal, optimization of these two conceptually different objectives leads to gradient search solutions with different convergence properties.

A. Empirical Average Cost and Gradient Descent Algorithm

Given a set of design observations $\mathfrak{L} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\}$, we can define an empirical average cost as

$$L_0(\Lambda) = \frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{M} \ell_k(\boldsymbol{x}_i;\ \Lambda)\, 1(\boldsymbol{x}_i \in C_k). \tag{22}$$

This well-defined cost function can be conveniently minimized by a gradient descent algorithm, using the following adaptation rule:

$$\Lambda_{t+1} = \Lambda_t - \epsilon \nabla L_0(\Lambda_t) \tag{23}$$

where Λ_t denotes the parameter set at the tth iteration. The usual care and considerations associated with the gradient descent algorithm, e.g., the choice of ϵ, of course apply here. Furthermore, the adaptation schedule can be defined arbitrarily. One extreme is that the classifier parameters are adjusted upon presentation of each training pattern $\boldsymbol{x}_i \in C_k$ and the gradient is taken as $\nabla \ell_k(\boldsymbol{x}_i;\ \Lambda)$ as the indicator function in (22) dictates. Another extreme is to adjust the parameters after the entire training set $\mathfrak{L}$ is classified. In this case, the gradient in (23) becomes proportional to the average gradient according to (22). The latter case is expected to produce a much smoother learning curve than the former case. Other adaptation "schedules" that operate sequentially on subsets of the design sample $\mathfrak{L}$ are obviously possible. Note that the minimization is only for the (approximate) classification cost incurred in classifying the *design sample* $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\}$, although we can use empirical arguments to infer that L_0 is asymptotically the expected performance as $N \to \infty$. This point is one of the fundamental differences between the new method and the classical distribution estimation approach to pattern recognition, in that the asymptotic results are with regard to the classification error instead of the distribution estimates.

B. Expected Cost and Probabilistic Descent Algorithm

The expected cost can be expressed as

$$L(\Lambda) = E\{\ell(\boldsymbol{x};\ \Lambda)\} = \sum_k P(C_k) \int \ell_k(\boldsymbol{x};\ \Lambda) p(\boldsymbol{x}|C_k)\, d\boldsymbol{x} \tag{24}$$

where $P(C_k)$ and $p(\boldsymbol{x}|C_k)$ are the class *a priori* and conditional probabilities, respectively. Obviously, the expectation operator in (24) indicates that the minimization is for the true expected error, not just the errors incurred for the finite design sample set $\mathfrak{L}$. However, since both the *a*

priori and conditional distributions are unknown, the expected cost cannot be directly minimized. Fortunately as suggested by the theorem below, we still can seek to minimize L by adaptively adjusting Λ in response to the incurred cost each time a training pattern $\boldsymbol{x}$ is presented. The adjustment of Λ is again according to

$$\Lambda_{t+1} = \Lambda_t + \delta\Lambda_t \tag{25}$$

where the "correction" term $\delta\Lambda_t$ is a function of the input pattern $\boldsymbol{x}$, its class label C_i, and the current parameter state Λ_t, i.e., $\delta\Lambda_t = \delta\Lambda(\boldsymbol{x}, C_k, \Lambda_t)$. The magnitude of the correction term is small such that the first-order approximation

$$L(\Lambda_{t+1}) \cong L(\Lambda_t) + \delta\Lambda_t \nabla L(\Lambda)|_{\Lambda=\Lambda_t} \tag{26}$$

holds. As will be shown below, this allows us to address $E[L(\Lambda_{t+1}) - L(\Lambda_t)] = E[\delta L(\Lambda_t)]$ directly rather than $L(\Lambda_{t+1}) - L(\Lambda_t) = \delta L(\Lambda_t)$. Note that

$$E[\delta L(\Lambda_t)] = E[\delta\Lambda(\boldsymbol{x}, C_k, \Lambda_t)]\nabla L(\Lambda_t). \tag{27}$$

Therefore, the goal is to find an adaptation rule such that $E[\delta L(\Lambda_t)] < 0$ and such that Λ_t converges at least to a locally optimum solution Λ^*. The probabilistic descent algorithm can be summarized in the following theorem [14], [15].

Probabilistic Descent Theorem: Given $\boldsymbol{x} \in C_k$ if the classifier parameter adjustment $\delta\Lambda(\boldsymbol{x}, C_k, \Lambda)$ is specified by

$$\delta\Lambda(\boldsymbol{x}, C_k, \Lambda) = -\epsilon \boldsymbol{U} \nabla \ell_k(\boldsymbol{x}; \Lambda) \tag{28}$$

where $\boldsymbol{U}$ is a positive-definite matrix and ϵ is a small positive real number, then

$$E[\delta L(\Lambda)] \leq 0. \tag{29}$$

Furthermore, if an infinite sequence of random observations $\boldsymbol{x}_t$ are presented for training and the parameter adjustment rule of (28) is utilized with a corresponding step size sequence ϵ_t which satisfies

$$\text{i)} \quad \sum_{t=1}^{\infty} \epsilon_t \to \infty \tag{30}$$

and

$$\text{ii)} \quad \sum_{t=1}^{\infty} \epsilon_t^2 < \infty \tag{31}$$

then the parameter sequence Λ_t according to

$$\Lambda_{t+1} = \Lambda_t + \delta\Lambda(\boldsymbol{x}_t, C_k, \Lambda_t) \tag{32}$$

converges with probability one to a Λ^* which results in a local minimum of $L(\Lambda)$.

As can be seen, the difference between the two adaptation rules (23) and (28) is minimal, but the probabilistic descent algorithms provides what can be considered the basis of adaptive learning in which as more data are presented, the classifier is further refined in the sense of reducing the expected misclassification cost. The empirical average cost and the associated gradient descent algorithm, on the other hand, are important for pragmatic reasons. For example, when the *a posteriori* probabilities are used in the discriminant function, and thus in the class cost function ℓ_k, the expected cost of (24) becomes unwieldy for optimization because the expectation would also be a function of the classifier parameters, i.e., $p(\boldsymbol{x}, C_k) = p_\Lambda(\boldsymbol{x}, C_k)$ in (24). This particular difficulty is avoided in the empirical average cost of (22).

V. Minimum Classification Error Multilayer Feedforward Networks

The above minimum classification error formulation can be applied to many new classifier structures such as the multilayer perceptron (MLP) [8], learning vector quantizer (LVQ) [7], and distance network [16]. The formulation has a profound effect on these new classifier structures and leads to improved learning rules for a better pattern recognition performance. We discuss in this section how the new minimum classification error criterion can be incorporated in a multilayer perceptron. The relationship among these new classifiers under the common criterion of minimum misclassification probability will be addressed in a separate paper [17].

A multilayer perceptron is a feedforward network, as illustrated in Fig. 1 for a two-layer perceptron (or three-layer if the input layer is also counted as one), that has been widely considered in pattern recognition applications. Let m be the total number of layers, n_j the number of nodes in the jth layer ($n_m = M$) and z_{ij} the activation output of the ith node in the jth layer, with $\boldsymbol{x}^* = (x_1, x_2, \cdots, x_K) = (z_{10}, z_{20}, \cdots, z_{K0}) = \boldsymbol{z}_0^*$ being the input. The activation output z_{ij} is obtained according to

$$z_{ij} = f(\boldsymbol{w}_{ij}^* \boldsymbol{z}_{j-1} + w_{0ij}) \tag{33}$$

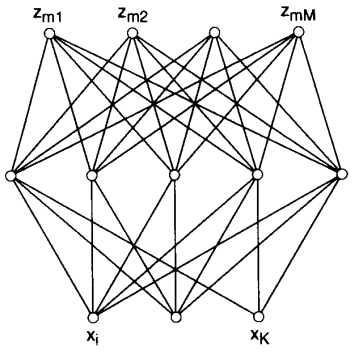
where $\boldsymbol{w}_{ij}^* = (w_{1ij}, w_{2ij}, \cdots, w_{n_{j-1}ij})$ is the weight vector connecting the n_{j-1} nodes in the $(j-1)$th layer to the ith node in the jth layer and f is the activation function, an example of which is the sigmoid function of (16). Without loss of generality, we shall neglect the bias term and denote $y_{ij} = \boldsymbol{w}_{ij}^* \boldsymbol{z}_{j-1}$ such that $z_{ij} = f(y_{ij})$. The set of weights $\boldsymbol{W} = \{\boldsymbol{w}_{ij}\}$ and the prescribed activation function thus define an MLP classifier.

To train an MLP classifier, one employs the error back-propagation (EBP) algorithm [8] which is a supervised learning scheme based on a training vector $\boldsymbol{x}$ and its corresponding target (or teaching) vector $\boldsymbol{t}$. The target vector $\boldsymbol{t}^* = (t_1, t_2, \cdots, t_M)$ associated with a given training vector $\boldsymbol{x} \in C_i$ for an M-class classification task typically is binary valued with

$$t_j = \begin{cases} 1, & j = i \\ 0, & \text{otherwise.} \end{cases} \tag{34}$$

The classifier is so trained as to reduce the difference between the output vector $\boldsymbol{z}_m$ and the target vector $\boldsymbol{t}$. The sum-squared error function

$$E_{se} = \sum_{i=1}^{M} (t_i - z_{im})^2 \tag{35}$$



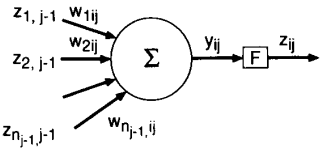


Fig. 1. A three-layer perceptron.

is often chosen as the cost function for optimization.

The error back-propagation algorithm is a gradient descent algorithm that adjusts the weights to minimize E_{se} (or $\Sigma_{\mathcal{L}} E_{se}$) according to

$$\boldsymbol{W}^{(\tau+1)} = \boldsymbol{W}^{(\tau)} - \mu \nabla_{\boldsymbol{W}} E_{se}\big|_{\boldsymbol{W}=\boldsymbol{W}^{(\tau)}} \tag{36}$$

where superscript τ indicates the time instance when the weight values are recorded/adjusted. More specifically, the adjustment for $w_{kij}^{(\tau)}$, $\Delta w_{kij}^{(\tau)}$, is

$$\Delta w_{kij}^{(\tau)} = -\mu \frac{\partial E_{se}}{\partial w_{kij}} = -\mu h_{ij} z_{k,j-1} \tag{37}$$

where μ is a positive number,

$$h_{ij} = 2(z_{ij} - t_i) f'(y_{ij}), \qquad \text{for } j = m \tag{38}$$

and

$$h_{ij} = \left(\sum_{k=1}^{n_j - 1} w_{i,k,j+1} h_{k,j+1} \right) f'(y_{ij}), \qquad \text{for } j = 1, 2, \cdots, m-1. \tag{39}$$

It is important to note that use of a target vector as defined by (34), or variations thereof, is required for the formulation of a sum-squared error function for optimization by descent methods. It, however, does not necessarily lead to minimum classification error; that is, the solution $\boldsymbol{W}$ that minimizes E_{se}, or expectation of E_{se}, may not coincide with the solution that minimizes the misclassification probability, as argued previously. One way to make the error back-propagation algorithm consistent with the minimum classification error objective is to use the cost of (21) in lieu of E_{se}. In particular, following the above three-step formulation procedure, we propose, for a given training sample $\boldsymbol{x} \in C_i$, to use

$$E = \ell_i(d_i(\boldsymbol{x})) \tag{40}$$

where ℓ_i is defined by (14)–(16), $d_i(\boldsymbol{x})$ is expressed in terms of $\boldsymbol{y}_m$ as

$$d_i(\boldsymbol{x}) = -y_{im} + \left[\frac{1}{M-1} \sum_{j, j \neq i} y_{jm}^{\eta} \right]^{1/\eta} \tag{41}$$

and η is a large positive number. The usual nonlinearity in the final layer is no longer strictly necessary. (In case y_{jm}'s need to maintain nonnegativeness, a supplementary nonlinearity such as $e^{y_{jm}}$ can be used.) Without nonlinearity, the chain rule that led to the error back-propagation formula of (37)–(39) remain the same, i.e.,

$$\begin{aligned} \Delta w_{kij} = -\mu \frac{\partial E}{\partial w_{kij}} = \sum_{i_m=1}^{n_m} \frac{\partial E}{\partial y_{i_m,m}} \\ \cdot \sum_{i_{m-1}=1}^{n_{m-1}} \frac{\partial y_{i_m,m}}{\partial z_{i_{m-1},m-1}} \frac{\partial z_{i_{m-1},m-1}}{\partial y_{i_{m-1},m-1}} \cdots \\ \sum_{i_{j+1}=1}^{n_{j+1}} \frac{\partial y_{i_j+2,j+2}}{\partial z_{i_{j+1},j+1}} \cdot \frac{\partial y_{i_{j+1},j+1}}{\partial z_{ij}} \frac{\partial z_{ij}}{\partial w_{kij}} \end{aligned} \tag{42}$$

where the recursion of $\Sigma_i (\partial y_{i,j+1}/\partial z_{ij})(\partial z_{ij}/\partial y_{ij})$ for $j = M-1, M-2, \cdots$ is obvious as in (39). The only difference is in the first derivative of E with respect to z_{km}. Specifically,

$$\frac{\partial E}{\partial y_{km}} = \ell_i'(d_i) \frac{\partial d_i}{\partial y_{km}} \tag{43}$$

and

$$\frac{\partial d_i}{\partial y_{km}} = \begin{cases} -1, & k = i \\ \dfrac{y_{km}^{\eta-1}}{M-1} \left\{ \dfrac{1}{M-1} \displaystyle\sum_{j, j \neq i} y_{jm}^{\eta} \right\}^{-1+1/\eta}, & k \neq i, \end{cases} \tag{44}$$

according to the definition of $d_i(\boldsymbol{x})$ in (41). Note that for large η,

$$\frac{\partial d_i}{\partial y_{km}} \cong y_{km}^{\eta-1} \left\{ \sum_{j, j \neq i} y_{jm}^{\eta} \right\}^{-1} \qquad \text{for } k \neq i. \tag{45}$$

Therefore, with a modification in the error criterion, a multilayer perceptron can be trained for minimum classification error.

An advantage with the modified multilayer perceptron is related to the consistency issue raised in the Introduction. In the distribution estimation approach to the classification problem, optimality of the solution cannot be addressed when the form of the data distribution is unknown and, as usually is the case, incorrectly assumed. Similarly, in the discriminant based approach, the choice of the discriminant function is crucial to the classifier performance, even though the optimization objective has been correctly defined as the minimum misclassification probability. It should be understood that our definition of minimum misclassification probability has two meanings: One that is conditioned on the prescribed choice of the

discriminant function, g_i, and the other that is the absolute minimum Bayes risk which occurs only when full knowledge of the *a posteriori* probabilities is correctly used in classification. These two meanings become identical when the approximation in (21) becomes exact, a case requiring that the loss function $\ell(d_k)$ approach the error counts, $1(P_\Lambda(C_k|\boldsymbol{x}) \neq \max_i P_\Lambda(C_i|\boldsymbol{x}))$. Since a multilayer perceptron has been shown to be able to define an arbitrary real function [18], with a flexible architecture in terms of the number of hidden units, it thus offers the potential of automatically converging to the true minimum Bayes risk, particularly when the training objective is correctly chosen.

VI. Classification Experiment

Several classification experiments were conducted to study the characteristics of the minimum classification error criterion, and the difference in classification performances as compared to traditional criteria as well as classifier architectures. We report two sets of experimental results here: one involves a set of artificially generated data with mixture distributions and the other pertains to the Fisher's iris data, well known to the pattern recognition community.

The artificially generated data set consists of two classes. The distributions are of Gaussian mixture types with two components in each class. Each token is of two dimensions. For class 1, 700 tokens with mean $(-5.\ 0.)^t$ and covariance matrix

$$\begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{pmatrix}$$

and 300 tokens with mean $(0.\ 0.)^t$ and covariance

$$\begin{pmatrix} 0.25 & 0.0 \\ 0.0 & 0.01 \end{pmatrix}$$

were generated. Similarly, for class 2, we generated 500 tokens with mean $(1.0\ 5.0)^t$ and covariance

$$\begin{pmatrix} 0.1 & 0.0 \\ 0.0 & 1.0 \end{pmatrix}$$

and another 500 tokens with mean $(1.0\ 0.0)^t$ and covariance

$$\begin{pmatrix} 1.00.0 \\ 0.01.0 \end{pmatrix}.$$

Fig. 2 shows a scatter plot of the 2000 training tokens generated for the two classes. Note that for the two-class case, the misclassification measure of (12) reduces to only two terms, a situation similar to Amari's formulation.

These training tokens (2000 in total) were used to train three linear classifiers (LC) with three different criterion functions: the perceptron criterion of (5), the minimum squared error criterion of (8) and the minimum classification error of (12), (14), and (16). For brevity, we denote these criterion functions by PE (perceptron error), MSE (minimum squared error), and MCE (minimum classification error), respectively. We further generated

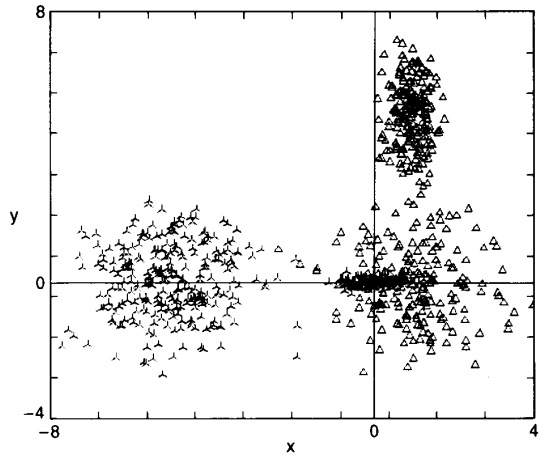


Fig. 2. A scatter plot of 2000 two-dimensional tokens generated by two-component mixture distribution sources; class 1 with mean $(-5\ 0)^t$, covariance $\begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{pmatrix}$, mixture weight 0.7 and mean $(0\ 0)^t$, covariance $\begin{pmatrix} 0.25 & 0.0 \\ 0.0 & 0.01 \end{pmatrix}$, mixture weight 0.3; class 2 with mean $(1.0\ 5.0)^t$, covariance $\begin{pmatrix} 0.1 & 0.0 \\ 0.0 & 1.0 \end{pmatrix}$, mixture weight 0.5 and mean $(1.0\ 0.0)^t$, covariance $\begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{pmatrix}$, mixture weight 0.5.

an independent set of 2000 tokens according to the same distributions for testing purposes. The classification results in terms of the recognition error (averaged over four independent runs) for both the training set and the test set are listed in Table I for comparison. We have run similar experiments on a number of artificially generated data sets with various distributions. The superior performance of LC + MCE was consistently observed in the experiments. The classification performances of LC + PE and LC + MSE, while being consistently inferior to that of LC + MCE, did not show a conclusive pattern in terms of the relative merit between the two. (In the table, the results pertaining to the independent test set are slightly better than that of the training set. This can be attributed to the fact that the training data and the test data are generated according to the same, simple distributions, ensuring data consistency.)

It is probably more important to examine how the error (or loss) function is minimized in relation to the classification error rate in the present, well-controlled case than a simple comparison on the final classification performance. We plot in Figs. 3(a)–(c) the learning (error minimization) curves of the four experiment runs for the three error criteria respectively. These curves were obtained for the training data but the curves for the test data are essentially the same. In each figure, we show two sets of plots: the upper part displays the criterion function as it is being minimized in terms of data epochs (an epoch represents a completion of processing on the entire training data set) and the lower part is the recognition rate evaluated at each training epoch. Of particular significance is Fig. 3(c) for

TABLE I
CLASSIFICATION PERFORMANCE COMPARISON FOR 3 LINEAR CLASSIFIER DESIGN CRITERIA USING 2-CLASS 2-COMPONENT MIXTURE DISTRIBUTION DATA

Recognition Error (%)	LC + PE	LC + MSE	LC + MCE
Training set	12.24	15.25	10.00
Test set	11.18	14.91	9.85

the MCE case. It is clearly seen that the minimized error criterion closely approximates the actual classification error rate, thus accomplishing the original objective of minimizing the classification error. This phenomenon is not observed in either the PE case or the MSE case. For example, in Fig. 3(a), the perceptron error is fluctuating near 0 while the classification rate is varying around 80%.

The other set of experiments involves Fisher's iris data. The iris data consists of four measurements made by E. Anderson on 150 samples of three species of iris. The four measurements are the calyx length, the calyx width, the petal length and the petal width. Fifty tokens are available for each of the three species. The task is to classify these measured tokens into the three individual species. (This three-class problem thus has an increased sophistication in the misclassification measure, compared to the above two-class problem.) Fisher used the data in his classic paper on discriminant analysis [10]. Many clustering experiments were studied in the past using this data set. In our current experiment, all the tokens were used for training the classifiers and the classification results in the following are restricted to the training data. This thus allows us to compare different error criteria in a well-defined classifier setup. (This is a scenario where one is interested in *classification* of the given data rather than *recognition* of future data.)

We investigated two types of classifiers for the classification task. One, similar to the above mixture data case, is a linear classifier according to (3) with the corresponding decision rule of (4). The other is a three-layer perceptron according to (33). For the linear classifier case, we again investigated the three error criterion functions, PE, MSE, and MCE.

The three-layer perceptron structure requires further explanation as it involves nonlinearity. The sum-squared error function of (35) with nonlinearity at the output layer is the prevalent choice in most of the traditional MLP classifiers. The minimum classification error criterion as defined in (40) and (41) does not require a nonlinearity at the output. These two criteria can thus be directly compared based on the same feedforward structure. To provide additional insights,we also implemented a three-layer perceptron using the sum-squared error function but without the nonlinearity at the output layer. Since the error back-propagation algorithm is also to minimize the sum-squared error, we again use MSE to designate the traditional MLP training objective. Therefore, these three cases are denoted by 3PNET + MSE (the case without nonlinearity), 3PNET + MSE + N (the traditional case with nonlinearity) and 3PNET + MCE (the proposed minimum classification error modification) respectively.

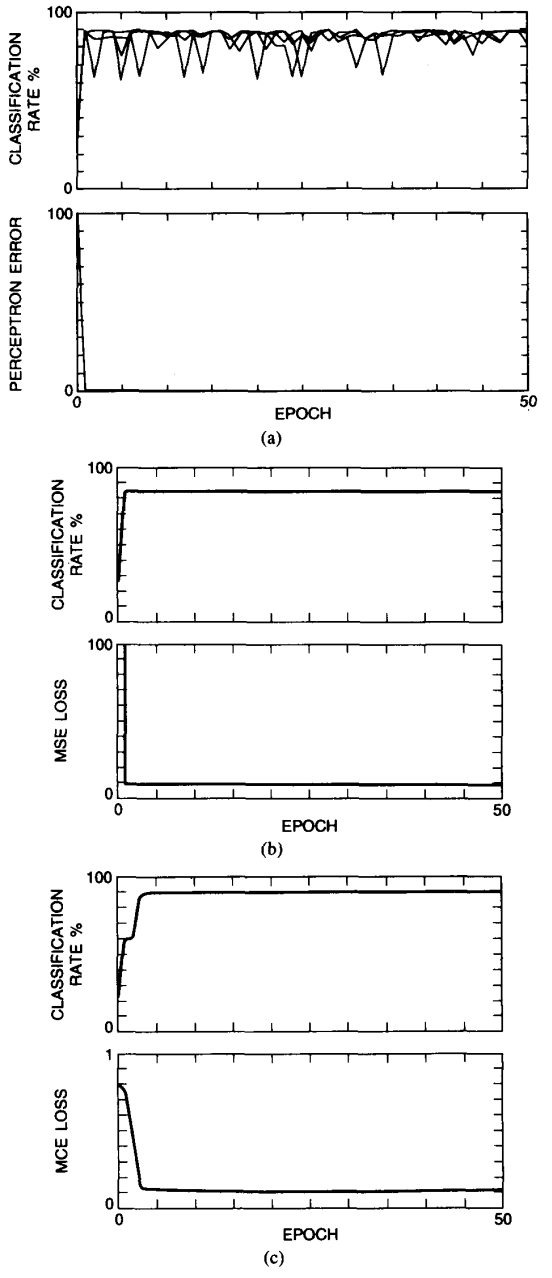


Fig. 3. (a) Learning curves in terms of training epochs: upper curve, recognition rate (% correct) for four experiment runs; lower curve, PE criterion. (b) Learning curves in terms of training epochs: upper curve, recognition rate (% correct) for four experiment runs; lower curve, MSE criterion. (c) Learning curves in terms of training epochs: upper curve, recognition rate (% correct) for four experiment runs; lower curve, MCE criterion.

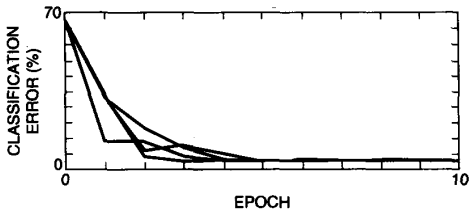


Fig. 4. Learning curves in terms of the classification error rate as a function of training epochs for a linear classifier based on the minimum classification error criterion designed for the iris data set. (Four runs.)

TABLE II
CLASSIFICATION PERFORMANCE COMPARISON FOR 3 LINEAR CLASSIFIER DESIGN CRITERIA USING FISHER'S IRIS DATA

	LC + PE	LC + MSE	LC + MCE
Classification error (%)	14	15.1	4

TABLE III
CLASSIFICATION PERFORMANCE COMPARISON FOR 3 FEEDFORWARD NETWORK CLASSIFIER DESIGN CRITERIA USING FISHER'S IRIS DATA

	3PNET + MSE	3PNET + MSE + N	3PNET + MCE
Classification error (%)	19.8	12.3	2.2

To account for the intrinsic characteristics of stochastic training, we performed four runs of classification experiments for each classifier training, using different orders of training data presentation. Fig. 4 shows the four learning curves in terms of the classification error rate for the LC + MCE case. The learning behavior for the MCE criterion is seen to be quite steady and effective. The average classification error rate is 4% (i.e., 96% correct) when MCE was used as the criterion. Table II lists the average error rates of the four runs for the three linear classifier cases, PE, MSE, and MCE, respectively for performance comparison. Obviously, the MCE criterion led to a performance far better than the other two criterion functions.

With three-layer feedforward networks or MLP's, the classification performance in general improves. The only exception is the case of 3PNET + MSE without a nonlinearity at the output layer. We list again the average error rates of the four experiment runs in Table III for the three 3PNET cases, MSE, MSE + N and MCE, respectively, for comparison. The multiple layer structure and the nonlinearity indeed led to performance improvements over the linear classifier case. The three-layer network trained by the MCE criterion gave again the best result of 2.2% error rate. The performance advantage of MCE is clearly seen.

VII. SPEECH RECOGNITION EXPERIMENTS

Another set of experiments were conducted to examine the characteristic differences between the traditional minimum sum-squared error and the minimum classification error when applied to a layered perceptron structure for speech recognition. The experiments involved recognition of the highly confusible English E-set alphabet, namely, b, c, d, e, g, p, t, v, and z. The speech signal was recorded from 100 native Americans, including 50 male and 50 female, through local dial-up telephone lines. The sampling rate was 6.67 kHz and the bandwidth of the antialiasing filter was from 100 to 3200 Hz for a digital implementation of analysis processing. Each talker spoke each word twice, producing two sets of data bases. One was used as the training set and the other as the test set. (Other parameters include: a 300-sample analysis window, a 200-sample overlap between adjacent analysis windows, an eighth-order linear prediction analysis and a 24 cepstral coefficient (cepstrum and delta cepstrum) representation [19], [20].)

To normalize the speaking rate variation inherent in the spoken utterances, a conventional speech recognizer with dynamic time warping (DTW) was used as the baseline classifier. Fig. 5 shows a block diagram of the conventional DTW recognizer. The unknown input utterance is first analyzed and then matched to each of the reference templates, resulting in sequences of spectral distortions. Let us denote these distortion sequences by $D_j = \{d_j(i), i = 1, 2, \cdots, m_j\}$ where j is the template index and m_j is the length of the jth template. Each vocabulary word may be represented by a multiplicity of reference templates although in this paper we report only the simplest case where we use one reference template for each word. These distortion sequences $\{D_j\}, j = 1, 2, \cdots, M$ ($M = 9$ in the present case), are thus the input to the classifier to be designed. Note that the total dimension of the input is $M_T = \sum_{j=1}^{M} m_j$.

Traditionally, the classification decision is made based on a simple average distortion

$$y_j = \sum_{i=1}^{m_j} \frac{1}{m_j} d_j(i). \tag{46}$$

The recognized word k is the one that satisfies

$$y_k = \min_{1 \le j \le M} y_j. \tag{47}$$

A more general discriminant function of the distortion sequences is thus

$$y_j = \sum_{i=1}^{m_j} w_{ji} d_j(i) + w_{j0}. \tag{48}$$

As in most linear discriminant cases, the function of (48) can be implemented in a particular perceptron structure as shown in Fig. 6. An expanded architecture of the scaled perceptron of Fig. 6 is of course the original, unscaled, fully connected perceptron, taking the entire set of distortion sequences $\{D_j\}$ as the input. The output value before nonlinearity operation in this case is

$$y_j = \sum_{i=1}^{M_T} w_{ji} d(i) + w_{j0} \tag{49}$$

where $\{d(i)\}$, with index $i = 1$ to M_T, is the concatenated distortion sequence of $\{D_j\}, j = 1, 2, \cdots, M$.

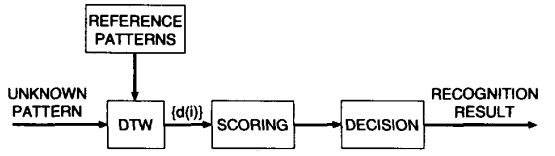


Fig. 5. A block diagram of the conventional dynamic time warping based speech recognition system.

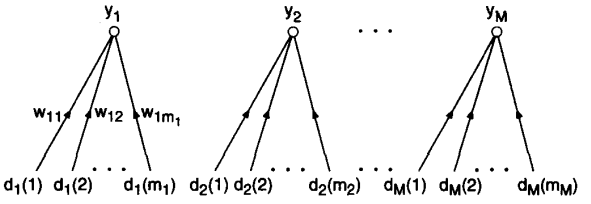


Fig. 6. A scaled, simplified perceptron with pruned connections.

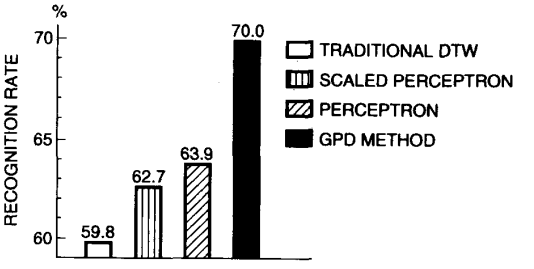


Fig. 7. Performance comparison (% correct) of several training methods for speaker-independent recognition of the English E-set vocabulary.

The weight functions in these perceptron structures can be trained, as explained in the previous sections, by the error back-propagation algorithm [8] with the usual sum-squared error criterion (37)–(39), or by the generalized descent algorithm with a minimum classification error criterion, (40)–(45). The nonlinearity in each node is identically the sigmoid function $1/(1 + e^{-y})$. Note that the error back-propagation algorithm for training works the same way in the scaled perceptron and the fully connected network. The only difference is that the broken connections in the scaled perceptron represent a constant zero weight in the fully connected network.

The recognition rates (% correct) pertaining to these various methods are summarized in Fig. 7 with reference to the traditional simple uniform weighting method (46). The traditional method yielded a result of 59.8% accuracy rate which was improved to 62.7% and 63.9% by the scaled perceptron and the fully connected perceptron with the sum-squared error criterion, respectively. The minimum classification error criterion, optimized by the generalized descent algorithm, achieved a recognition rate of 70.0%, the highest among all the competing methods.

Besides demonstrating the effectiveness of the minimum classification error formulation, the above experiment is of interest from a very specific point of view. This pertains to the classifier design problem where the inputs are of different dimensions and thus the classical vector space approach to classifier design may not be directly applicable. This particular problem is addressed in [15].

VIII. Summary

In this paper, we present a new formulation of the pattern recognition problem, aiming at achieving a minimum error rate classification. The classical discriminant analysis methodology is blended with the classification rule (traditionally expressed in an operational form) in a new functional form and is used as the design objective criterion to be optimized by numerical search algorithms. The new formulation results in a smooth error function which approximates the empirical error rate for the design sample set arbitrarily closely. We have applied the minimum error formulation to several recognition tasks and demonstrated the advantages of the new method. The minimum classification error formulation can also be incorporated in new classifier structures such as the multi-layer perceptron. We further suggest how the error back-propagation algorithm can work with the new error criterion and achieve the minimum error result. In a speech recognition experiment involving the English E-set vocabulary, it was demonstrated that the new minimum error method achieves the best recognition performance. The proposed learning method and formulation provides a solid analytical ground for the long-standing minimum error classifier design problem.

Acknowledgment

The authors would like to thank P. C. Chang of Telecommunication Laboratories, Taiwan, for conducting the speech recognition experiment reported in Section VII.

References

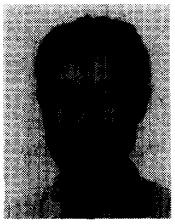
[1] R. O. Duda and Peter E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.
[2] W. H. Highleyman, "Linear decision functions with application to pattern recognition," *Proc. IRE*, vol. 50, pp. 1501–1514, June 1962.
[3] F. Rosenblatt, "The perceptron—a perceiving and recognizing automation," Rep. 85-460-1, Cornell Aeronautical Lab., Ithaca, NY, Jan. 1957.
[4] N. J. Nilsson, *Learning Machines: Foundations of Trainable Pattern-Classifying Systems*. New York: McGraw-Hill, 1965.
[5] J. K. Hawkins, "Self-organizing systems—a review and commentary," *Proc. IRE*, vol. 49, pp. 31–48, Jan. 1961.
[6] R. P. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Mag.*, pp. 4–22, Apr. 1987.
[7] T. Kohonen, G. Barna, and R. Chrisley, "Statistical pattern recognition with neural networks: Benchmarking studies," in *IEEE Proc. ICNN*, vol. 1, July 1988, pp. 61–68.
[8] D. Rumelhart, E. Hinton, and J. Williams, "Learning internal representation by error propagation," in *Parallel Distributed Processing*, vol. 1, Rumelhart and McClelland, Eds. Cambridge, MA: M.I.T. Press, 1986, pp. 318–364.
[9] W. Chou and B. H. Juang, "Adaptive discriminative learning in pattern recognition," in preparation.
[10] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, part II, vol. 7, pp. 179–188, 1936.
[11] S. Agmon, "The relaxation method for linear inequalities," *Canadian J. Math.*, vol. 6, pp. 382–392, 1954.
[12] Y.-C. Ho and R. L. Kashyap, "An algorithm for linear inequalities and its applications," *IEEE Trans. Elec. Comput.*, vol. EC-14, pp. 683–688, Oct. 1965.

[13] J. D. Patterson and B. F. Womack, "An adaptive pattern classification system," *IEEE Trans. Syst., Sci., Cybern.*, vol. SSC-2, pp. 62–67, Aug. 1966.
[14] S. Amari, "A theory of adaptive pattern classifiers," *IEEE Trans. Elec. Comput.*, vol. EC-16, pp. 299–307, June 1967.
[15] S. Katagiri, C. H. Lee, and B. H. Juang, "New discriminative training algorithm based on the generalized probabilistic descent method," in *Proc. 1991 IEEE Workshop Neural Networks for Signal Processing*, Piscataway, NJ, Aug. 1991, pp. 299–308.
[16] S. Katagiri, "Systematic explanation of learning vector quantization and multilayer perceptron—proposition of distance network," *IEICE*, MBE 88-72, Oct. 1988 (in Japanese).
[17] S. Katagiri, C. H. Lee, and B. H. Juang, "Discriminative multilayer feedforward networks," in *Proc. 1991 IEEE Workshop Neural Networks for Signal Processing*, Piscataway, NJ, Aug. 1991, pp. 11–20.
[18] K. Funahashi, "On the approximate realization of continuous mapping by neural networks," *Neural Networks*, vol. 2, pp. 183–192, 1989.
[19] B. H. Juang, L. R. Rabiner, and J. G. Wilpon, "On the use of bandpass liftering in speech recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-37, no. 7, pp. 947–954, July 1987.
[20] F. K. Soong and A. E. Rosenberg, "On the use of instantaneous and transitional spectral information in speaker recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-36, no. 6, pp. 871–879, June 1988.

Biing-Hwang Juang (S'79–M'80–SM'87–F'92) received the B.Sc. degree in electrical engineering from National Taiwan University, Taipei, in 1973 and the M.Sc. and Ph.D. degrees in electrical and computer engineering from the University of California, Santa Barbara, in 1979 and 1981, respectively.

In 1978, he did research on vocal tract modeling at the Speech Communications Research Laboratory (SCRL). He then joined Signal Technology, Inc., in 1979 as Research Scientist, working on signal and speech related topics. Since 1982, he has been with AT&T Bell Laboratories where he is engaged in a wide range of speech related research activities. He has published extensively in the area of speech communication and holds two sets of patents.

Dr. Juang has served on several IEEE Technical Committees and chaired IEEE Workshops in the past. He was Associate Editor for the IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING from 1986 to 1988. He currently chairs the Technical Committee on Neural Networks for Signal Processing in the IEEE Signal Processing Society. He also serves on several international advisory boards outside the United States and is Associate Editor of the *Journal of Speech Communication*.



Shigeru Katagiri (M'88) was born in Japan on November 3, 1953. He received the B.E. degree in electrical engineering and the M.E. and Dr. Eng. degrees in information engineering from Tohoku University, Sendai, Japan, in 1977, 1979, and 1982, respectively.

From 1982 to 1986, he worked at the Electrical Communication Laboratories, Nippon Telegraph and Telephone Public Corporation, Tokyo, Japan, where he was engaged in speech recognition research. Since 1986, he has been with the ATR Auditory and Visual Perception Research Laboratories, Kyoto, Japan, where he is currently a Senior Researcher in the Department of Hearing and Speech Perception. During 1989–1990, he was a Visiting Researcher with the Speech Research Department, AT&T Bell Laboratories. He received the 22nd and 27th Sato Paper Awards of the Acoustical Society of Japan, in 1982 and 1987, respectively. His current research interests include studies on discriminative training, the learning capability of artificial neural networks, and speech recognition.

Dr. Katagiri is a member of the IEEE Signal Processing Society, the Acoustical Society of America, the Acoustical Society of Japan, the Institute of Electronics, Information, and Communication Engineers, the Japanese Cognitive Science Society, and the Japanese Society for Artificial Intelligence.