Improving the Generalization Ability of MCE/GPD Learning and its Application to Multistage Building Learning

Jun Rokui
Department of the Interdisciplinary Faculty of
Science and Engineering, Shimane University.
1060 Nishikawatsu-cho, Matsue-shi, 690-8504 Japan, addresses
e-mail - rokui@cis.shimane-u.ac.jp

Abstract

The Minimum Classification Error (MCE) / Generalized Probabilistic Descent (GPD) learning proposed by Katagiri and Juang in 1992 has attracted a great deal of attention for its high recognition performance and wide range of applications including the case where the length of feature vectors is variable like speech recognition. In this report, we propose a new method to improve the generalization performance of the MCE learning by employing an regularization technique which is widely used to solve ill-posed problems. Feed-forward neural networks are employed to evaluate the performance of the proposed method.

Keywords : MCE/GPD, Generalization Ability, ill-posed problem, over-fitting.

1 Introduction

In the classical pattern recognition theory, if one can predict the exact probability densities of the target categories beforehand, the Bayes decision rule would give the optimum decision that achieves the minimum error risk. In order to estimate the densities and design a set of classifiers, the maximum-likelihood estimation (ML) is widely used in various areas of machine learning and pattern recognition. However, in the real-world pattern recognition problems, the number of data available is restricted and therefore the Bayes-type classifiers trained by the ML method sometimes result in insufficient recognition performance. Instead of estimating the probabilistic distributions, one can employ the discriminative learning in which the parameters of the classifier are adapted to minimize the classification error.

However, it is for the very limited situations of real-world problems that the classical discriminative learning such as perceptron gives better classification performance than the ML-based learning. This comes from the fact that the cost function employed in the learning scheme is not differentiable in respect to the parameters that are to be adapted, and therefore parameter adaptation can not be done adequately.

In order to overcome the difficultly, Amari proposed Probabilistic Descent (PD). In 1992, Katagiri and Juang generalized the idea of the PD learning and proposed the Minimum Classification Error (MCE) / Generalized Probabilistic Descent (GPD) learning[1]. The MCE/GPD learning successfully defines an object function that can be optimized by means of the gradient descent technique. The key idea of the MCE formulation is to employ a smooth loss function which represents the classification error instead of using a hard decision function.

As a result, compared to other discriminative learning, the MCE/GPD learning is crucial in the sense that it is applicable to arbitrary discriminant functions that are differentiable in respect to the parameters that are to be adapted. To be specific, it can be applied to discriminant functions that deal with variable record length of data like speech recognition.

The superiority of the MCE learning to the conventional ML-based learning has been shown for various functions such as linear-discriminant functions, multi-layer perceptron (MLP), dynamic time warping (DTW) and hidden Markov model (HMM). However, it suffers from a problem of generalization performance for testing data as it is with other learning methods. This is due to the fact that the MCE learning has an inclination to adapt the parameters specifically to the training data in order to achieve the minimum classification error.

In this paper, we propose a new approach of improving the generalization performance of the MCE learning and to use the information of mis classified data by incorporating a new feature value derived from the misclassification measure of MCE/GPD learning.

The proposed learning scheme easily combines MCE/GPD learning with other different learning methods, which are expected to work supplementaly.

2 Minimum Classification Error Learning

Let $g_k(\mathbf{x}; \Lambda_k)$ be a discriminant function with positive value to discriminate a data of class C_k from the other classes, where $\mathbf{x}$, Λ_k denote a vector in D-dimensional feature space and the set of parameter of the discriminant function, respectively. For an input vector $\mathbf{x}$, if the following equation holds

$$g_k(\mathbf{x}; \Lambda_k) \geq g_i(\mathbf{x}; \Lambda_i) \text{ for all } i \neq k, \tag{1}$$

x is classified to class C_k.

In the framework of the MCE learning, misclassification measure for class C_k is defined as

$$d_k(\mathbf{x}) = -g_k(\mathbf{x}; \Lambda_k) + \left[\frac{1}{M-1} \sum_{j, j \neq k} g_j(\mathbf{x}; \Lambda_k)^\eta \right]^{1/\eta} \tag{2}$$

where M represents the number of classes and η is a positive number. In an extreme case where η goes to infinity, the misclassification measure becomes

$$d_k(\mathbf{x}) = -g_k(\mathbf{x}; \Lambda_k) + g_s(\mathbf{x}; \Lambda_s). \tag{3}$$

Here, s stands for the class number with the largest discriminant value among the rest of $M-1$ classes. Obviously $d_k(\mathbf{x}) \leq 0$ in case of correct classification, $d_k(\mathbf{x}) > 0$ in case of misclassification.

Using the misclassification measure for a set of training data $X = (\mathbf{x}_1, \cdots, \mathbf{x}_N)$, the objective function to be minimized is defined as an empirical average loss function given by

$$L_0(\Lambda|X) = \frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{M} \ell(d_k(\mathbf{x}_i)) 1(\mathbf{x}_i \in C_k). \tag{4}$$

Here, $\Lambda = (\Lambda_1, \cdots, \Lambda_M)$,

$$\ell(d) = \frac{1}{1 + e^{-\xi(d+\theta)}}, \tag{5}$$

and $1()$ of (4) is an indicator function which has a value of one when the argument is true and zero otherwise.

In order to minimize the objective function of (4), the well-known gradient descent method can be applied and the set of parameters of each discriminant function is adapted by the following rule:

$$\Lambda^{(t+1)} = \Lambda^{(t)} - \varepsilon \nabla L_0(\Lambda^{(t)}|X) \tag{6}$$

where $\Lambda^{(t)}$ denotes the set of parameters at the t-th iteration and ε denotes the learning parameter of a positive small value.

3 Improvement of the Generalization Performance

The MCE learning using the object function given in (4) tries to minimize the misclassification rate for the finite number of training data. As a result, the set of parameters Λ of the discriminant function specifically adapted to the training data. In consequence, this causes a decline of the generalization performance.

In order to improve the generalization performance of the MCE learning, the parameter ξ of (5) is expected to control the sensitivity of forming the decision boundary against the distribution of training data. However, the relationship between ξ and the shape of decision boundary in the feature space is not clear.

From the view point of generalization for real-world problems, the function that the recognizer tries to learn should be, in some sense, smooth. Based on this assumption as an a priori knowledge, we propose a new approach to improve the generalization performance by employing a regularization technique to the MCE learning. In the framework of regularization, the new objective function $L(\Lambda)$ can be defined as

$$L(\Lambda|X) = L_0(\Lambda|X) + \gamma F(\Lambda) \tag{7}$$

where F is the penalty term, and the parameter γ controls the extent to which the penalty term F influences the form of the solution.

Tikhonov proposed the class of Tikhonov regularizers to solve ill-posed problems, whose form is given by

$$F = \frac{1}{2}\sum_{r=0}^{R}\int_a^b h_r(x)\left(\frac{\partial^r y}{\partial x^r}\right)^2 \partial x. \tag{8}$$

Here, x, y denote input variable and output variable, respectively, and $h_r(x) \geq 0$ for $r = 0, \cdots, R-1$ and $h_R(x) > 0$. In the present study, as a simple case of the Tikhonov regularizer, we have employed the following empirical term given in, which is

$$F(\Lambda|X) = \frac{1}{2M}\sum_{k=1}^{M}\sum_{n=1}^{N}\sum_{i=1}^{D}\left(\frac{\partial^2 g_k(\mathbf{x}_n)}{\partial x_{ni}^2}\right)^2 \tag{9}$$

where $\mathbf{x}_n = (x_{n1}, \cdots, x_{nD})$ represents the n-th training data in D-dimensional space. The parameter updating rule of (6) is now

$$\Lambda^{(t+1)} = \Lambda^{(t)} - \varepsilon \nabla L(\Lambda^{(t)}|X). \tag{10}$$

The MCE learning algorithm based on the proposed criterion will be referred as mMCE in the following text.

4 Application to Neural Networks

The modified MCE learning criterion given in (21) can be applied to arbitrary discriminant functions that are second order differentiable in respect to the variables of the functions. For the present study, multi-layer perceptron type neural network is employed as a platform of recognizer to evaluate the performance.

For the p-th training data x_p, let $I_{pi}^{(m)}$ and $O_{pi}^{(m)}$ be the input and output of the i-th cell of layer m, respectively, where $1 \leq m \leq \mathcal{M}$. Then the input value of the i-th cell of layer m is given as

$$I_{pi}^{(m)} = \sum_{j=1}^{n_{m-1}} w_{ij}^{(mm-1)} O_{pj}^{(m-1)} + \theta_i^{(m)}. \tag{11}$$

Here, $w_{ij}^{(mm-1)}$ is the connection weight between the i-th cell of layer m and the j-th cell of layer $m-1$, $\theta_j^{(m)}$ is a constant and n_m represents the number of cells in layer m. The output of each cell is given as

$$O_{pi}^{(m)} = f(I_{pi}^{(m)}) \tag{12}$$

in which $f()$ is a sigmoid function of the form

$$f(x) = \frac{1}{1+e^{-x}}. \tag{13}$$

In the framework of the conventional error back-propagation (EBP), the object function is defined on the basis of least squared error (LSE), which is

$$E_{sq} = \frac{1}{2}\sum_{n=1}^{N}\sum_{i=1}^{n_{\mathcal{M}}}\left(t_{ni} - O_{ni}^{(\mathcal{M})}\right)^2 \tag{14}$$

where t_{ni} denotes the desired output for the i-th output cell against the n-th input.

On the other hand, in the proposed mMCE learning, the objective function that should be minimized is given in the form of (21). Under the new definition, the minimization of the object function is done by adjusting the weights of the network with the following algorithm.

$$w_{ij}^{(mm-1)} \leftarrow w_{ij}^{(mm-1)} - \alpha\frac{1}{N}\sum_{p=1}^{N}\Delta w_{pij}^{(m)} \tag{15}$$

The weight adjustment $\Delta w_{pij}^{(m)}$ is

$$\begin{aligned}\Delta w_{pij}^{(m)} &= \frac{\partial L(\Lambda|X)}{\partial w_{ij}^{(mm-1)}} \\ &= \frac{\partial L_0(\Lambda|X)}{\partial w_{ij}^{(mm-1)}} + \gamma\frac{\partial F(\Lambda)}{\partial w_{ij}^{(mm-1)}}\end{aligned} \tag{16}$$

In case of the three-layer network, each term in (16) is given as follows. In the output layer ($m=3$),

$$\begin{aligned}\frac{\partial L_0}{\partial w_{kj}^{(mm-1)}} &= \sum_{k=1}^{n_m}\ell_k'(I_{pk}^{(m)};\Lambda)1(I_{pk}^{(m)} \in C_k) \\ &\quad \frac{\partial d_k(I_{pk}^{(m)};\Lambda)}{\partial I_{pk}^{(m)}}O_{pj}^{(m-1)}\end{aligned} \tag{17}$$

$$\begin{aligned}\frac{\partial F}{\partial w_{kj}^{(mm-1)}} &= \frac{1}{2}\sum_{l=1}^{n_{m-2}}\left(\sum_{t=1}^{n_{m-1}} w_{kt}^{(mm-1)}\left(w_{tl}^{(m-1m-2)}\right)^2\right. \\ &\quad \left. f''(I_{pt}^{(m-1)})\right)w_{jl}^{(m-1m-2)}f''(I_{pj}^{(m-1)})\end{aligned} \tag{18}$$

where $I_{pj}^{(m)}$ denotes the input to the j-th cell of layer m.

In the hidden layer ($m = 2$),

$$\frac{\partial L_0}{\partial w_{ji}^{(mm-1)}} = \sum_{k=1}^{n_{m+1}} \left(\frac{\partial L}{\partial I_{pk}^{(m+1)}} w_{kj}^{(m+1m)} \right) \frac{\partial d_j(I_{pj}^{(m)}; \Lambda)}{\partial I_{pj}^{(m)}} O_{pi}^{(m-1)}, \tag{19}$$

$$\begin{aligned} &\frac{\partial F}{\partial w_{ji^*}^{(mm-1)}} \\ &= \frac{1}{2} \sum_{i=1}^{n_{m-1}} \left\{ 2\delta_{ii^*} f''(I_{pj}^{(m)}) w_{ij}^{(mm-1)} + I_{pi^*}^{(m-1)} \right. \\ &\left. \left(w_{ij}^{(mm-1)}\right)^2 \left[(1 - 2f(I_{pj}^{(m)})) f''(I_{pj}^{(m)}) - 2\left(f'(I_{pj}^{(m)})\right)^2 \right] \right\} \\ &\sum_{k=1}^{n_{m+1}} w_{kj}^{(m+1m)} \left(\sum_{t=1}^{n_m} w_{kt}^{(m+1m)} \left(w_{ti}^{(mm-1)}\right)^2 f''(I_{pt}^{(m)}) \right) \end{aligned} \tag{20}$$

where δ_{nn^*} is the Kronecker symbol.

5 The 2-stage Building Learning

This senction describes the 2-stage Building Learning, which is the simplest case in the framework of Model Building Learning.

5.1 Outline

Figure 1 shows a basis construction of the 2-stage Building Learning.

The 2-stage Building Learning (2BL) is a method which re-evaluates the miss-classified data by using a classification method such as Bayes decision rule, Support Vector Machines (SVM), Hidden Markov Models (HMM) and so on. In the first step of 2BL, data that are difficult to classify correctly are chosen, and they are examined closely for the following second stage. It is well-know that one of the drawbacks of the MCE / GPD learning is its computational expensiveness. The 2BL makes it possible to decrease the computation time of the MCE/GPD learning by supplementarily employing a comparatively inexpensive method such as Bayes decision rule and K-nearest Neighbors in the second step.

Since the second stage of the 2BL is invoked only when misclassification error occurs in the first stage, the 2BL gives the same learning result with that of the MCE / GPD learning in case that there are no misclassification data found in the first stage.

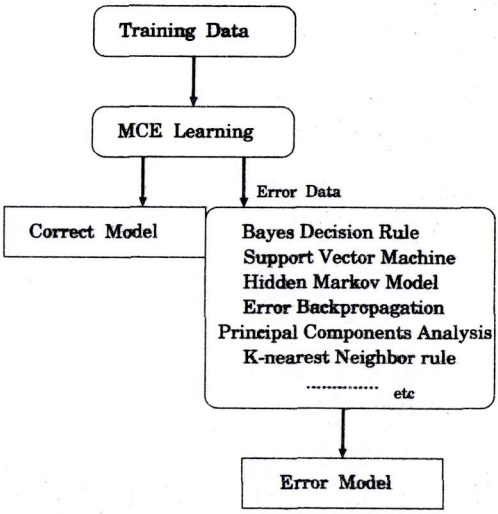

Fig. 1: a basis construction of the 2-stages Building Learning.

It is clear that simply applying the same classifier to the missclassification data in the second stage has no effect on improvement of the recognition performance. Additional features are needed to improve the classification performance. As a new feature for classification, we propose to use the value given by the misclassification measure of the MCE/GPD learing. As a result of adding a new feature to the original feature space, the misclassified date in the first stage are dealt in the new feature space.Figure 2 shows an idea on giving new features. The proposed 2BL can solve the both problems of declining the generalization performance and computatinally expensiveness as overlearning.

5.2 Evaluation

In case of two models based on input data and misclassification data, this section describes the method to estimate two models eqnally.

The set of data can be classified into either clear zone or gray zone according to the distribution of data.

The clear zone is the area where the distribution of the class data has no overlaps with others, whereas the gray zone is the area where the distribution overlaps with others. Most of the Misclassified data exist in gray zone. It is difficult to evaluate the two models produced together.

The evaluation process of 2BL is given as follows.

1. Calculate the misclassification measure of (2), and give a number of class to

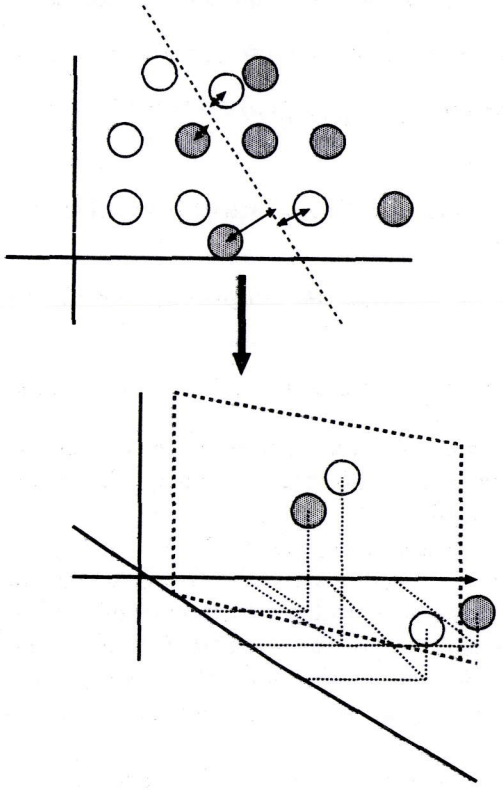

Fig. 2: additional feature.

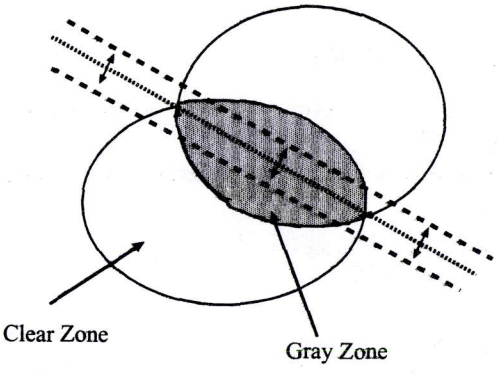

Fig. 3: a concept of proposed evaluation.

found data, if it has aa value of found data is negative.

2. Find the mean value μ_i of each misclassification measure.

3. Decision Parameter $Q = \frac{1}{AP}\sum_{p=1}^{P}\mu_i, A \geq 2$. P is the number of data.

4. If $Q \geq d_i(\mathbf{x}, \Lambda_i)$, class number of data decides i. Let reevaluate data in second step, if $Q < d_i(\mathbf{x}, \Lambda_i)$.

5. Let component of data add the value of misclassification measure as new feature.

6. Decide recognition result used second model as class number of data.

The greater the value A takes, the more the recognition result depends on the correct model. The smaller the value A takes, the more the recognition result depends on wrong model.

The discriminative learning such as the MCE / GPD learning has an inclination to adapt the parameters specifically to the training data in order to achive minimum classification error. The proposed 2BL method is able to avoid declining the generalization performance by reevaluating data around the decision boundary.

Some data decided correct in first evaluation have possibility to be decided fault in second evaluation according to circumstances, if input data are evaluated by this proposed evaluation.

But from figure 4, data in area A, B and C have possibility to be misrecognized by the conventinal methods. On the other hand, data in area C have possibility to be correctly classified by the proposed method. Thus, recognition performance as a whole is expected to improve, even if some data are decided correctly in the first evaluation.

5.3 Decision Parameter Rule

This section describes how to determine the value of decision parameter $Q = \frac{1}{A}\mu$. Let $B = \frac{1}{A}$ so that $Q = B\mu$.

We, at first, calculate the average normalized within-class distance γ as

$$\gamma = \frac{1}{P}\sum_{p=1}^{P}(\mathbf{x}_p - \mu^{(c)})^t \Sigma^{(c)^{-1}}(\mathbf{x}_p - \mu^{(c)}) \tag{21}$$

$$\Sigma^{(c)} = \begin{bmatrix} \sigma_{11}^{(c)} & \sigma 12^{(c)} & \cdots & \sigma_{1D}^{(c)} \\ \sigma_{21}^{(c)} & \sigma 22^{(c)} & \cdots & \sigma_{2D}^{(c)} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{D1}^{(c)} & \sigma_{D2}^{(c)} & \cdots & \sigma_{DD}^{(c)} \end{bmatrix} \tag{22}$$

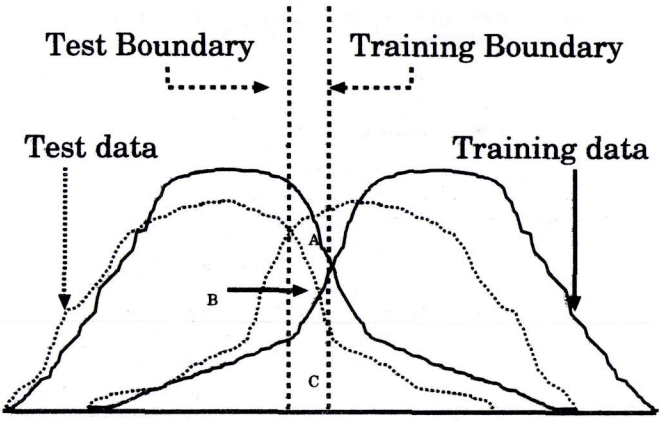

Fig. 4: reason of misevaluation.

$$\sigma_{ij}^{(c)} = E[(x_i^{(c)} - \mu_i^{(c)})(x_j^{(c)} - \mu_j^{(c)})] \tag{23}$$

where the $\mathbf{x}_p = (x_{1p}, x_{2p}, \cdots, x_{Dp})$ denotes the p-th training data, $\mu^{(c)}$ denotes the mean vector of class c, and $\Sigma^{(c)}$ denotes the covariance matrix of class c.

Using the distance γ, the parameter B is given by

$$B = \frac{\mathcal{P}_s C}{\mathcal{P}_{s-1}} \frac{1}{1 + e^{-\theta\gamma}},\ \theta > 0,\ s = 1, 2, \cdots, n \tag{24}$$

where C represents the number of classes, $\mathcal{P}_s$ represents the number of pattern vectors used in the s-th model. In case that γ takes a big value and B is close to $P_s C / P_{s-1}$, the distribution of each class probably overlaps with others. Hence, further inspectation of misclassified data is important to improve the classification performance.

6 Multi-stage Building Learning

This section describes Multistage Building Learning (MBL), an extension of the 2-stage Building Learning (2BL). Figure 5 shows a basis construction of Multistages Building Learning.

The MCE / GPD learning has a problem of overlearning.

MBL incorporates the misclassification measure into the feature vector space used in the former stage. As a result, the dimension of feature space increases as the stage progresses. On the other hand, the number of data given to the next stage decreases. The stage-building process terminates when no more data remains for the next stage.

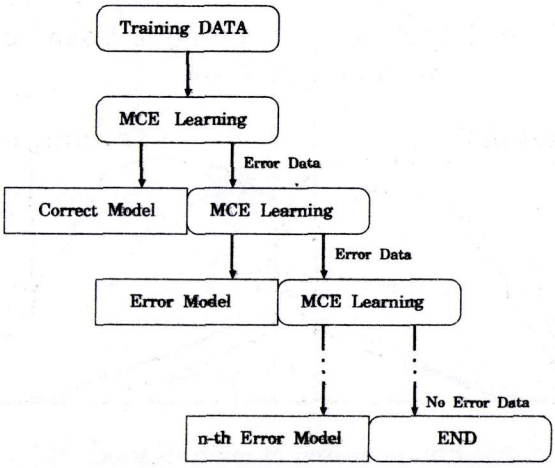

Fig. 5: a basis construction of Multistages Building Learning.

1. Let $g_k(\mathbf{x}; \Lambda_k)$ be a discriminant function with positive value to discriminate a data of class C_k from the other classes, where $\mathbf{x}$, Λ_k denote a vector in D-dimensional feature space and the set of parameter of the discriminant function, respectively.

2. For each data with feature vector $\mathbf{x}$, calculate the following misclassification measure

$$d_k(\mathbf{x}) = -g_k(\mathbf{x}; \Lambda_k) + \left[\frac{1}{M-1} \sum_{j, j \neq k} g_j(\mathbf{x}; \Lambda_k)^{\eta} \right]^{1/\eta} \tag{25}$$

3. convergent decision

$$L_0(\Lambda) = \frac{1}{P} \sum_{p=1}^{P} \sum_{i=1}^{M} \ell(d_i(\mathbf{x}_p, \Lambda_i)) 1(\mathbf{x}_p \in C^i) \tag{26}$$

4. In case of convergence, for each data $\mathbf{x} \in R^D$ that satisfies $d_k(\mathbf{x}, \Lambda_i) > 0$, replace the $\mathbf{x}$ with a new vector $\mathbf{x} in R^{D+1}$ in which the $D+1$-th element of $\mathbf{x}$ is $d_k(\mathbf{x}, \Lambda_i)$.

5. Repeat step $1 \sim 4$ after giving misclassification data added new component to next learning.

6. Stop the procedure in case of $s < M$, where s denotes the number of transfered data, M : the number of classes.

It should be noted that, like the case of 2BL, the training method used in the last stage in MBL can be different from the MCE/GPD learning that is used in the other stages.

7 Experiments

In order to evaluate the classification performance of the proposed method, three-layer feed-forward neural networks were employed. Since the MCE learning is computationally expensive, the network was at first trained by the conventional error back-propagation learning (EBP) that minimizes squared error given in (14), and then the MCE or mMCE learning was applied to the network.

In the experiments on real-world data, three datasets of two-class problems, "cancer", "house" and "sonar" in the UCI Machine Learning Repository [1] from University of California Irvine were used.

Table 1: Correct classification rate [%]

	database	cancer	house	sonar
	# classes	2	2	2
	# training data	420	265	141
	# testing data	279	170	67
	# attributes	9	15	60
	# hidden unit	12	12	12
Bayes(ML)	Training	95.0	98.8	100.0
NN(EBP)		91.9	96.3	95.0
NN(MCE)		93.6	97.4	92.9
NN(mMCE)		95.0	94.3	91.5
Bayes(ML)	Test	95.7	96.4	74.6
NN(EBP)		90.3	96.5	82.1
NN(MCE)		94.3	95.3	85.1
NN(mMCE)		95.7	97.7	89.6

Table 1 shows the experimental result for the four different learning algorithms, Bayes+ML, NN(EBP), MCE and mMCE. Here, Bayes+ML denotes the quadratic discriminant functions in which single normal distribution with full-covariance matrix is assumed for each category. It can be found that mMCE gives the best classification performance on each testing set. Compared to the performance improvements

[1] C.J. Merz and P.M. Murphy. UCI repository of machine learning database, 1996.

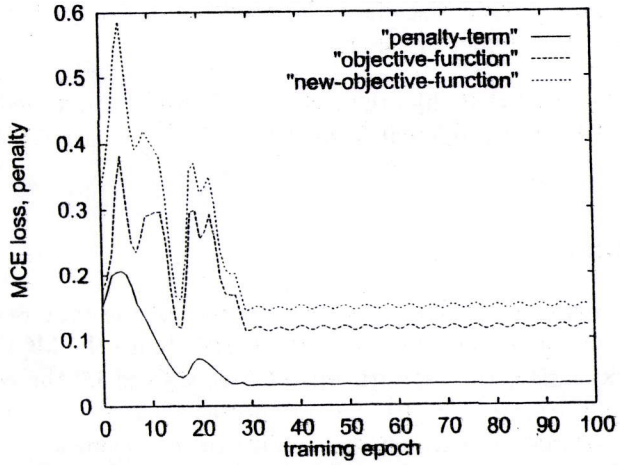

Fig. 6: Learning curves of the loss functions L_0, L and penalty function F in terms of training epochs.

from MCE to mMCE for the training set and testing set, the improvement on testing set is larger than that of the training set. This certifies that the employed penalty term of (23) is effective for improving the generalization performance of the recognizer.

Fig. 6 shows the learning curves of the MCE loss function L_0, the penalty function F and the mMCE's total loss function L in (21).

Fig. 8 shows the correct classification rates in terms of the parameter ξ in (5). Although ξ influences the correct rate, mMCE performs better than MCE for any value of ξ.

The relationship between the parameter γ in (21) and the correct classification rates on the test set "house" is shown in Fig. **??**. It can be found in the figure that choosing the proper value of γ is crucial to get good genenaralization.

Another recognition experiment was conducted on a speech database of Japanese five vowels. The speech data of each vowel were extracted from the ATR continuous speech database (B-set) according to the phoneme transcription given to the database. Among the data of six male subjects, the data of four subjects (msh, mmy, mht, mho) was used for training, and the data of the remained subjects (myi, mtk) was used for testing. Table 2 shows the experimental results on the speech database.

It can be seen in the table that mMCE shows better generalization performance than MCE.

For the performance evaluation on real-world problem, three datasets of two-classes problem, "cancer", "house" and "sonar" in the UCI machine Learning repos-

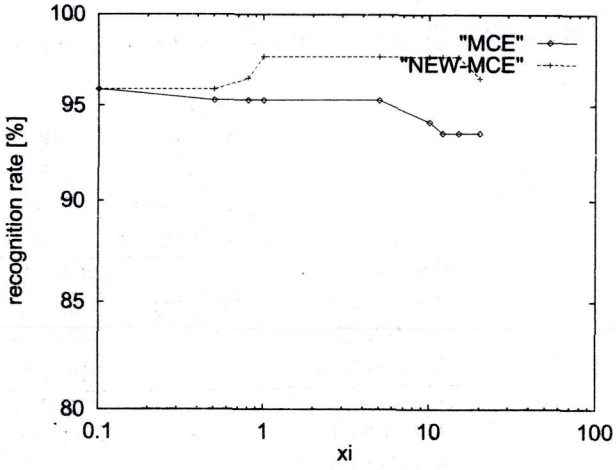

Fig. 7: Recognition performance on the test set "house" in terms of the parameter ξ.

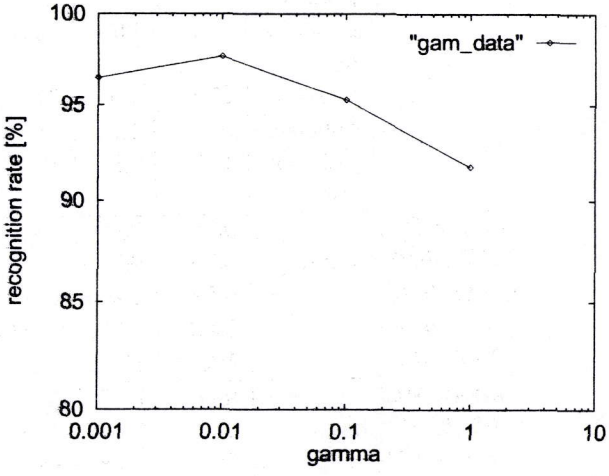

Fig. 8: Recognition performance on the test set "house" in terms of the parameter γ.

Table 2: Correct classification rate for Japanese 5 vowels

	♯ classes	5
	♯ training data	4000
	♯ testing data	1000
	♯ attributes	12
	♯ hidden units	12
Bayes(ML)	training	86.3 %
NN(EBP)		89.0 %
NN(MCE)		89.0 %
NN(mMCE)		88.1 %
Bayes(ML)	test	79.3 %
NN(EBP)		83.1 %
NN(MCE)		87.7 %
NN(mMCE)		90.4 %

itory [2] from University of California Irvine were used.

	database	Cancer	House	Sonar
	♯ classes	2	2	2
	♯ training	420	265	141
	♯ test	279	170	67
	dimension	9	15	60
	♯ hidden units	12	12	12
Bayes(ML)	training	95.0	98.8	100.0
NN(EBP)		99.3	99.6	98.6
NN(MCE)		97.8	100.0	98.6
NN(2BLno)		99.3	100.0	100.0
NN(2BL)		100.0	100.0	100.0
Bayes(ML)	test	95.7	96.5	74.6
NN(EBP)		91.8	95.3	79.1
NN(MCE)		92.5	95.8	86.6
NN(2BLno)		91.8	95.8	88.1
NN(2BL)		**94.6**	**96.5**	**88.1**

Table 3: Recognition rate for UCI machine learning databases(unit : %)

[2]C.J.Marz and P.M.Murphy. UCI repository of machine learning databases, 1996

Table 3 shows the experimental results for the four different learning methods, Bayes(ML), Error Back Propagation learning (NN(EBP)), MCE, 2BL and 2BLno. Bayes(ML) denotes the quadratic discriminant function in which single normal distribution with full-covariance matrix is assumed for each category. 2BLno is same with 2BL excepting that the step of adding a new feature to the original feature vector is discarded. Both 2BL and 2BLno use the MCE / GPD learning in the first stage and the second stage. It can be seen that 2BL and 2BLno give good classification performance on each testing set. Specially, 2BL gives very good classification performance on "sonar". It can be said that the misclassification measure is effective to improve the classification performance, especially for the case when the dimension of the feature is smaller enough than the number of data.

	♯ classes	26
	♯ training data	6238
	♯ testing data	1559
	♯ attributes	617
	♯ hidden units	32
NN(EBP)	training	99.39 %
NN(MCE)		96.94 %
NN(2BL)		99.94 %
NN(EBP)	test	94.29 %
NN(MCE)		95.45 %
NN(2BL)		**95.96** %

Table 4: Recognition rate for UCI machine learning database "Isolet"

Learning Method	Time	ratio object/MCE
NN(EBP)	24180 sec	0.935
NN(MCE)	25860 sec	1.000
NN(2BL)	16320 sec	0.631

Table 5: Training time for UCI machine learning database "Isolet"

Table 4 shows the experimential results of multiclass problem for the three different learning algorithms, NN(EBP), MCE and 2BL. Table 5 shows the computation time of training for the dataset "Isolet". ¿From these results, it can be seen that 2BL gives both the best test-set recognition performance and fastest learning speed.

Table 6 shows the experimential results used speech database of japanese five vowels as real-world data. The speech data used in this experiment consists of cutting speech sections of five vowels based on inspected labels from six spe a kers in ATR speech database (B-set). Training data consists of speech data from f

	♯ classes	5
	♯ training data	4000
	♯ testing data	1000
	♯ attributes	12
	♯ hidden units	12
Bayes(ML)	training	86.3 %
NN(EBP)		89.0 %
NN(MCE)		89.0 %
NN(2BL)		89.4 %
Bayes(ML)	test	79.3 %
NN(EBP)		83.1 %
NN(MCE)		87.7 %
NN(2BL)		**88.2** %

Table 6: Recognition rate for 5 vowels speech data from japanese speakers

our speakers (msh, mmy, mht, mho), and test data does of speech data from two sp e akers (myi, mtk). It can be seen in the table that 2BL shows the best recognitio n performance of all methods.

8 Conclusion

Improvement of generalization performance of the Minimum Classification Error (MCE) learning was proposed by employing a regularizer to the objective function. Three-layer feed-forward neural networks were employed to demonstrate the effectiveness of the proposed method. Compared to the original MCE learning, the proposed mMCE learning showed better recognition performance on testing data while it showed comparable performance on training data. This implies that the proposed regularizer is effective for improving the generalization performance of the recognizer.

Since the weight parameter γ for the penalty function was heuristically determined in the experiments, further investigataion should be taken in order to develop a criterion for determining the parameter.

Thereto, 2-stage Building Learning (2BL) and Multi-stage Building Learning (MBL) were proposed. Both methods consists of more than one recognition models by using misclassification measure. The three-layers feed-forward neural networks were employed to demonstrate the effectiveness of the proposed 2BL. Comparing with other learning methods, the proposed methods shows generally good recognition performance for test data than that for training data. Specially, It could be found that the proposed method give high recognition performance for hard classification data in case of using other learnng methods. It is conceivable that one

of reason to give high recognition performance is to employ the misclassification measures as new features for few elements.

There is a method to decide to stop learning well-timed in previous step as one of problems to be solved from these experiments. If the the learning process of the previous stage stops very early, the model can not have good recognition performance. On the other hand, if it takes much time to converge, the proposed method is computatinally expensive and has a possibility of overlearning.

So we have to establish a criterion to stop the learning in previous step. This problem can be solved by using mMCE to the last learning stage of MBL.

Authors wish to thank Dr Gunnar Rätsch and Dr Kanad Keeni for discussions in preparing the form of the paper

References

[1] Biing-Hwang Juang and Shigeru Katagiri, "Discriminative Leaning for Minimum Error Classification", IEEE Transaction on Signal Processing, Vol.40, No.12, December, (1992).

[2] S. Amari, "A theory of adaptive pattern classifers", IEEE Trans. EC, Vol.16, pp.299–307, (1967).

[3] Eric McDermott and Shigeru Katagiri, "Prototype-based minimum classification error / generalized probabilistic desent training for various speech units", Computer Speech and Language, pp.351–356, August, (1994).

[4] W. Chou, C-H. Lee and B-H. Juang, "Minimum Error Rate Training of Inter-word Context Dependent Acoustic Model Units in Speech Recognition", Proc. ICASSP94 II, pp.652–655, (1994).

[5] Biing-Hwang Juang and Wu Chooud and Chin-Hui Lee, "Minimum Classification Error rate methods for speech recognition", IEEE Trans. SAP, 5(3), pp.257–265, (1997).

[6] A.N.Tikhonov and V.Y.Arsenin, "Solutions of Ill-posed Problems", V.H.Winston, (1977).

[7] A.N.Tikhonov and A.v.Goncharsky and V.V.Stepanov and A.G.Yagola, "Numerical Methods for the Solution of ill-posed Probrems", Kluwer Academic Publishers, (1990).

[8] Christopher M.Bishop, "Curvature-Driven Smoothing: A Learning Algorithm for Feed-forward Networks", IEEE Transactions on Neural Networks, Vol.4, No.5, pp.882–884, (1993).

[9] Christopher M.Bishop, "Neural Networks for Pattern Recognition", Oxford University Press, (1995).

[10] H. Kuwahara et al., "Construction of a large-scale Japanese speech database and its management system", Proc. ICASSP-89, pp.560–563 (1989).