FAST LEARNING BY AN ANTI-REGULARIZATION TECHNIQUE

Yoshihiko HAMAMOTO†, Yoshihiro MITANI††, Toshinori HASE†††, and Shingo TOMITA†

†Department of Computer Science and Systems Engineering, Faculty of Engineering, Yamaguchi University
††Graduate Student, Department of Computer Science and Systems Engineering, Faculty of Engineering, Yamaguchi University
†††NEC Engineering

An anti-regularization technique recently proposed by Raudys is studied in small training sample size situations. Experimental results show that the anti-regularization technique offers significant advantages over the BP algorithm in terms of the learning time.

Key Words : neural network classifier, regularization, generalization ability, learning time

1. INTRODUCTION

In designing artificial neural network (ANN) classifiers, the Back-Propagation algorithm [11] has been used. However, there are two serious problems in the BP algorithm: One is the extremely long learning times, and the other is the possibility of trapping in local minima. Recently, Raudys [10] proposes an anti-regularization technique to improve the BP algorithm. He points out that the magnitude of the weights of a network is a major factor which influences the generalization ability. In the anti-regularization technique, the magnitude of the weights is addressed. However, very little is known about the properties of the anti-regularization technique, particularly in small training sample size situations. In this paper, we study the properties of the anti-regularization technique in small training sample size situations. Experimental results show that the anti-regularization technique offers significant advantages over the conventional BP algorithm in terms of the learning time.

2. ANTI-REGULARIZATION

We will consider ANN classifiers with one hidden layer. The units in the input layer correspond to the components of the feature vector to be classified. The hidden layer has m units. The units in the output layer are associated with pattern class labels. Here, we consider the L-class problems. Thus, the output layer has L units. In the network discussed here, the inputs to the units in each successive layer are the outputs of the preceding layer. Initial weights were distributed uniformly in -0.5 to 0.5.

The cost function in the anti-regularization can be defined by

$$J = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{L}(t_{ij} - y_{ij})^2 + \frac{1}{2}\lambda \sum_{k \in C} w_k^2 \qquad (1)$$

where y_{ij} is the output of output unit j corresponding to the i-th training sample $\boldsymbol{x}_j$, N is the total number of training samples, t_{ij} denotes a desired output for $\boldsymbol{x}_j$, λ is the regularization parameter, w_k is a weight, and C denotes a set of weights in the network.

In order to train the ANN classifier, we adopted the BP algorithm in which the momentum was not used for simplicity. Note that when $\lambda = 0$, regularization effects disappear. In the anti-regularization, a negative value of λ is used. On the other hand, in the conventional regularization [1], a positive value of λ has been used. Learning was terminated when the mean-squared error dropped below a specified threshold, or when there is little change in the mean-squared error. Here, the maximum number of iterations was set to 10000.

3. EXPERIMENTAL RESULTS

(1) Experiment 1

We used the Ness data set [7] which consists of n-dimensional Gaussian data. This data set was used to study the performance of ANN classifiers in the small sample, high dimensional setting [2]. The distribution parameters, $\boldsymbol{\mu}_i$ and Σ_i, are shown as follows:

$$\boldsymbol{\mu}_1 = [0, 0, \cdots, 0]^T \quad \boldsymbol{\mu}_2 = [\Delta/2, 0, \cdots, 0, \Delta/2]^T$$

$$\Sigma_1 = I_n \qquad \Sigma_2 = \begin{bmatrix} I_{n/2} & O \\ O & \frac{1}{2} I_{n/2} \end{bmatrix}$$

where Δ is the Mahalanobis distance between class ω_1 and class ω_2, and I_n is the $n \times n$ identity matrix. In this data set, the true Bayes error can be controlled by varying the values of Δ and n. Hence, we adopted this data set. We assume that the class prior probabilities are equal. In practice, a fixed number of training samples is used when designing a classifier. We are mainly interested in the effects of the anti-regularization in small training sample size situations. It is particularly worth noting that the number of training samples per class should be at least five to ten times the dimensionality [3]. On the other hand, in order to neglect the test sample size effect, a large test sample should be used for error estimation. Note that the training samples are statistically independent of the test samples. Hence, the generalization error can be accurately estimated by using these test samples. Graphs were obtained by averaging the results of 100 Monte Carlo trials with different sample sets of fixed size and different initial weights.

The purpose of this experiment is to study the influence of the regularization parameter λ on the generalization error as well as the learning time. The following experiment was conducted.

Dimensionality	:	$n = 2,\ 20$
Training sample size	:	10 per class
Test sample size	:	1000 per class
Values of Δ	:	2, 3
Hidden unit size	:	$m = 256$

Figs. 1 and 2 show results. Note that the use of $\lambda = 0$ leads to the conventional BP algorithm. In the anti-regularization, the magnitude of the weights increases extremely. This often results in the early stopping. This property of the anti-regularization results in considerable computational savings. Experimental results show that the anti-regularization technique with proper value of λ significantly increases the learning speed, providing smaller generalization errors than the BP algorithm. The proper value of λ depends on the given data set.

(2) Experiment 2

It is well known that the performance of ANN classifiers is influenced by the hidden unit size m. The purpose of this experiment is to study the effect of the anti-regularization technique for various values of m using a real data set.

We used the 8OX data set [4]. This consists of 45 8-dimensional characters: 8, O, and X extracted from the Munson's character database. This data set was also used to study the performance of ANN classifiers [12]. The 8OX data set was randomly partitioned into two sets: One set was used to train the classifier and the other was used to evaluate the performance. This procedure was independently repeated ten times. The following experiment was conducted.

Dimensionality	:	$n = 8$
Training sample size	:	7 per class
Test sample size	:	8 per class
Value of λ	:	-0.1

Fig. 3 shows graphs which were obtained by averaging the results of the 10 trials. Again, the learning time was significantly reduced by the anti-regularization technique. Nevertheless, the anti-regularization technique provided the generalization error comparable to the BP algorithm.

4. DISCUSSION

Motivated by Raudys' work, we have studied an anti-regularization technique for training ANN classifiers. The properties of the anti-regularization technique have been studied on artificial and real data sets. As Raudys [10] points out, the anti-regularization technique is one of possible means to control the magnitude of the weights. Experimental results show that the anti-regularization technique with the proper value of λ significantly outperforms the BP algorithm in terms of the learning time. The previous algorithms proposed to reduce the learning time do not address the generalization error directly [5, 6, 8, 9, 13]. Thus, these algorithms may not guarantee good generalization. On the other hand, the anti-regularization technique is at least ten times faster than the BP algorithm, providing the generalization error comparable to the BP algorithm. Therefore, the anti-regularization technique should be considered in the design of ANN classifiers, especially in the high dimensional setting.

REFERENCES

1) Girosi, F., Jones, M. and Poggio, T.: Regularization theory and neural networks architectures, *Neural Comp.*, **7**, pp.219-269, 1995.

2) Hamamoto, Y., Uchimura, S. and Tomita, S.: On the behavior of artificial neural network classifiers in high-dimensional spaces, *IEEE Trans. PAMI*, **18**(5), pp.571-574, 1996.

3) Jain, A. K. and Chandrasekaran, B.: Dimensionality and sample size considerations in pattern recognition practice, In *Handbook of Statistics*, **2**, P. R. Krishnaiah and L. N. Kanal, Eds., North-Holland Publishing Company, pp.835-855, 1982.

4) Jain, A. K. and Dubes, R. C.: Algorithms for clustering data, Prentice Hall, 1988.

5) Lee, C. W.: Learning in neural networks by using tangent planes to constraint surfaces, *Neural Networks*, **6**(3), pp.385-392, 1993.

6) Møller, M. F.: A scaled conjugate gradient algorithm for fast supervised learning, *Neural Networks*, **6**(4), pp.525-533, 1993.

7) Ness, J. V.: On the dominance of non-parametric Bayes rule discriminant algorithms in high dimensions, *Pattern Recognition*, **12**, pp.355-368, 1980.

8) Ochiai, K., Toda, N. and Usui, S.: Kick-out learning algorithm to reduce the oscillation of weights, *Neural Networks*, **7**(5), pp.797-807, 1994.

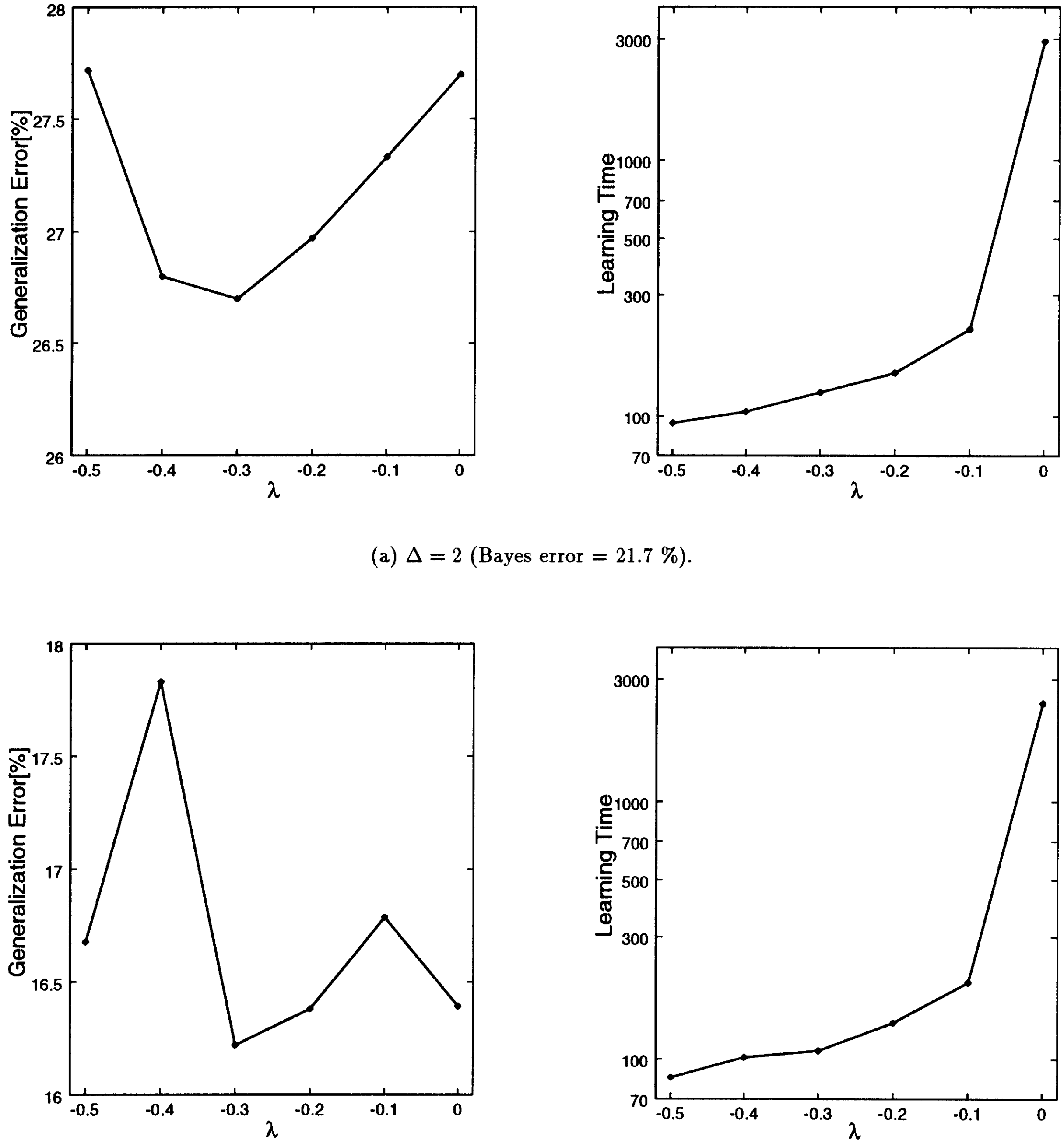
9) Parlos, A. G., Fernandez, B., Atiya, A. F., Muthusami, J. and Tsai, W. K.: An accelerated learning algorithm for multilayer perceptron networks, *IEEE Trans. Neural Networks*, **5**(3), pp.493-497, 1994.

10) Raudys, S.: A negative weight decay or antiregularization, *Proc. Int. Conf. Artificial Neural Networks*, Paris, 1995.

11) Rumelhart, D. E., Hinton, G. E. and Williams, R. J.: Learning internal representations by error propagation, In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, ch. 8, MIT Press, 1986.

12) Schmidt, W. F., Levelt, D. F. and Duin, R. P. W.: An experimental comparison of neural classifiers with 'traditional' classifiers, In *Pattern Recognition in Practice* IV, E. S. Gelsema and L. N. Kanal, Eds., Elsevier Science B. V., pp.391-402, 1994.

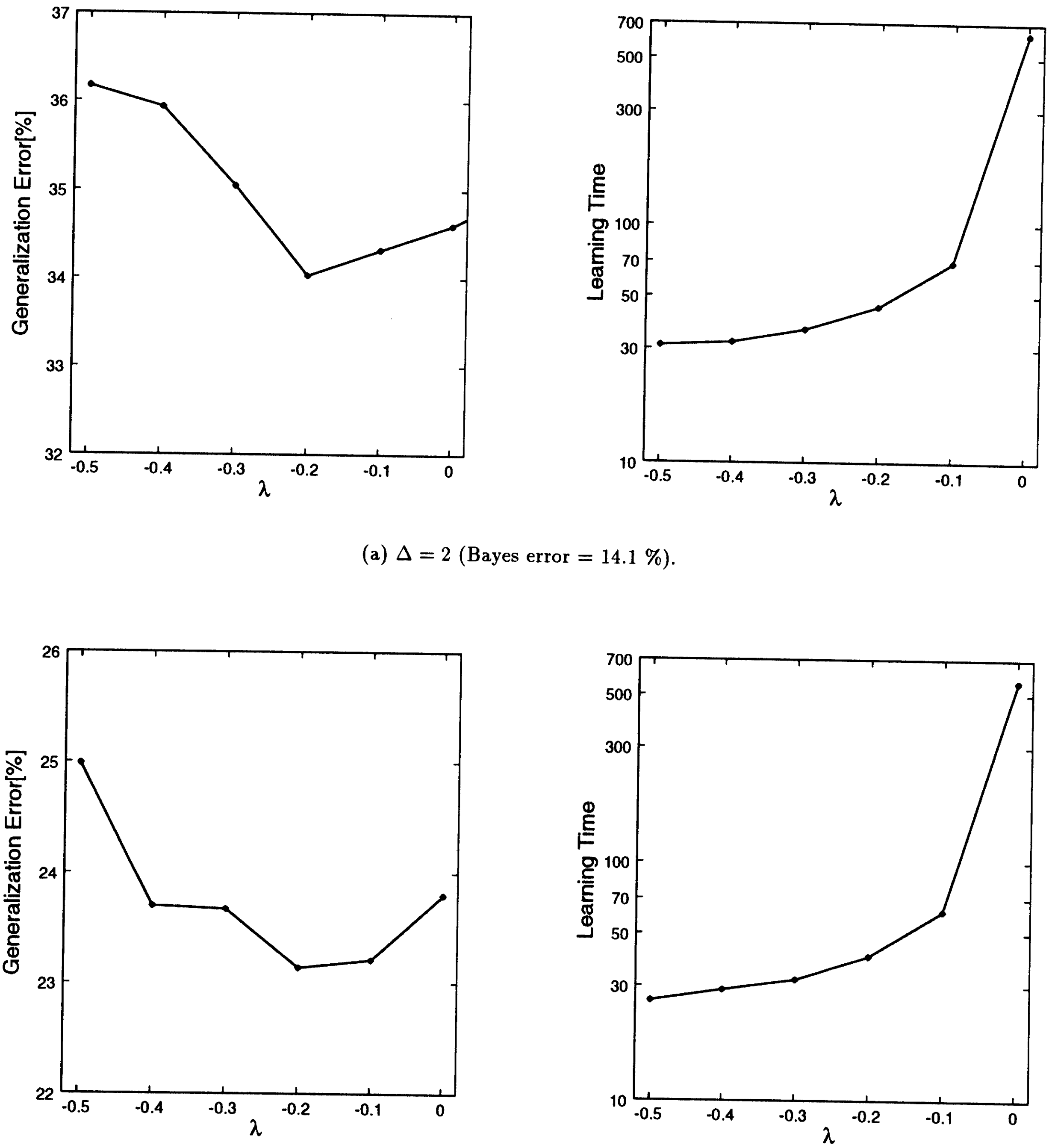
13) Yu, X.-H., Chen, G.-A. and Cheng, S.-X.: Dynamic learning rate optimization of the backpropagation algorithm, *IEEE Trans. Neural Networks*, **6**(3), pp.669-677, 1995.



(a) $\Delta = 2$ (Bayes error = 21.7 %).

(b) $\Delta = 3$ (Bayes error = 12.3 %).

Fig.1 Learning results on the 2-dimensional Ness data set.



(a) $\Delta = 2$ (Bayes error = 14.1 %).

(b) $\Delta = 3$ (Bayes error = 8.5 %).

Fig.2 Learning results on the 20-dimensional Ness data set.

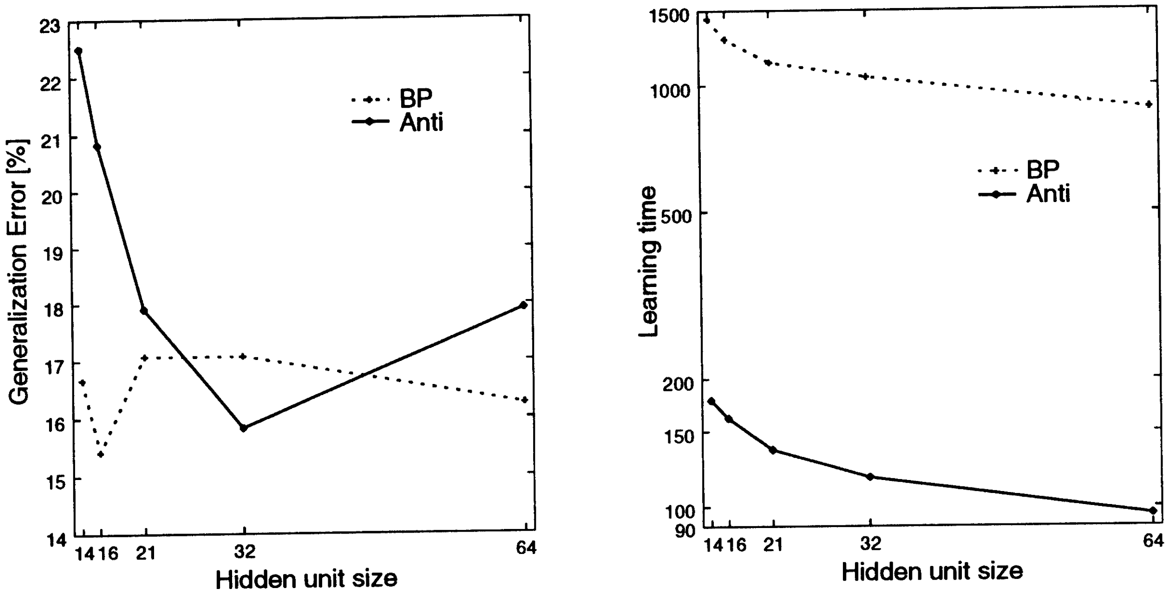


Fig.3 Learning results on the 8OX data set.

(Received April 15, 1997)

アンチ正則化法による高速学習

浜本義彦，三谷芳弘，長谷俊徳，富田眞吾

ニューラルネット識別器を設計する場合，これまでBP法が用いられてきた．しかしながら，BP法には，1．局所解が得られる，2．学習時間が大である，という問題点がある．これらの問題点を解決するため，最近Raudysはアンチ正則化法を提案した．これは，通常の正則化法とは異なり，正則化パラメータに負の値を用いるものである．本論文では，このアンチ正則化法を，訓練サンプル数が少ないという現実的な状況下で詳しく調べ，その特長を明らかにした．