Optimal Learning Rates for Each Pattern and Neuron in Gradient Descent Training of Multilayer Perceptrons

Sang-Hoon Oh and Soo-Young Lee
Department of Electrical Engineering & Brain Science Research Center
KAIST, 373-1 Kusong-dong, Yusong-gu, Taejon, Korea
shoh@eeinfo.kaist.ac.kr, sylee@ee.kaist.ac.kr

Abstract— **This paper proposes optimal learning rates in gradient descent training of multilayer perceptrons, which are a separate learning rate for weights associated with each neuron and a separate one for assigning virtual hidden targets associated with each training pattern. That is, a hidden weight vector has two optimal learning rates, the one for assigning virtual hidden targets and the other for minimizing a hidden error function proposed in this paper. Effectiveness of the proposed error function was demonstrated for a handwritten digit recognition and an isolated-word recognition tasks and very fast learning convergence was obtained.**

I. Introduction

A popular method of training multilayer perceptrons(MLPs) is the error backpropagation(EBP) algorithm which is a gradient descent with fixed learning rate[1]. In order to accelerate the EBP algorithm, which has a drawback with slow convergence, many proposed a modified error function which showed improved convergence[2]. However, they are still based on the gradient descent with non-optimal learning rates.

Another approach is second-order nonlinear optimizing methods such as conjugate gradient or Newton's methods. The critical drawbacks of these methods, however, are the ill conditioning of the Hessian matrix in many applications and the computational complexity related to the Hessian[3]. In order to overcome the drawbacks of the Hessian, many proposed the layer-by-layer(LBL) optimizing algorithm which decomposes each layer of MLP into both a linear part and a nonlinear part[3]. This method also showed a stalling problem due to assigning hidden targets[4].

Besides the above approaches, an adaptive learning rate in the EBP algorithm was also proposed. We concentrate on this approach since the EBP algorithm will be more powerful if it has an optimal learning rate. The optimum value will depend on a particular problem, and will typically vary during training.

One for adapting the learning rate is the bold driver(BD) technique[5]. This increases the learning rate if the error decreases at a given step. Otherwise, the learning rate is decreased. Jacobs, also, proposed the delta-delta rule in which a separate learning rate for each weight was adapted during the training process[6]. That is, a particular learning rate is increased when the derivative of error with respect to the corresponding parameter has the same sign on consecutive steps. If not, the learning rate is decreased. A modification to this algorithm, known as the delta-bar-delta(DBD) rule, was also introduced on the assumption that the weights could be regarded as relatively independent. The weights, however, are strongly coupled in a typical neural networks. Besides the above methods, Fahlman proposed the quickprop algorithm in which the error surface was approximated by a quadratic polynomial of each weight[7]. Although all the heuristics showed improved convergence in some problems, any of them is not optimum.

This paper proposes optimal learning rates for each neuron and pattern in the EBP algorithm. Section II briefly reviews MLPs and section III introduces an optimal learning rate for an output weight vector. Section IV proposes optimal learning rates for a hidden weight vector. To derive them, virtual hidden targets are assigned and a new hidden error function is defined. In section V, effectiveness of the optimal learning rate is demonstrated for two recognition tasks. Finally, section VI concludes this paper.

II. Multilayer Perceptrons

Consider an MLP consisting of N inputs, H hidden neurons, and M output neurons. When an input pattern $\mathbf{x} = [x_1, x_2, \ldots, x_N]$ is presented to the MLP, the jth hidden neuron's value is given by

$$h_j = f(\hat{h_j}) = \tanh(\hat{h_j}/2), j = 1, 2, \ldots, H, \qquad (1)$$

where $f(.)$ is the sigmoidal activation function of hidden neuron and

$$\hat{h_j} = \sum_{i=0}^{N} w_{ji} x_i \qquad (2)$$

is the net-input to h_j. Here w_{j0} with $x_0 = 1$ is the bias and w_{ji} is the weight connecting x_i to h_j. Also, the net-input to y_k(the kth output neuron) is

$$\hat{y_k} = \sum_{j=0}^{H} v_{kj} h_j, k = 1, 2, \ldots, M, \quad (3)$$

where v_{k0} with $h_0 = 1$ is the bias and v_{kj} is the connection weight between h_j and y_k. We here assume that the output neuron is linear, that is $y_k = \hat{y_k}$.

For P training patterns $\mathbf{x}^{(p)}(p = 1, 2, \ldots, P)$ with associated target vectors of output layer $\mathbf{t}^{(p)} = [t_1^{(p)}, t_2^{(p)}, \ldots, t_M^{(p)}]$, the weights should be optimized to minimize the MSE of output layer given by

$$E^{out} = \frac{1}{2} \sum_{p=1}^{P} \sum_{k=1}^{M} (t_k^{(p)} - y_k^{(p)})^2. \quad (4)$$

III. Optimal Learning Rate for an Output Weight Vector

In the EBP algorithm with a separate learning rate for weights associated with an output neuron, the output weight v_{kj} is updated according to

$$\Delta v_{kj} = -\eta_k^{out} \frac{\partial E^{out}}{\partial v_{kj}} = \eta_k^{out} \sum_{p=1}^{P} (\hat{t_k}^{(p)} - \hat{y_k}^{(p)}) h_j^{(p)} \quad (5)$$

where η_k^{out} is the learning rate for the weight vector associated with y_k. After updating the output weights v_{kj}'s, the MSE will be

$$E^{out}(\eta_k^{out}) = \frac{1}{2} \sum_{p=1}^{P} \sum_{k=1}^{M} (\hat{t_k}^{(p)} - \hat{y_k}^{(p)} + \eta_k^{out} \sum_{j=0}^{H} \frac{\partial E^{out}}{\partial v_{kj}} h_j^{(p)})^2. \quad (6)$$

If $h_j^{(p)}$ is fixed, the optimum η_k^{out} is derived by

$$\eta_k^{out} = \frac{\sum_{j=0}^{H} (\frac{\partial E^{out}}{\partial v_{kj}})^2}{\sum_{p=1}^{P} (\sum_{j=0}^{H} \frac{\partial E^{out}}{\partial v_{kj}} h_j^{(p)})^2}, k = 1, 2, \ldots, M, \quad (7)$$

under the condition that $\partial E^{out}(\eta_k^{out}) / \partial \eta_k^{out} = 0$. Thus, the output weights are updated using the optimum learning rates.

IV. Optimal Learning Rate for a Hidden Weight Vector

In this section, firstly hidden targets are assigned and a new hidden error function (E_n^{hid}) is defined based on the targets. After then, optimizing equation of hidden weights (Δw_{ji}) to minimize E_n^{hid} is derived. Finally, optimal learning rates of the EBP algorithm for hidden weights are derived by removing virtual hidden targets.

The target of $h_j^{(p)}$ is given by

$$z_j^{(p)} = h_j^{(p)} + \zeta_p \beta_j^{(p)}, \quad (8)$$

where $z_j^{(p)}$ denotes the hidden target and

$$\beta_j^{(p)} \equiv -\frac{\partial E^{out}}{\partial h_j^{(p)}} = \sum_{k=1}^{M} (\hat{t_k}^{(p)} - \hat{y_k}^{(p)}) v_{kj}. \quad (9)$$

Also ζ_p is the learning rate for assigning hidden targets. If v_{kj} is fixed, the MSE of output layer is a quadratic functional of ζ_p and the optimum ζ_p can be derived under the condition that $\partial E^{out} / \partial \zeta_p = 0$. That is, the optimum ζ_p is derived by

$$\zeta_p = \frac{\sum_{j=1}^{H} (\beta_j^{(p)})^2}{\sum_{k=1}^{M} (\sum_{j=1}^{H} v_{kj} \beta_j^{(p)})^2}, p = 1, 2, \ldots, P. \quad (10)$$

After assigning the hidden targets with the optimum ζ_p's, it is truncated to satisfy $-1 < z_j^{(p)} < 1$ and then the target of net-input to $h_j^{(p)}$ is given by

$$\hat{z_j}^{(p)} = f^{-1}(z_j^{(p)}) = 2 \tanh^{-1}(z_j^{(p)}). \quad (11)$$

Based on the assigned hidden targets, a new error function for hidden layer is defined by

$$E_n^{hid} = \frac{1}{2} \sum_{p=1}^{P} \sum_{j=1}^{H} (\hat{z_j}^{(p)} - \hat{h_j}^{(p)})^2 [f'(\hat{z_j}^{(p)})]^2. \quad (12)$$

In this error function, the second term related to the slope of $z_j^{(p)}$ controls the updating amount of weights according to whether $\hat{z_j}^{(p)}$ is in the linear region or saturation region of sigmoid function. If $\hat{z_j}^{(p)}$ and $\hat{h_j}^{(p)}$ are in the same saturation region, the difference between $z_j^{(p)}$ and $h_j^{(p)}$ is small although the one between $\hat{z_j}^{(p)}$ and $\hat{h_j}^{(p)}$ is large. In this case, it is not adequate to change the associated weights according to the large difference between $\hat{z_j}^{(p)}$ and $\hat{h_j}^{(p)}$. When $\hat{z_j}^{(p)}$ is in the linear region, on the contrary, the associated weights need to be changed according to the difference between $\hat{z_j}^{(p)}$ and $\hat{h_j}^{(p)}$.

For minimizing E_n^{hid}, the hidden weight w_{ji} is updated by

$$\Delta w_{ji} = -\eta_j^{hid} \frac{\partial E_n^{hid}}{\partial w_{ji}}, \quad (13)$$

where η_j^{hid} is the learning rate and

$$\frac{\partial E_n^{hid}}{\partial w_{ji}} = -\sum_{p=1}^{P} (\hat{z_j}^{(p)} - \hat{h_j}^{(p)}) [f'(\hat{z_j}^{(p)})]^2 x_i^{(p)}. \quad (14)$$

Substituting $w_{ji} + \Delta w_{ji}$ into Eq. (12), the error will be

$$E_n^{hid}(\eta_j^{hid}) = \frac{1}{2}\sum_{p=1}^{P}\sum_{j=1}^{H}(\hat{z_j}^{(p)} - \hat{h_j}^{(p)} + \eta_j^{hid} \sum_{i=0}^{N}\frac{\partial E_n^{hid}}{\partial w_{ji}}x_i^{(p)})^2[f'(\hat{z_j}^{(p)})]^2. \quad (15)$$

And the optimum $\eta_j^{hid}(j = 1, 2, \ldots, H)$ is derived by

$$\eta_j^{hid} = \frac{\sum_{i=0}^{N}(\frac{\partial E_n^{hid}}{\partial w_{ji}})^2}{\sum_{p=1}^{P}(\sum_{i=0}^{N}\frac{\partial E_n^{hid}}{\partial w_{ji}}x_i^{(p)})^2[f'(\hat{z_j}^{(p)})]^2} \quad (16)$$

under the condition that $\partial E_n^{hid}(\eta_j^{hid})/\partial \eta_j^{hid} = 0$.

Now, we will remove the virtual targets in the updating equations of hidden weights. For small values of $\zeta_p\beta_j^{(p)}$ in Eq. (8), the target of $\hat{h_j}^{(p)}$ in Eq. (11) can be approximated by a first order Taylor series:

$$\hat{z_j}^{(p)} = f^{-1}(h_j^{(p)} + \zeta_p\beta_j^{(p)}) \approx \hat{h_j}^{(p)} + \zeta_p\beta_j^{(p)}\frac{2}{1 - h_j^{(p)\,2}}. \quad (17)$$

Also under the assumption that there is no need of abrupt change on $z_j^{(p)}$, it can be approximated that

$$f'(\hat{z_j}^{(p)}) \approx f'(\hat{h_j}^{(p)}) = \frac{1 - h_j^{(p)\,2}}{2}. \quad (18)$$

By substituting (17) and (18) into (14), the derivative is approximated by

$$\frac{\partial E_n^{hid}}{\partial w_{ji}} \approx \sum_{p=1}^{P}\zeta_p\delta_j^{hid}(\mathbf{x}^{(p)})x_i^{(p)}. \quad (19)$$

where

$$\delta_j^{hid}(\mathbf{x}^{(p)}) = -\frac{\partial E^{out}}{\partial \hat{h_j}^{(p)}} = \beta_j^{(p)}f'(\hat{h_j}^{(p)}). \quad (20)$$

Therefore, Δw_{ji} using E_n^{hid} (Eq. (13)) is also approximated by

$$\Delta w_{ji} \approx \eta_j^{hid}\sum_{p=1}^{P}\zeta_p\delta_j^{hid}(\mathbf{x}^{(p)})x_i^{(p)}. \quad (21)$$

This reveals two optimal learning rates of the EBP algorithm for a hidden weight vector, one (ζ_p) for assigning virtual hidden targets associated with each training pattern and the other (η_j^{hid}) for minimizing the new hidden error function. The hidden weights are updated according to the above Eqs. (19) and (21) without assigning hidden targets. That is, the hidden targets are only virtual ones to derive the optimal learning rates using the new hidden error function.

If MSE is used as an hidden error function to update Δw_{ji}, it can be approximated that

$$\Delta w_{ji} \approx \eta_j^{hid}\sum_{p=1}^{P}\frac{\zeta_p}{f'(\hat{h_j}^{(p)})^2}\delta_j^{hid}(\mathbf{x}^{(p)})x_i^{(p)}. \quad (22)$$

In this case, the slope term in the denominator may make the updating amount of hidden weights very large during the learning process and finally the hidden neurons may be heavily saturated.

The proposed gradient descent learning algorithm with optimal learning rates are summarized in the next three steps.

- Step 1: Update output weights with η_k^{out} as in section III under the assumption that hidden weights are fixed.
- Step 2: With the updated output weights, calculate $\beta_j^{(p)}$ (Eq. (9)), ζ_p (Eq. (10)), and $\partial E_n^{hid}/\partial w_{ji}$ (Eq. (19)).
- Step 3: Update hidden weights according to Eq. (21) with η_j^{hid} (Eq. (16)).

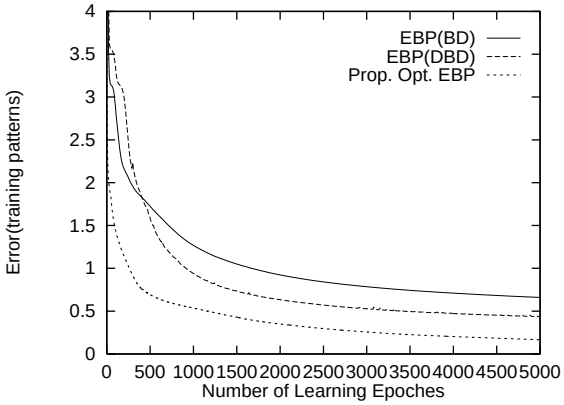
In the proposed algorithm, Δv_{kj} and Δw_{ji} are conducted alternatively since they are optimal under the condition that each other is fixed. If all $h_j^{(p)}$'s are saturated, the proposed method does not work. However, this will not occur in real problems since hidden neurons are trained not to be saturated but to extract near-orthogonal features of input patterns.

V. Simulation

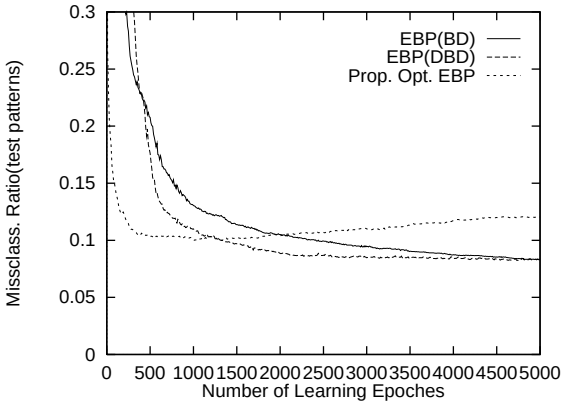
A handwritten digit recognition problem was used to verify the effectiveness of the proposed optimal learning rates. A total of 5,000 handwritten digitized images from the CEDAR database were used for training after size normalization[8]. A digit consisted of 12×12 pixels and each pixel took an integer values from zero to 15. The MLP consisted of 144 inputs, 30 hidden neurons, and ten output neurons.

We simulated the EBP algorithm with the BD technique[5], one also with the DBD rule, and the proposed gradient descent algorithm with optimal learning rates. The BD technique of the EBP algorithm took the initial learning rate of 0.05 with $\rho = 1.1$ and $\sigma = 0.5$. The DBD parameters were $\theta = 0.9, \phi = 0.01$, and $\kappa = 1 \times 10^{-7}$. And η_k^{out} in the proposed method is restricted within one at the first and the second epoches to prevent that output weights become very large to minimize MSE while hidden neurons are near zero. Four simulations were conducted using each method with initial weights drawn at random from a uniform distribution on $[-1 \times 10^{-4}, 1 \times 10^{-4}]$, and the results were averaged to draw figures.

Fig. 1(a) shows the MSE for the training patterns in each training methods. The BD and DBD methods



(a)MSE for training patterns



(b)Misclassification ratio for test patterns

Fig. 1. Simulation Results in the handwritten digit recognition task.

show slow convergence since their learning rates are not optimum. Contrary to the above result, the proposed method displays very fast decreasing of MSE since it is the EBP algorithm with optimal learning rates. Especially, the MSE drastically decreases in the initial stage of learning. Fig. 1(b) shows the misclassification ratio for 2,213 test patterns.

For more verification of the proposed method, an isolated-word recognition task was simulated, in which the vocabulary consisted of 50 words and 900 patterns were used for training MLP with 50 hidden neurons after extracting the ZCPA feature of 1,024 dimensions[9]. Fig. 2 shows the misclassification ratio for 1,050 test patterns in this task. The proposed method shows faster training speed than BD and DBD with better recognition ratio.

VI. Conclusion

To accelerate the EBP algorithm, this paper proposed optimal learning rates for each neuron and training pattern. The optimal learning rate for weights associated with an output neuron was introduced under the assumption that hidden weights are fixed. Especially, two

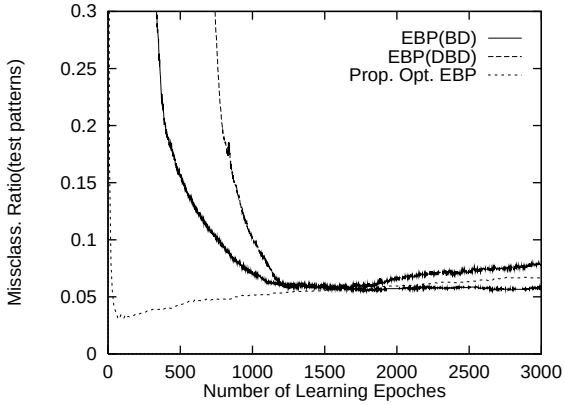


Fig. 2. Misclassification ratio for test patterns in the isolated-word recognition task.

optimal learning rates were derived for updating a hidden weight vector. One was for assigning virtual hidden targets and the other for minimizing the proposed hidden error function.

In the simulation of a handwritten digit recognition and an isolated-word recognition tasks, the proposed method showed faster learning speed than the BD and DBD without sacrificing generalization performance.

Acknowledgement: This research was funded by the Brain Science & Engineering Research Program (the Ministry of Science and Technology) in Korea.

References

[1] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing*. MIT Press, Cambridge, MA, 1986.

[2] S.-H. Oh, "Improving the error backpropagation algorithm with a modified error function," *IEEE Trans. Neural Networks*, vol. 8, no. 3, pp. 799–803, 1997.

[3] R. Parisi, E. D. Di Claudio, G. Orlandi, and B. D. Rao, "A generalized learning paradigm exploiting the structure of feedforward neural networks," *IEEE Trans. Neural Networks*, vol. 7, pp. 1450–1459, 1996.

[4] G.-J. Wang and C.-C. Chen, "A fast multilayer neural networks training algorithm based on the layer-by-layer optimizing procedures," *IEEE Trans. Neural Networks*, vol. 7, pp. 768–775, 1996.

[5] T. P. Vogl, J. K. Mangis, A. K. Rigler, W. T. Zink, and D. L. Alkon, "Accelerating the convergence of the back-propagation method," *Biol. Cybern.*, vol. 59, pp. 257–263, 1988.

[6] R. A. Jacobs, "Increased rates of convergence through learning rate adaptation," *Neural Networks*, vol. 1, pp. 295–307, 1988.

[7] S. E. Fahlman, "Faster-learning variations on back-propagation: an empirical study," *Proc. of the 1988 Connectionist Models Summer School*, pp. 38–51, 1988.

[8] J. J. Hull, "A database for handwritten text recognition research," *IEEE Trans. Pat. Ana. Mach. Int.*, vol. 16, no. 5, pp. 550–554, May 1994.

[9] D.-S. Kim, S.-Y. Lee, and R. M. Kil, "Auditory processing of speech signals for robust speech recogtion in real-world noisy environments," *IEEE Trans. Speech and Audio Processing*, vol. 7, pp. 55–69, 1999.