An Improved Backpropagation Algorithm Using Absolute Error Function*

Jiancheng Lv and Zhang Yi

Computational Intelligence Laboratory, School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chegndu 610054, China
zhangyi@uestc.edu.cn, jeson_cha@yahoo.com
http://cilab.uestc.edu.cn

Abstract. An improved backpropagation algorithm is proposed by using the Lyapunov method to minimize the absolution error function. The improved algorithm can make both error and gradient approach zero so that the local minima problem can be avoided. In addition, since the absolute error function is used, this algorithm is more robust and faster for learning than the backpropagation with the traditional square error function when target signals include some incorrect data. This paper also proposes a method of using Lyapunov stability theory to derive a learning algorithm which directly minimize the absolute error function.

1 Introduction

The backpropagation algorithm [1] for training the multi-layer feed-forward neural network is a very popular learning algorithm. The multilayer neural networks have been used in many fields such as pattern recognition, image processing, classification, diagnosis and so on. But the standard backpropagation encounters two problems in practice, i.e., slow convergence and local minima problem. Several improvements have been proposed to overcome the problems [4], [5], [6], [7], [10], [11], [12].

To accelerate the convergence and escape from the local minima, this paper proposes an improved algorithm that makes the error and gradient go to zero together. This algorithm is obtained by minimizing the absolute error function directly based on Lyapunov stability theory. Usually, the standard backpropagation algorithm is a steepest gradient descent algorithm based on the square error function[2]. The algorithm iteratively and gradually modifies parameters according to the deviation between an actual output and target signal. The square error function is used to calculate the deviation. To minimizing the deviation, the absolute error may sometimes be better than the square error. In fact, the absolute error function is less influenced by anomalous data than the square error function[3]. So, the algorithms with the absolute error function are

* This work was supported by National Science Foundation of China under Grant 60471055 and Specialized Research Fund for the Doctoral Program of Higher Education under Grant 20040614017.

J. Wang, X. Liao, and Z. Yi (Eds.): ISNN 2005, LNCS 3496, pp. 585–590, 2005.

more robust and learns faster than the backpropagation with the square error function when target signals include some incorrect data[3]. In [3], K. Taji et al took the differentiable approximate function for the absolute error as the objective function. However, in this paper, the absolute error will directly be used to calculate the backpropagation error. At the same time, the improved algorithm based on Lyaponuv stability theory is able to make the error and gradient go to zero together so that the evolution can escape from the local minima.

The rest part of this paper is organized as follows. In Section 2, this improved backpropagation algorithm is obtained by using Lyapunov method to minimize the absolute error function. In Section 3, the simulation are carried out to illustrate the result. Finally, conclusions is described in Section 4.

2 An Improved BP Algorithm

Consider a multiplayer feed-forward neural network with L layers and N output neurons (Fig. 1). As for the ith neuron in the l layer, let a_i^l be its output and n_i^l is its net input. It follows that

$$a_i^l = f_i^l(n_i^l), \tag{1}$$

$$n_i^l = \sum_{j=1}^{N_{l-1}} w_{ij}^l a_j^{l-1}, \tag{2}$$

where N_{l-1} is the number of neurons in $l-1$ layer.

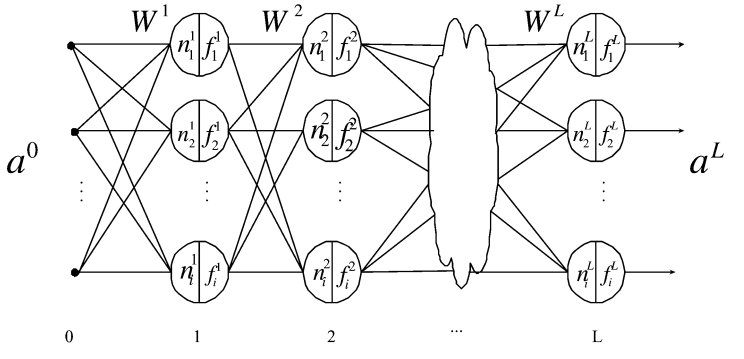

Fig. 1. An L layers BP Neural Network with N output neurons

Suppose the training data set consists of p patterns. For a specific pattern, the network outputs $\mathbf{y}_i$ and targets $\mathbf{y}_i^d$ are wrote that

$$\{\mathbf{y}^d, \mathbf{y}\} = \left\{ \begin{pmatrix} y_1^d \\ y_2^d \\ \vdots \\ y_N^d \end{pmatrix}, \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix} \right\}. \tag{3}$$

The absolute error function is constructed as the cost function, it follows that

$$E = \sum_{p} \sum_{q=1}^{N} |y_q^d - y_q|. \tag{4}$$

We consider a Lyapunov function that

$$V(t) = \sum_{p} \sum_{q=1}^{N} |y_q^d(t) - y_q(t)|. \tag{5}$$

The time derivative of the Lyapunov function is given by

$$\begin{aligned} \frac{dV(t)}{dt} = \frac{\partial V(t)}{\partial w_{ij}} \cdot \frac{dw_{ij}}{dt} &= \sum_{p} \left[\sum_{q=1}^{N} (sign(y_q^d(t) - y_q(t))) \frac{\partial (y_q^d(t) - y_q(t))}{\partial w_{ij}} \frac{dw_{ij}}{dt} \right] \\ &= -\sum_{p} \left[\sum_{q=1}^{N} (sign(y_q^d(t) - y_q(t))) \frac{\partial y_q(t))}{\partial w_{ij}} \frac{dw_{ij}}{dt} \right] \end{aligned} \tag{6}$$

Theorem 1. *Given any initial value $w_{ij}(0)$, if the wight is updated by*

$$w_{ij}(k) = w_{ij}(0) + \int_0^k \frac{dw_{ij}}{dt} dt, \tag{7}$$

where

$$\frac{dw_{ij}}{dt} = \frac{y_q^d(t) - y_q(t)}{\dfrac{\partial y_q(t)}{\partial w_{ij}}}, \tag{8}$$

then E goes to zero along the trajectory of the weight w_{ij}.

Poof: From (6) and (8), it gives that

$$\frac{dV(t)}{dt} = -2V(t) \tag{9}$$

then

$$V(t) = V(0)e^{-2t}. \tag{10}$$

Clearly, $V(t) \to 0$ as $t \to 0$, i.e. E goes to zero with exponential rate. The proof is completed.

According to the equation (8), the weight update equation in the l layer can be wrote as follows:

$$w_{ij}^l(k+1) = w_{ij}^l(k) + \eta \frac{y^d - y}{\dfrac{\partial y}{\partial w_{ij}^l}}. \tag{11}$$

By chain rule, if l is the last layer L, it holds that

$$\frac{\partial y}{\partial w_{ij}^l} = \frac{\partial f_i^l(n_i^l)}{\partial n_i^l} \cdot \frac{\partial n_i^l}{\partial w_{ij}} = \dot{f}_i^l(n_i^l) \cdot a_{ij}^l. \tag{12}$$

Otherwise, we have

$$\begin{aligned} \frac{\partial y}{\partial w_{ij}^l} &= \sum_{k=1}^{N_{l+1}} \left(\frac{\partial f_k^{l+1}(n_k^{l+1})}{\partial n_k^{l+1}} \cdot \frac{\partial n_k^{l+1}}{\partial n_k^l} \cdot \frac{\partial n_k^l}{\partial w_{ij}}\right) \\ &= \sum_{k=1}^{N_{l+1}} \dot{f}_k^{l+1}(n_k^{l+1}) \cdot \dot{f}_i^l(n_i^l) w_{ki}^{l+1} \cdot a_{ij}^l \\ &= \dot{f}_i^l(n_i^l) \cdot a_{ij}^l \sum_{k=1}^{N_{l+1}} \dot{f}_k^{l+1}(n_k^{l+1}) \cdot w_{ki}^{l+1}, \end{aligned} \tag{13}$$

where N_{l+1} is the number of the $l+1$ layer. So, from (11), (12) and (13), the improved algorithm can be described as follows:

$$\begin{cases} w_{ij}^l(k+1) = w_{ij}^l(k) + \dfrac{\eta}{\dot{f}_i^l(n_i) \cdot a_{ij}^l} \cdot (y^d - y), \quad l = L \\ w_{ij}^l(k+1) = w_{ij}^l(k) + \dfrac{\eta}{\dot{f}_i^l(n_i) \cdot a_{ij}^l \cdot \sum_{p=1}^{N_{l+1}} \dot{f}_p^{l+1}(n_p^{l+1}) w_{pi}^{l+1}} \cdot (y^d - y), otherwise, \end{cases} \tag{14}$$

where N_{l+1} is the number of the $l+1$ layer. In addition, to avoid the algorithm becomes unstable when the gradient goes to zero, a small constant is added to the denominator. From (14), it is clear that the evolution stops only when the error goes to zero. If the gradient goes to zero and the error is still large, a large update value makes the evolution escape from the minima.

3 Simulations and Discussions

This improved algorithm for multilayer networks makes the weight evolution escape from the local minima. The XOR problem experiment will be carried out to illustrate it. A simple architecture(2-2-1) is used to learn the problem. The bipolar sigmoid function is taken as the active function. To avoid the active function goes to the saturation[4], [5], [8], [9], the parameter η should is a relatively small constant. And, once the update value is enough large so that the evolution goes to premature saturation, the update value must be scaled down to avoid the premature saturation.

Fig. 2 and Fig. 3 show the experiment result with 500 epochs. On the left picture in Fig. 3, the 60 sample points are divide into two class ones by two decision lines. The class 0 is presented by star points and the class 1 is presented

by circinal points. The Fig. 2 shows the error evolution. It is clear the evolution escapes from the local minima. Finally, the 200 random test points are provided for the network. Obviously, these points have been divided rightly on the right one in Fig. 3.

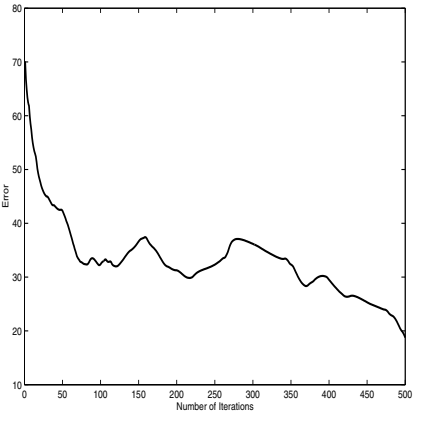

Fig. 2. The error evolution

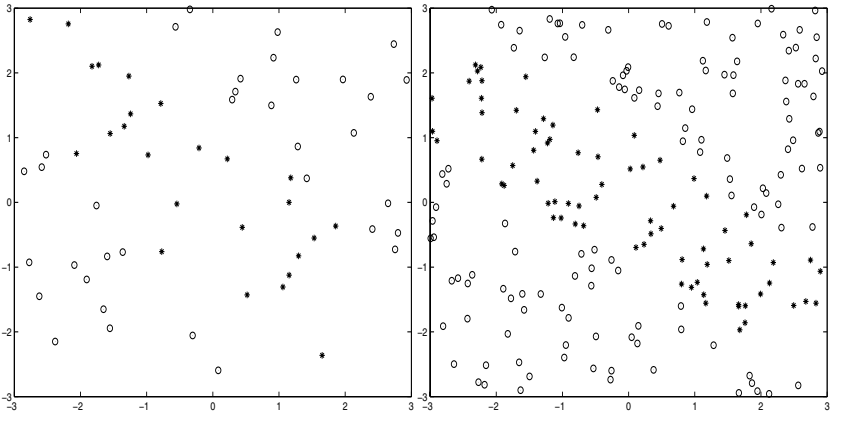

Fig. 3. The samples (left) and test result (right)

4 Conclusions

This paper gives an improved learning method for multilayer neural networks to escape local minima. This algorithm is obtained by using Lyapunov method to minimize the absolute error function. Since the algorithm makes the error and gradient go to zero together, the local minima problem could be avoided. The

algorithm is more robust since it uses the absolute error function. The simulation for XOR problem shows the effectiveness of the proposed algorithm.

References

1. Werbos, P. J.: The Roots of Backpropagation. Wiley, New York (1994)
2. Rumelhart, D.E., Hinton, G.E. Williams, R.J.: Learning Internal Representations by Error Propagation. Parallel distributed Processing, MIT Press (1986) 318-362
3. Taji, K., Miyake, T., Tarmura, H.: On Error Backpropagation Algorithm Using Absolute Error Function. IEEE SMC '99 Conference Proceedings, **5** (1999) 12-15
4. Wang, X.G., Tang, Z., Tamura, H., Ishii, M.: A Modified Error Function for the Backpropagation Algorithm. Neurocomputing, **57** (2004) 477-484
5. Wang, X.G., Tang, Z., Tamura, H., Ishii, M., Sum, W.D.: An Improved Backpropagation Algorithm to Avoid the Local Minima Problem. Neurocomputing, **56** (2004) 455-460
6. Owen, C.B., Abunawass, A.M.: Application of Simulated Annealing to the Backpropagation Model Improves Convergence. Proceeding of the SPIE Conference on the Science of Artificial Neural Networks, **II** (1993) 269-276
7. Von Lehmen, A., Paek, E.G., Liao, P.F., Marrakchi, A., Patel, J.S.: Factors Influencing Learning by Backpropagation. Proceedings of the IEEE International Conference On Neural Networks, **I** (1988) 335-341
8. Fukuoka, Y., Matsuki, H., Minamitani, H., Ishida, A.: A Modified Backpropagation Method to Avoid False Local Minima. Neural Networks, **11** (1998) 1059-1072
9. Vitela, J. E., Reifman, J.: Premature Saturation in Backpropagation Networks: Mechanism and Necessary Conditions. Neural Networks, **10** (1997) 721-735
10. Wand, C., Principe, J. C.: Training Neural Networks with Additive Noise in the Desired Signal. IEEE Trans. Neural Networks, **10** (1999) 1511-1517
11. Battiti, R., Masulli, F.: BFGS Optimization for Faster and Automated Supervised Learning. Proceedings of the Internatioanl Neural Network Conference. Kluwer, paris, France (1990) 757-760
12. Kollias, S., Anastassiou, D.: An Adaptive Least Squares Algorithm for Efficient Training of Artificial Neural Networks. IEEE Trans. on Circuits and systems, **36** (1989) 1092-1101