A BACKPROPAGATION LEARNING FRAMEWORK FOR FEEDFORWARD NEURAL NETWORKS

Xinghuo Yu[1]*, M. Onder Efe*[2]*, Okyay Kaynak*[2]
[1]Faculty of Informatics and Communication, Central Queensland University
Rockhampton QLD 4702 Australia
[2]Department of Electrical and Electronic Engineering, Bogazici University
Bebek, 80815, Istanbul, Turkey

ABSTRACT

In this paper, a general backpropagation learning framework for the training of feedforward neural networks is proposed. The convergence to global minimum under the framework is investigated using the Lyapunov stability theory. It is shown the existing feedforward neural networks training algorithms are special cases of the proposed framework.

1. INTRODUCTION

Feedforward neural networks (FNN) have been widely used for various tasks, such as pattern recognition, function approximation, dynamical modeling, data mining, time series forecasting, to name just a few. [1,2]. The training of feedforward neural networks is mainly undertaken using the back-propagation (BP) based learning algorithms. A number of different kinds of BP learning algorithms have been proposed, such as an online neural network learning algorithm for dealing with time varying inputs [3], fast learning algorithms based on gradient descent of neuron space [4], and the Levenberg-Marquardt algorithm [5,6]. Different merits of their effectiveness have been observed.

In this paper, we develop a general BP learning framework for FNN training with time varying inputs. The Lyapunov theory is used to prove the convergence of the algorithm to the global minimum.

2. THE GENERAL FRAMEWORK

Before we proceed, denote the inputs, weights, desired outputs and actual outputs of the feedforward neural networks as

$$x(t)=(x_1,x_2,\cdots,x_n)^T\in R^n \qquad (1)$$

$$\phi(t)=(\phi_1,\phi_2,\cdots,\phi_l)^T\in R^l \qquad (2)$$

$$y_d(t)=(y_{d1},y_{d2},\cdots,y_{dm})^T\in R^m \qquad (3)$$

$$y(t)=(y_1,y_2,\cdots,y_m)^T\in R^m \qquad (4)$$

where $x(t)$ is the input vector, $\phi(t)$ the weight vector, $y_d(t)$ the desired output vector and $y(t)$ the output vector of the neural network respectively. The error at any instant is represented as

$$J=\varepsilon(t)=\frac{1}{2}(y(t)-y_d(t))^T(y(t)-y_d(t))=\frac{1}{2}\left\|y(t)-y_d(t)\right\|^2 \qquad (5)$$

where the symbol 'T' represents the transpose. Note that here the input $x(t)$ is of a general type, and it can be discrete, continuous and time varying. The weight vector $\phi(t)$ represents weights for perceptrons as well as multi-layer FNN.

We now investigate the convergence issue in the learning process of FNN. Most of BP learning algorithms can be considered as finding zeros of $\partial J/\partial\phi$ which correspond to (possibly local) minima. The search performance of this class of learning algorithms somehow relies on initial values of weights, and oftentimes, it traps into local minima.

To develop the general learning framework, we now make use of the Lyapunov theory [7, 9]. First, choose the positive definite Lyapunov function with respect to J and $\partial J/\partial\phi$

$$V=\mu J+\frac{1}{2}\sigma\left\|\frac{\partial J}{\partial\phi}\right\|^2, \qquad (6)$$

where the parameters μ,σ determine the relative importance of each term and

$$\left\|\frac{\partial J}{\partial\phi}\right\|^2=\left(\frac{\partial J}{\partial\phi}\right)\left(\frac{\partial J}{\partial\phi^T}\right)$$

where $\frac{\partial J}{\partial\phi}=(\frac{\partial J}{\partial\phi_1},\cdots,\frac{\partial J}{\partial\phi_l})$ is the gradient represented in a row vector form [7]. For convenience, we also denote $\frac{\partial J}{\partial\phi^T}=(\frac{\partial J}{\partial\phi_1},\cdots,\frac{\partial J}{\partial\phi_l})^T$. The purpose of selecting the Lyapunov function (6) is that, by finding an appropriate learning algorithm represented by $\dot{\phi}$, the positive definite function V is minimized to reach its global minimum

0-7803-6685-9/01/$10.00©2001 IEEE

$$J = 0 \text{ and } \frac{\partial J}{\partial \phi} = 0 \tag{7}$$

which corresponds the global minimum. The question is how. From the Lyapunov theory [5,9], if we can develop a learning algorithm for updating the weights ϕ so that

$$\dot{V} < 0 \tag{8}$$

that is $\dot{V}$ is negative definite with respect to J and $\partial J / \partial \phi$, then the equilibrium of $V = 0$, which corresponds to the global minimum (9), will be reached asymptotically. The time derivative of the Lyapunov function V is derived as

$$\begin{aligned}
\dot{V} &= \mu\left[\frac{\partial J}{\partial \phi}\dot{\phi} + \frac{\partial J}{\partial x}\dot{x} + \frac{\partial J}{\partial t}\right] + \\
&\quad \sigma \frac{\partial J}{\partial \phi}\left(\frac{\partial^2 J}{\partial t \partial \phi^T} + \frac{\partial^2 J}{\partial \phi \partial \phi^T}\dot{\phi} + \frac{\partial^2 J}{\partial x \partial \phi^T}\dot{x}\right) \\
&= \left(\mu \frac{\partial J}{\partial \phi} + \sigma \frac{\partial J}{\partial \phi}\frac{\partial^2 J}{\partial \phi \partial \phi^T}\right)\dot{\phi} + \mu \frac{\partial J}{\partial t} + \mu \frac{\partial J}{\partial x}\dot{x} \\
&\quad + \sigma \frac{\partial J}{\partial \phi}\frac{\partial^2 J}{\partial \phi \partial t} + \sigma \frac{\partial J}{\partial \phi}\frac{\partial^2 J}{\partial x \partial \phi^T}\dot{x} \\
&= \frac{\partial J}{\partial \phi}\left(\mu I_l + \sigma \frac{\partial^2 J}{\partial \phi \partial \phi^T}\right)\dot{\phi} + \mu \frac{\partial J}{\partial t} + \mu \frac{\partial J}{\partial x}\dot{x} + \\
&\quad \sigma \frac{\partial J}{\partial \phi}\frac{\partial^2 J}{\partial t \partial \phi^T} + \sigma \frac{\partial J}{\partial \phi}\frac{\partial^2 J}{\partial x \partial \phi^T}\dot{x}
\end{aligned} \tag{9}$$

It is evident that if the general learning framework (algorithm) for $\dot{\phi}$ is designed as

$$\begin{aligned}
\dot{\phi} = -\left(\mu + \sigma \frac{\partial^2 J}{\partial \phi \partial \phi^T}\right)^{-1}\left(\frac{\dfrac{\partial J}{\partial \phi^T}}{\left\|\dfrac{\partial J}{\partial \phi}\right\|^2}\right)\Bigg(\mu \frac{\partial J}{\partial t} + \mu \frac{\partial J}{\partial x}\dot{x} + \sigma \frac{\partial J}{\partial \phi}\frac{\partial^2 J}{\partial t \partial \phi^T} + \\
\sigma \frac{\partial J}{\partial \phi}\frac{\partial^2 J}{\partial x \partial \phi^T}\dot{x} + \varsigma\left\|\frac{\partial J}{\partial \phi}\right\|^2 + \eta\|J\|^2\Bigg)
\end{aligned} \tag{10}$$

with $\varsigma > 0, \eta > 0$, then

$$\dot{V} = -\varsigma\left\|\frac{\partial J}{\partial \phi}\right\|^2 - \eta\|J\|^2 < 0 \tag{11}$$

which is negative definite with respect to J and $\partial J / \partial \phi$. According to the LaSalle-Yoshizawa theorem [9, p. 24], the negative definiteness of $\dot{V}$ indicates that the learning algorithm (10) will lead the weights in $\phi(t)$ to converge asymptotically to the values of weights such that $\left\|\frac{\partial J}{\partial \phi}\right\| = 0$ and $\|J\| = 0$, which corresponds to the global minimum. Hence the global minimum is achieved under the learning rule (10).

Note that conditions $\left\|\frac{\partial J}{\partial \phi}\right\| = 0$ and $\|J\| = 0$ do not necessarily warrant a unique solution of $\phi(t)$, rather correspond to a set of solutions $\phi(t)$ which can make (9) valid. Note also that $\left\|\frac{\partial J}{\partial \phi}\right\| = 0$ would impose a singularity in (10). One way to avoid is to set $\dot{\phi} = 0$ when $\left\|\frac{\partial J}{\partial \phi}\right\| = 0$. The parameters $\varsigma > 0, \eta > 0$ will determine the convergence speed of $\dot{V}$.

The following theorem as the main result of this paper summarizes the above discussion.

Theorem: For a FNN structure whose input-output relationship is $y(t) = f(t, \phi(t), x(t))$, with the general learning framework (10), the J converges to zero asymptotically and the global minimum of J is achieved.

The Theorem can interpret many existing BP learning algorithms. Training FNN with discrete input data set is a quite common learning task. In this case, we have

$$\frac{\partial J}{\partial t} = 0, \quad \dot{x} = 0, \quad \frac{\partial^2 J}{\partial t \partial \phi^T} = 0 \tag{12}$$

The learning algorithm for discrete data set is generally expressed as

$$\phi(k+1) = \phi(k) - \Delta t \dot{\phi}$$

where the term $\dot{\phi}$ acts as a “gradient” and we will demonstrate how to derive several commonly used learning algorithms.

The conventional gradient descent learning algorithm can be easily obtained by setting $\sigma = 0$, $\eta = 0$ and using (10). Since $\frac{\partial^2 J}{\partial t \partial \phi^T} = 0$ and $\dot{x} = 0$, then we have

$$\dot{\phi} = -\mu^{-1}\varsigma \frac{\partial J}{\partial \phi^T} = -\lambda \frac{\partial J}{\partial \phi^T}, \ \lambda = \mu^{-1}\varsigma .$$

The Gauss-Newton algorithm can be obtained by setting $\mu = 0$, $\eta = 0$ and using (10). Since $\frac{\partial J}{\partial t} = 0$, $\frac{\partial^2 J}{\partial t \partial \phi^T} = 0$ and $\dot{x} = 0$, then we have

$$\dot{\phi} = -\left(\sigma \frac{\partial^2 J}{\partial\phi\partial\phi^T}\right)^{-1}\left(\varsigma \frac{\partial J}{\partial\phi^T}\right) = -\lambda\left(\frac{\partial^2 J}{\partial\phi\partial\phi^T}\right)^{-1}\left(\frac{\partial J}{\partial\phi^T}\right)$$

with $\lambda = \sigma^{-1}\varsigma$

The Levenberg-Marquardt algorithm can be easily obtained by setting $\eta = 0$ and using (10). Since $\frac{\partial J}{\partial t} = 0$, $\frac{\partial^2 J}{\partial t\partial\phi^T} = 0$ and $\dot{x} = 0$, then we have

$$\dot{\phi} = -\left(\mu + \sigma \frac{\partial^2 J}{\partial\phi\partial\phi^T}\right)^{-1}\left(\varsigma \frac{\partial J}{\partial\phi^T}\right).$$

If we assume $\eta = 0$ and $\mu = 0$ but $x(t)$ is a time varying function, then we will end up with the online learning algorithm [3] which gives rise to an exponentially convergent learning.

Note that in the above derivations, $\eta = 0$ is assumed. This implies that the Lyapunov function becomes $V = \frac{1}{2}\sigma\left\|\frac{\partial J}{\partial\phi}\right\|^2$ hence minimization of this Lyapunov function only result in possible local minima as $\partial J / \partial\phi = 0$ does not necessarily determine the global minimum.

Furthermore, if we design the learning algorithm as

$$\dot{\phi} = -\left(\mu + \sigma\frac{\partial^2 J}{\partial\phi^2}\right)^{-1}\left(\frac{\frac{\partial J}{\partial\phi^T}}{\left\|\frac{\partial J}{\partial\phi}\right\|^2}\right)\left(\mu\frac{\partial J}{\partial t} + \mu\frac{\partial J}{\partial x}\dot{x} + \right.$$
$$\sigma\frac{\partial J}{\partial\phi}\frac{\partial^2 J}{\partial t\partial\phi^T} + \sigma\frac{\partial J}{\partial\phi}\frac{\partial^2 J}{\partial x\partial\phi^T}\dot{x} +$$
$$\varsigma\sqrt{\mu J + \frac{1}{2}\sigma\left\|\frac{\partial J}{\partial\phi}\right\|^2}$$

then we will end up with

$$\dot{V} < -\varsigma V^{1/2},\ \varsigma > 0$$

which will enable a finite time convergent learning.

3. CONCLUSION

A general framework for FNN training algorithms has been proposed. Its convergence to the global minimum has been proved using the Lyapunov theory. It has been shown that several commonly used BP learning algorithms are a special case of the general BP learning algorithm. However, the strength of the algorithm lays in its ability to handle any time varying inputs.

References

1) J. M. Zurada, Introduction to Artificial Neural Systems, West Publishing, 1992.
2) P. Mehra and B. W. Wah, Artificial Neural Networks: Concepts and Theory, IEEE Computer Society Press, 1992.
3) Y. Zhao, “On-line neural network learning algorithm with exponential convergence rate,” Electronic Letters, vol. 32, no. 15, pp. 1381-1382, July 1996.
4) G. Zhou and J. Si, “Advanced neural network training algorithm with reduced complexity based on Jacobian deficiency,” IEEE Trans. Neural Networks, vol. 9, no. 3, pp. 448-453, May 1998.
5) R. Parisi, E. D. Di Claudio, G. Orlandi, B. D. Rao, “A generalized learning paradigm exploiting the structure of feedforward neural networks,” IEEE Trans. Neural Networks, vol. 7, no. 6, pp. 1450-1459, November 1996.
6) M. T. Hagan and M. B. Menhaj, “Training feedforward neural networks with the Marquardt algorithm,” IEEE Trans. Neural Networks, vol. 5, no. 6, November 1994.
7) J.-J. Slotine and W. Li, Applied Nonlinear Control, Prentice Hall, Englewood Cliffs, NJ 07632, 1991.
8) H. Bersini and V. Gorrini, “A simplification of the backpropagation through time algorithm for optimal neurocontroller,” IEEE Trans. Neural Networks, vol. 8, no. 2, pp. 437-441, March 1997.
9) M. Krstic, I. Kanellakopoulos, P. Kokotovic, Nonlinear and Adaptive Control Design, Wiley Interscience, 1995.