Communications
in
Mathematical
Research
26(1)(2010), 67–75

Convergence of Online Gradient Method with Penalty for BP Neural Networks*

Shao Hong-mei[1], Wu Wei[2] and Liu Li-jun[3]

(*1. College of Mathematics and Computational Science, China University of Petroleum, Dongying, Shandong, 257061*)

(*2. Department of Applied Mathematics, Dalian University of Technology, Dalian, Liaoning, 116024*)

(*3. Department of Mathematics, Dalian Nationalities University, Dalian, Liaoning, 116605*)

Communicated by Ma Fu-ming

Abstract: Online gradient method has been widely used as a learning algorithm for training feedforward neural networks. Penalty is often introduced into the training procedure to improve the generalization performance and to decrease the magnitude of network weights. In this paper, some weight boundedness and deterministic convergence theorems are proved for the online gradient method with penalty for BP neural network with a hidden layer, assuming that the training samples are supplied with the network in a fixed order within each epoch. The monotonicity of the error function with penalty is also guaranteed in the training iteration. Simulation results for a 3-bits parity problem are presented to support our theoretical results.

Key words: convergence, online gradient method, penalty, monotonicity

2000 MR subject classification: 92B20, 68T05

Document code: A

Article ID: 1674-5647(2010)01-0067-09

1 Introduction

Online gradient method (OGM for short) is a popular and commonly used learning algorithm for training the weights of BP networks (see [1]–[5]). Penalty methods are often introduced into the training procedure and have proved efficient to improve the generalization performance and to decrease the complexity of neural networks (see [6]–[12]). Here the generalization performance refers to the capacity of a neural network to give correct outputs for untrained data. A simple and commonly used penalty added to the conventional error function is the squared penalty, a term proportional to the magnitude of the network weights

*Received date: Dec. 14, 2007.

Foundation item: The NSF (10871220) of China and the Doctoral Foundation (Y080820) of China University of Petroleum.

(see [2] and [3]). Applied to the weight updating rule of batch gradient descent algorithm, the influence of penalty on the training can be seen clearly:

$$\Delta w(n) = -\eta \frac{\partial E(w)}{\partial w(n)} - \lambda w(n) \tag{1.1}$$

where w, $\Delta w(n)$, $E(w)$, η and λ represent the vector of all weights, the modification of w at the n-th iteration, the conventional error function, the learning rate and the penalty parameter, respectively. As shown in (1.1), in addition to the update by the gradient algorithm, the weight is decreased by λ times of its old value. Consequently, the weights with small magnitudes are encouraged to decrease to zero and those with large magnitudes are constrained from growing too large during the training process. This will force the network response to be smoother and less likely to overfit, leading to good generalization (see [6], [7] and [11]). Many experiments have shown that as well as being beneficial from a generalization capacity prospective, such a term provides a way to control the magnitude of the weights during the training procedure in literature (see [6] and [10]–[12]). But there remains a lack of theoretical assurance on this experimental observation, especially for online cases.

For simplicity of analysis, the input sample ξ^j is provided to the network in a fixed order in each training epoch. We shall show that the online gradient method with penalty and fixed inputs (OGM-PF) is deterministically convergent. A boundedness theorem is established for the network weights connecting the input and hidden layers, which is also a desired outcome of adding penalty. Another key point of our proofs lies in the monotonicity of the error function with such a penalty term during the training iteration.

In this paper, $\|\cdot\|$ stands for the Euclidean norm and C_i stand for suitable positive constants which are independent of the iteration step n.

The rest of this paper is organized as follows. OGM-PF is described in detail in Section 2. The main theorems are presented in Section 3. In Section 4, the algorithm OGM-PF is applied to a 3-bits parity problem to illustrate our theoretical findings. Some lemmas and detailed proofs of the theorems are gathered as an Appendix.

2 Online Gradient Method with Penalty

Consider a three-layer BP network consisting of p input units, q hidden units and one output unit. Let $w_0 = (w_{01}, \cdots, w_{0q})$ be the weights between the hidden units and the output unit, and $w_i = (w_{i1}, \cdots, w_{ip})$ be the weights between the input units and the hidden unit i $(i = 1, 2, \cdots, q)$. To simplify the presentation, we write all the weight parameters in a compact form, i.e., $W = (w_0, w_1, \cdots, w_q) \in \mathbf{R}^{q+pq}$. And we define a matrix $V = (w_1^T, \cdots, w_q^T)^T \in \mathbf{R}^{q\times p}$ and a vector function $G(x) = (g(x_1), \cdots, g(x_q))$ for $x = (x_1, \cdots, x_q) \in \mathbf{R}^q$. Assume that $\{\xi^j, O^j\}_{j=1}^J$ is the given set of training samples and $g : \mathbf{R} \to \mathbf{R}$ is a transfer function for both hidden and output layers. Then for each input $\xi \in \mathbf{R}^p$, the actual output vector of the hidden layer is $G(V\xi)$ and the final output of the network is $\zeta = g(w_0 \cdot G(V\xi))$. A