Boundedness of a Batch Gradient Method with Penalty for Feedforward Neural Networks

HUISHENG ZHANG[1,2], WEI WU[1,2]
[1]Department of Mathematics
Dalian Maritime University
Dalian, China, 116026

MINGCHEN YAO[2]
[2]Department of Applied Mathematics
Dalian University of Technology
Dalian, China, 116023

Abstract: This paper considers a batch gradient method with penalty for training feedforward neural networks. The role of the penalty term is to control the magnitude of the weights and to improve the generalization performance of the network. An usual penalty is considered, which is a term proportional to the norm of the weights. The boundedness of the weights of the network is proved. The boundedness is assumed as a precondition in an existing convergence result, and thus our result improves this convergence result.

Key–Words: Batch gradient method; Feedforward neural network; Boundedness; Penalty

1 Introduction

We are concerned in this paper with the batch gradient method with penalty for training feedforward neural networks. The penalty term is often introduced into the network training algorithms so as to control the magnitude of the weights and to improve the generalization performance of the network [1, 2]. An usual penalty is considered, which is a term proportional to the norm of the weights. It is generally agreed, but not mathematically proved as far as we know, that the weights of the network will keep bounded in the training process. The aim of this short note is to prove the boundedness.

Wu et. al. [3] considered a batch gradient method with penalty for training feedforward neural networks, and proved a convergence theorem under a condition that the weights between the hidden and input layers were bounded during the training process. This precondition is difficult to check in practice and limits the applicability of their results. We will show that the weights are indeed bounded in this case, and hence the boundedness condition in [3] is not necessary.

The rest of this paper is organized as follows. The network model and the batch gradient method with penalty are described in the next section. A convergence result in [3] mentioned above is cited in Section 3 for comparison with our result. Section 4 presents the main results of the paper. In this paper, the notation $\|\cdot\|$ denotes the Euclidean vector norm.

2 Batch gradient method with penalty

Consider a three-layer network consisting of p input nodes, q hidden nodes, and 1 output node. Let $w_0 = (w_{01}, w_{02}, \cdots, w_{0q})^T \in \mathbb{R}^q$ be the weight vector between all the hidden units and the output unit, and $w_i = (w_{i1}, w_{i2}, \cdots, w_{ip})^T \in \mathbb{R}^p$ be the weight vector between all the input units and the hidden unit $i \quad (i = 1, 2, \cdots, q)$. To simplify the presentation, we write all the weight parameters in a compact form, i.e., $W = (w_0^T, w_1^T, \cdots, w_q^T)^T \in \mathbb{R}^{q+pq}$ and we define a matrix $V = (w_1, w_2, \cdots, w_q)^T \in \mathbb{R}^{q\times p}$. Let $g : \mathbb{R} \to \mathbb{R}$ be a transfer function for the hidden and output nodes, which is typically, but not necessarily, a sigmoid function. We define a vector function for any $x = (x_1, x_2, \cdots, x_q) \in \mathbb{R}^q$ as follows

$$G(x) = \big(g(x_1), g(x_2), \cdots, g(x_q)\big)^T. \tag{1}$$

For an input vector ξ, the output of the network is

$$\zeta = g(w \cdot G(V\xi)). \tag{2}$$

Suppose that $\{\xi^j, O^j\}_{j=1}^J \subset \mathbb{R}^p \times \mathbb{R}$ is a given set of training samples. The error function with penalty (see [1, 3, 5]) is defined as

$$\begin{aligned} E(W) &= \frac{1}{2}\sum_{j=1}^{J}\big(O^j - g(w_0 \cdot G(V\xi^j))\big)^2 + \lambda\|W\|^2 \\ &= \sum_{j=1}^{J} g_j\left(w_0 \cdot G(V\xi^j)\right) + \lambda\|W\|^2, \end{aligned} \tag{3}$$

where $g_j(t) := \frac{1}{2}\big(O^j - g(t)\big)^2$. The gradient of the error function is given by

$$E_W(W) = \big(E_{w_0}^T(W), E_{w_1}^T(W), \cdots, E_{w_q}^T(W)\big)^T \tag{4}$$

with

$$E_{w_0}(W) = \sum_{j=1}^{J} g_j'\big(w_0 \cdot G(V\xi^j)\big)G(V\xi^j) + 2\lambda w_0,$$

$$E_{w_i}(W) = \sum_{j=1}^{J} g_j'\big(w_0 \cdot G(V\xi^j)\big)w_{0i}g'(w_i \cdot \xi^j)\xi^j + 2\lambda w_i, \quad i = 1, 2, \cdots, q.$$

Starting from an arbitrary initial value W^0, the weights $\{W^n\}$ are updated iteratively by (cf. [3, 6])

$$W^{n+1} = W^n + \Delta W^n, \quad n = 0, 1, 2, \cdots \tag{5}$$

with

$$\Delta w_0^n = -\eta\big[\sum_{j=1}^{J} g_j'(w_0^n \cdot G(V^n\xi^j))G(V^n\xi^j) + 2\lambda w_0^n\big],$$

$$\Delta w_i^n = -\eta\big[\sum_{j=1}^{J} g_j'(w_0^n \cdot G(V^n\xi^j))w_{0i}^n g'(w_i^n \cdot \xi^j)\xi^j + 2\lambda w_i^n\big], \quad i = 1, 2, \cdots, q.$$

where the learning rate $\eta > 0$ is a constant.

3 A convergence result in [3]

Let us recall the convergence result in [3] so as to compare with our results. The main assumptions in [3] are as follows:

Assumption $(A1)$ $|g(t)|, |g'(t), |g''(t)|$ are uniformly bounded for $t \in \mathbb{R}$.

Assumption $(A2)$ $\|w_0^n\|$ $(n = 0, 1, 2, \ldots)$ are uniformly bounded.

Assumption $(A3)$ η and λ are chosen to satisfy $0 < \eta < \frac{1}{\lambda + C_1}$, where

$$C_1 = J(1 + C_2)C_3 \max\{C_2, C_5\} + \frac{1}{2}J(1 + C_2)C_3 + \frac{1}{2}JC_3^2C_4^2C_5,$$

$$C_2 = \max\big\{\sqrt{q}C_3, (C_3C_4)^2\big\},$$

$$C_3 = \max\big\{\sup_{t\in\mathbb{R}}|g(t)|, \sup_{t\in\mathbb{R}}|g'(t)|, \sup_{t\in\mathbb{R}}|g''(t)|, \sup_{t\in\mathbb{R},1\le j\le J}|g_j'(t)|, \sup_{t\in\mathbb{R},1\le j\le J}|g_j''(t)|\big\},$$

$$C_4 = \max_{1\le j\le J}\|\xi^j\|, \quad C_5 = \sup_{n\in\mathbb{N}}\|w_0^n\|. \tag{6}$$

Assumption $(A4)$ There exists a closed bounded region Φ such that $\{W^n\} \subset \Phi$, and the set $\Phi_0 = \{W \in \Phi : E_W(W) = 0\}$ contains only finite points.

The following convergence theorem is proved in [3].

Theorem 1 [3] *Suppose that the error function is given by* (3)*, that the weight sequence* $\{W^n\}$ *is generated by the algorithm* (5) *for any initial value* W^0*, and that Assumptions* $(A1) - (A3)$ *are valid. Then we have*

(a) $E(W^{n+1}) \le E(W^n), \quad n = 0, 1, 2, \cdots;$

(b) *There is* $E^* \ge 0$ *such that* $\lim_{n\to\infty} E(W^n) = E^*;$

(c) *There is* $M > 0$ *such that* $\|V^n\| \le \sqrt{\frac{1}{\lambda}E(W^n)} \le M$ *for all* $n = 0, 1, 2, \cdots;$

(d) $\lim_{n\to\infty}\big\|\Delta W^n\big\| = 0, \lim_{n\to\infty}\big\|E_W(W^n)\big\| = 0.$

Moreover, if Assumption $(A4)$ *is valid, then we have the strong convergence:*

(e) *There exists* $W^* \in \Phi_0$ *such that* $\lim_{n\to\infty} W^n = W^*$.

As a relative reference, we mention a similar result [4] for an online gradient method for training a feedforward network without hidden layer.

4 Main results

Our main result is the following boundedness theorem.

Theorem 2 *Suppose that the weight sequence* $\{W^n\}$ *is generated by the algorithm* (5) *for any initial value* W^0*, that Assumptions* $(A1)$ *is valid, and that* $0 < 2\lambda\eta < 1$*. Then,* $\{W^n\}$ *is uniformly bounded, i.e., there exists a constant* $C_6 > 0$ *such that*

$$\|W^n\| \le C_6, \ \forall n = 0, 1, \cdots. \tag{7}$$

Proof: (7) is equivalent to the existence of constants C_7 and C_8 such that for any nonnegative integer n

$$\|w_0^n\| \le C_7, \tag{8}$$

$$\|w_i^n\| \le C_8, \quad i = 1, 2, \cdots, q \tag{9}$$

First, we show (8). By $0 < 2\lambda\eta < 1$, we have

$$0 < 1 - 2\lambda\eta < 1. \tag{10}$$

By (5), we have

$$w_0^{n+1} = (1 - 2\lambda\eta)w_0^n - \eta\sum_{j=1}^{J} g_j'(w_0^n \cdot G(V^n\xi^j))G(V^n\xi^j). \tag{11}$$

(10) and (11) result in

$$\|w_0^{n+1}\| \le (1-2\lambda\eta)\|w_0^n\| + \eta\sum_{j=1}^{J}|g_j'(w_0^n\cdot G(V^n\xi^j))|\|G(V^n\xi^j)\|. \tag{12}$$

By Assumption $(A1)$, there is a constant $C_9>0$ such that for all $n=0,1,\cdots,$

$$\sum_{j=1}^{J}|g_j'(w_0^n\cdot G(V^n\xi^j))|\|G(V^n\xi^j)\| \le C_9. \tag{13}$$

We proceed to prove (8) by considering the following two cases.

Case (i): For any n $(n\ge 0)$, the inequality $\|w_0^n\|\le\frac{C_9}{\lambda}$ always holds. In this case, one can simply set $C_7=\frac{C_9}{\lambda}$ to validate (8).

Case (ii): There exists an integer N $(N\ge 0)$ such that

$$\|w_0^N\| > \frac{C_9}{\lambda}. \tag{14}$$

In this case, we can prove by induction on n that

$$\|w_0^n\| \le \|w_0^N\| + C_9,\ \forall n = N, N+1,\cdots. \tag{15}$$

(15) is evidently valid for $n=N$. So we suppose that (15) is valid for an integer n $(n\ge N)$, and we try to show that (15) is also valid for $n+1$.

If $\|w_0^n\|<\frac{C_9}{\lambda}$, by (10), (12) and (13) we have

$$\begin{aligned}\|w_0^{n+1}\| &\le (1-2\eta\lambda)\|w_0^n\| + \eta C_9\\ &\le (1-2\eta\lambda)\frac{C_9}{\lambda}+\eta C_9\\ &\le \frac{C_9}{\lambda}\le\|w_0^N\|\le\|w_0^N\|+C_9.\end{aligned} \tag{16}$$

On the other hand, if $\|w_0^n\|\ge\frac{C_9}{\lambda}$, a combination of (10), (12), (13) and (14) produces

$$\begin{aligned}\|w_0^{n+1}\| &\le (1-2\eta\lambda)\|w_0^n\| + \eta C_9\\ &\le (1-2\eta\lambda)\|w_0^n\| + \eta\lambda\|w_0^n\|\\ &= (1-\eta\lambda)\|w_0^n\|\\ &\le \|w_0^N\| + C_9.\end{aligned} \tag{17}$$

Now we have shown by induction that (15) is always true in this case. Hence, (8) is valid for Case (ii) by setting

$$C_7 = \max\{\|w_0^0\|,\|w_0^1\|,\cdots,\|w_0^{N-1}\|,\|w_0^N\|+C_9\}.$$

So (8) is true in both cases (i) and (ii), and the proof to (8) is completed.

With the help of (8), now we can prove (9). By (5), we have

$$\begin{aligned}w_i^{n+1} &= (1-2\lambda\eta)w_i^n\\ &\quad -\eta\sum_{j=1}^{J}g_j'(w_0^n\cdot G(V^n\xi^j))w_{0i}^n g'(w_i^n\cdot\xi^j)\xi^j,\\ &\quad i=1,2,\ldots,q.\end{aligned}$$

This gives

$$\begin{aligned}\|w_i^{n+1}\| &\le (1-2\lambda\eta)\|w_i^n\|\\ &\quad +\eta\sum_{j=1}^{J}|g_j'(w_0^n\cdot G(V^n\xi^j))w_{0i}^n g'(w_i^n\cdot\xi^j)|\|\xi^j\|,\\ &\quad i=1,2,\ldots,q.\end{aligned}$$

Due to (8) and Assumption $(A1)$, there is a constant $C_{10}>0$ such that for any $n=0,1,\cdots$ and $i=1,2,\cdots,q$,

$$\sum_{j=1}^{J}|g_j'(w_0^n\cdot G(V^n\xi^j))w_{0i}^n g'(w_i^n\cdot\xi^j)|\|\xi^j\|\le C_{10}.$$

Now, the remaining part of the proof to (9) can copy the corresponding proof to (8). The detail is omitted.

Finally we write $C_6=\sqrt{{C_7}^2+q{C_8}^2}$ to obtain for any nonnegative n that

$$\begin{aligned}\|W^n\| &= \sqrt{\|w_0^n\|^2+\|w_1^n\|^2+\cdots+\|w_q^n\|^2}\\ &\le \sqrt{{C_7}^2+q{C_8}^2}=C_6\end{aligned}$$

This completes the proof. □

The condition $0<2\lambda\eta<1$ is required for the above boundedness result. This is not a restrictive condition. In practice, the learning rate η and the penalty parameter λ are small and satisfy $0<2\lambda\eta<1$ easily. On the other hand, if $C_1>\lambda$, which is very likely the case, then the condition $0<2\lambda\eta<1$ results from the condition $0<\eta<\frac{1}{\lambda+C_1}$ in Assumption $(A3)$.

Thanks to Theorem 2, we can improve the existing convergence result Theorem 1 by cutting out Assumption $(A2)$.

Theorem 3 *Suppose that the error function is given by* (3)*, that the weight sequence* $\{W^n\}$ *is generated by the algorithm* (5) *for any initial value* W^0*, that* $0<2\lambda\eta<1$*, and that the assumptions* $(A1)$ *and* $(A3)$ *are valid. Then we have*

(a) $E(W^{n+1})\le E(W^n),\quad n=0,1,2,\cdots;$

(b) *There is* $E^*\ge 0$ *such that* $\lim_{n\to\infty}E(W^n)=E^*;$

(c) *There is* $M > 0$ *such that* $\|W^n\| \leq M$ *for all* $n = 0, 1, 2, \cdots$;
(d) $\lim_{n\to\infty} \|\Delta W^n\| = 0, \quad \lim_{n\to\infty} \|E_W(W^n)\| = 0.$
Moreover, if Assumption $(A4)$ *is valid, then we have the strong convergence:*
(e) *There exists* $W^* \in \Phi_0$ *such that* $\lim_{n\to\infty} W^n = W^*$.

Proof: A combination of Theorems 1 and 2 immediately leads to the conclusion. □

References:

[1] G. Hinton, Connectionist learning procedures, *Artificial Intelligence* 40(1989)185-243.

[2] S. Loone and G. Irwin, Improving neural network training solutions using regularisation, *Neurocomputing* 37(2001)71-90.

[3] W. Wu, H.M. Shao and Z.X. Li, Convergence of batch BP algorithm with penalty for FNN training, *Lecture Notes in Computer Science* 4232(2006)562-569.

[4] H. Shao, W. Wu and L.J. Liu, Convergence and monotonicity of an online gradient method with penalty for neural networks *WSEAS Transactions on Mathematics* 6:3(2007)469-476.

[5] S. Haykin, *Neural Networks: A Comprehensive Foundation. 2nd edition*, Tsinghua University Press and Prentice Hall, Beijing, 2001.

[6] W. Wu, G. Feng, Z. Li and Y. Xu, Deterministic Convergence of an Online Gradient Method for BP Neural Networks, *IEEE Transactions on Neural Networks* 16(2005)533-540.