Neural Networks 82 (2016) 49–61

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Boundedness and convergence analysis of weight elimination for cyclic training of neural networks☆

Jian Wang [a,c,*], Zhenyun Ye [b], Weifeng Gao [a], Jacek M. Zurada [c,d]

[a] College of Science, China University of Petroleum, Qingdao, 266580, China
[b] College of Computer & Communication Engineering, China University of Petroleum, Qingdao, 266580, China
[c] Department of Electrical and Computer Engineering, University of Louisville, Louisville, KY, 40292, USA
[d] Information Technology Institute, University of Social Sciences, Łódź 90-113, Poland

ARTICLE INFO

Article history:
Received 9 February 2016
Received in revised form 14 May 2016
Accepted 21 June 2016
Available online 14 July 2016

Keywords:
Neural networks
Weight decay
Weight elimination
Boundedness
Convergence

ABSTRACT

Weight elimination offers a simple and efficient improvement of training algorithm of feedforward neural networks. It is a general regularization technique in terms of the flexible scaling parameters. Actually, the weight elimination technique also contains the weight decay regularization for a large scaling parameter. Many applications of this technique and its improvements have been reported. However, there is little research concentrated on its convergence behavior. In this paper, we theoretically analyze the weight elimination for cyclic learning method and determine the conditions for the uniform boundedness of weight sequence, and weak and strong convergence. Based on the assumed network parameters, the optimal choice for the scaling parameter can also be determined. Moreover, two illustrative simulations have been done to support the theoretical explorations as well.

1. Introduction

Artificial neural networks have been used in many applications such as machine learning, computational intelligence, image analysis and pattern recognition (Haykin, 2009; Ripley, 2008). Feedforward neural networks (FNNs) are one of the most popular architectures due to their ability to approximate nonlinear mappings. Furthermore, backpropagation (BP) is the most widely used algorithm to train such networks with the gradient-based methods (Rumelhart, Hinton, & Williams, 1986; Werbos, 1974).

There are two main modes of BP training: batch and incremental modes. Batch learning corresponds to the standard gradient method, and weights are updated after the complete presentation of the training set. The incremental learning is a variation of the standard gradient method, but weights are updated after the presentation of each training pair.

There are three main types of incremental learning: online, almost-cyclic and cyclic (Heskes & Wiegerinck, 1996). They differ by the ordering of training samples. For online learning, the training samples are fed in a completely random order and no epochs are discernible. In contrast, for cyclic learning, the training sequence is fixed and identical within each training epoch after it has been randomly determined before learning. Almost-cyclic learning is an intermediate mode between online and cyclic learning. It supplies samples in sequence that is randomized separately for each training epoch.

Generalization is one of the important metrics to evaluate the performance of a network. The BP training of neural networks can be viewed as a "curve-fitting" model (Cherkassky & Mulier, 2007; Friedman, 1994; Haykin, 2009). The objective here is to seek a suitable nonlinear mapping for the input–output relationships, so that the network is able to classify not only training samples but also testing samples.

Adding a penalty term to the cost function is important to improve the generalization on the test data. It is implemented by minimizing the following total risk:

$$R(\mathbf{w}) = E_c(\mathbf{w}) + \lambda \Phi[f(\mathbf{x}, \mathbf{w})]. \quad (1)$$

☆ This work was supported in part by the National Natural Science Foundation of China (No. 61305075), the China Postdoctoral Science Foundation (No. 2012M520624), the Natural Science Foundation of Shandong Province (No. ZR2013FQ004, ZR2013DM015), the Specialized Research Fund for the Doctoral Program of Higher Education of China (No. 20130133120014) and the Fundamental Research Funds for the Central Universities (No. 13CX05016A, 14CX05042A, 15CX05053A, 15CX08011A).

* Corresponding author at: College of Science, China University of Petroleum, Qingdao, 266580, China.
E-mail addresses: wangjiannl@upc.edu.cn (J. Wang), jacek.zurada@louisville.edu (J.M. Zurada).

http://dx.doi.org/10.1016/j.neunet.2016.06.005
0893-6080/

The first term, $E_c(\mathbf{w})$, is the standard error to ensure a good approximation for the training samples. The second term, $\Phi[f(\mathbf{x}, \mathbf{w})]$, is the complexity penalty, which imposes prior knowledge on the optimization solutions. The penalization coefficient λ controls the relative importance of the penalty term with respect to the standard error.

There are two main classes of penalty forms for neural networks: nonparametric and parametric penalties (Girosi, Jones, & Poggio, 1995). For nonparametric penalty, (1) evaluates the trade-off between training and testing samples by using a differential operator. By contrast, the parametric penalty deals with the complexity directly by imposing constraints on the training parameters. Its general form is

$$\Phi[f(\mathbf{x}, \mathbf{w})] = \|\mathbf{w}\|_k, \tag{2}$$

where $\|\mathbf{w}\|_k$ represents the k-norm of the weight vector $\mathbf{w}$.

Many existing BP learning algorithms can be considered as special cases of this regularization method. Particularly, (2) is referred to as the L_0 regularizer when $k = 0$. When $k = 1$, it is the L_1 regularizer, which is also called the Lasso penalty. When $k = 2$, it corresponds to the L_2 regularizer, which represents the well-known ridge regression or weight decay (Gupta & Lam, 1998; Haykin, 1999; Leung, Tsoi, & Chan, 2001).

Smaller weights result in better generalization for the trained networks even if they show similar performance on the training set. Thus, to improve the generalization it is important to study the boundedness of weights and convergence behavior of a trained network from theoretical point of view. Because of the nondifferentiable property of L_0 and L_1 regularizers, most published results are focused on neural networks with L_2 penalty (Shao & Zheng, 2011; Wang, Wu, & Zurada, 2011, 2012; Zhang & Wu, 2009).

The paragraphs below provide a review of recently published works closely related to this paper. In Wu, Shao, and Li (2006), the weak convergence and monotonicity are proved which correspond to the classical gradient descent method from optimization theory. One of its main contributions (Wu et al., 2006) is that the boundedness of the weights between input and hidden layers are rigorously stated.

For online training, Zhang and Wu (2009) shows the weight boundedness and convergence results for FNNs with the linear output. An extension in Zhang, Wu, Liu, and Yao (2009) reaches the same theoretical conclusions for the activation function of output layer is the common sigmoid function. We note that the convergence results of these papers are asymptotic convergence with probability 1 due to the completely random ordering of samples during training. Besides the convergence discussion on FNNs, extended finite time convergence results have been studied for the recurrent neural networks (Cao, Rakkiyappan, Maheswari, & Chandrasekar, 2016; Liu & Cao, 2010; Liu, Cao, & Chen, 2010; Liu, Dang, & Cao, 2010).

For cyclic learning, the deterministic convergence result has been comprehensively studied in Xu, Zhang, and Jing (2009) and Zhang, Xu, Huang, and Wang (2012) under mild constrains for the learning rates. In Wang et al. (2012), the cyclic and almost-cyclic learning with weight decay penalty (L_2 regularizer) has been studied under more relaxed conditions for training parameters. Interestingly, the deterministic convergence has been in detail proved in contrast to those probabilistic conclusions for online learning.

Weight decay regularization constrains the size of weights and penalizes large weights. Although the cost function for weight decay penalty is smooth, the method does not result in sparse networks and does not have a strong ability to effectively prune hidden units.

An improved algorithm with weight elimination penalty has been proposed in Weigend, Rumelhart, and Huberman (1991). It tries to reduce the small weights to zero and then trims them. This method is well suited for network pruning by eliminating units which offer little or no benefit in evaluating the correct output (Ennett & Frize, 2003; Frize, Ennett, Stevenson, & Trigg, 2001). Particularly, weight decay becomes a special case of weight elimination when the scaling parameter is large enough.

More variants and applications of weight elimination technique have been demonstrated in Bebis, Georgiopoulos, and Kaspalris (1996) Ennett and Frize (2003), Gupta and Lam (1998), Kuo, Wu, and Wang (2002) and Leung et al. (2001) which display their good performance. A particle swarm neural network was constructed in Rakitianskaia and Engelbrecht (2014) with weight elimination penalty. Swarm behavior had been detailedly studied and showed that weight elimination resulted in smaller networks and more convergent swarms.

The aim of this paper is to present a comprehensive study and theoretical analysis of weight elimination in cyclic learning of neural networks. The main contributions of this paper are as follows:

(1) *In addition to the qualitative discussion in* Wang, Zurada, Wang, Wang, and Xie (2014) *of the scaling parameter q, its optimal choice has been rigorously proved.*

In Wang et al. (2014), the boundedness of the weight sequence in batch learning mode has been studied. However, results are in a qualitative rather than a rigorous proof, and do not address the theoretical analysis of convergence. To obtain the optimal scaling parameter, this paper investigates in detail a specific quartic function, and the monotonicity and concavity of its roots.

(2) *The boundedness of the weight sequence has been guaranteed for the weight elimination in cyclic learning.*

Overfitting is a crucial problem in training neural networks. Generally speaking, a trained network with smaller weights performs better generalization and avoids overfitting. To achieve this goal, we have evaluated how to restrain the magnitude of weights. In the paper we state the uniform boundedness of weight sequence under some reasonable assumptions.

(3) *A comprehensive study of the weak and strong convergence results for weight elimination in cyclic learning has been demonstrated.*

The weak convergence means that the norm of gradient of the objective function with respect to the weight vectors approaches zero as the iterations continue. The strong convergence occurs when the weight updating sequence approaches a fixed point.

(4) *Numerical simulations demonstrate the advantages of weight elimination in comparison with the weight decay method. In addition, the theoretical conclusions are verified by the simulations.*

An illustrated experiment of function regression shows the merits of weight elimination method. The performance of trained network based on weight decay and weight elimination penalties have been compared in terms of pruning ability and generalization. Consequently, the uniform boundedness of weight sequence and convergence behavior have been clearly graphed.

The rest of the paper is as follows: the weight elimination algorithm for cyclic learning mode has been introduced in the next Section. The main results are presented in Section 3 under some mild assumptions for activation functions and learning rates. In Section 4, the detailed proofs of the main results are presented based on some fundamental lemmas. Two simulations in Section 5 illustrate the main results of this paper. Finally, we conclude with some useful remarks in Section 6.