CONJUGATE GRADIENT METHODS IN TRAINING NEURAL NETWORKS

Zarita Zainuddin, Saratha Sathasivam and Yahya Abu Hassan
School of Mathematical Sciences, Universiti Sains Malaysia,
11800 USM, Pulau Pinang, Malaysia
Tel no.: 604 - 6577888 ext 3648, Fax no.: 604 – 6570910,
e -mail: zarita@cs.usm.my

Abstract. *Training of artificial neural networks is normally a time consuming task due to iterative search imposed by the implicit nonlinearity of the network behavior. To tackle the supervised learning of multilayer feed forward neural networks, the backpropagation algorithm has been proven to be one of the most successful neural network algorithm. Although backpropagation training has proved to be efficient in many applications, its convergence tends to be very slow and it often yields suboptimal solutions. Standard backpropagation, as with many gradient based optimizaton methods converges slowly as neural networks problems become larger and more complex.*

This paper concentrates on conjugate gradient-based training methods originated from optimization theory, namely, Fletcher Reeves conjugate gradient, Polak-Ribierre conjugate gradient and Powell-Beale restart. The behavior of these training methods on several real life application problems is reported, thereby illuminating convergence and robustness. The real world problems which have been considered include Classification of Iris Plant, Gender Classification of Crabs and Classification of Face Images. By using these algorithms, the convergence rate can be improved immensely with only a minimal increase in the complexity. Numerical evidence shows that these methods do perform well. (ATCMA264)

1 Introduction

The back propagation method consists three main layers-input layers, output layers and hidden layers. The input nodes constitute the first layer and the output nodes constitute the output layer while the remaining nodes constitute hidden layers of the network. The input vector is presented to the input layer and the signals are propagated forward to the first hidden layer; the resulting output of the first hidden layer is in turn applied to the next hidden layer and the same procedure continues for the rest of the network. The objective of the learning process is to adjust the free parameters (i.e. synaptic weights and thresholds) of the network so as to minimize ξ_{av}.

The back propagation method uses the gradient or steepest descent method to perform the minimization where the weights are updated (adjusted) in accordance with the respective errors computed for each pattern to the network. The error signal $e_j(n)$ at the output of neuron j at iteration n is defined by

$$e_j(n) = d_j(n) - y_j(n) \tag{1}$$

where $d_j(n)$ and $y_j(n)$ is the desired and the actual response of neuron j at iteration n respectively.

The instantaneous value $\xi(n)$ of the sum of squared errors over all neurons is written as

$$\xi(n) = \frac{1}{2}\sum_{j \in C} e_j^2(n) \tag{2}$$

where C includes all the neurons in the output layer of the network. Lastly, the average squared error over the total number of patterns N is given by

$$\xi_{\text{av}} = \frac{1}{N}\sum_{n=1}^{N} \xi(n) \tag{3}$$

By using gradient descent method to perform the minimization, the correction $\Delta w_{ji}(n)$ applied to the synaptic weight $w_{ji}(n)$ is proportional to the instantaneous gradient $\partial\xi(n)/\partial w_{ji}(n)$.
The gradient $\partial\xi(n)/\partial w_{ji}(n)$ determines the direction of search in weight space for the synaptic weight w_{ji}. According to the delta rule, the correction $\Delta w_{ji}(n)$ is defined by

$$\Delta w_{ji}(n) = -\eta \frac{\partial \xi(n)}{\partial w_{ji}(n)} \tag{4}$$

where η is the learning rate parameter.

2 Algorithms

In Evans [1], the reasons for the slow convergence of the backpropagation have been discussed. To date, many techniques have been proposed to deal with the inherent problems of backpropagation. These techniques can be divided roughly into two main categories; those referred to as global techniques that use global knowledge of the state of the entire network, such as the direction of the overall weight update vector. Most of these techniques have their roots in the well-explained domain of optimization theory. The simplest is a first-order method that uses the steepest-descent (SD) direction [2]. An alternative is the conjugate gradient (CG) method, which modifies the SD direction by conjugating it with the previously used direction [3].

In contrast, local adaptation strategies are based on weight specific information only, such as the temporal behavior of the partial derivative of the current weight. These include the Delta-Bar-Delta method [4], Quickprop [5] and Rprop [6].

2.1 Conjugate Gradient Methods

In optimization theory, the conjugate gradient method has been known since Fletcher and Reeves [7]. Leonard and Kramer [3] introduced the original Fletcher – Reeves algorithm in the field of neural network research. Conjugate gradient does not require the calculation of second derivatives but, yet, it still has the quadratic convergence property.

Let us assume that the error function is quadratic in **w**, that is, it can be approximated to a quadratic function as

$$E(\mathbf{w}) = c - \mathbf{b}^{\mathrm{T}}\mathbf{w} + \frac{1}{2}\mathbf{w}^{\mathrm{T}}\mathbf{A}\mathbf{w} \tag{5}$$

where **A** is a symmetric positive definite matrix. Let **p**(n) denote the direction vector at iteration n of the algorithm. Then the weight vector of the network is updated in accordance with the rule

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta(n)\mathbf{p}(n) \tag{6}$$

where η(n) is the learning-rate parameter. Suppose the initial direction of minimization, which is started at **w**(0) is **p**(0) which is set equal to the negative gradient vector **g**(n) at the initial point n=0; that is,

$$\mathbf{p}(0) = -\mathbf{g}(0)\,. \tag{7}$$

A line minimization in the direction of **p**(0) results in a gradient at **w**(1) perpendicular to **p**(0). In general,

$$\mathbf{p}^{\mathrm{T}}(n)\mathbf{g}(n+1) = 0\,. \tag{8}$$

Because we do not want to spoil this minimization step in subsequent minimizations, the gradient of subsequent points of minimization must also be perpendicular to **p**(n):

$$\mathbf{p}^{\mathrm{T}}(n)\mathbf{g}(n+2) = 0\,. \tag{9}$$

Therefore, with Eq. (8) and Eq. (9),

$$\mathbf{p}^{\mathrm{T}}(n)(\mathbf{g}(n+2) - \mathbf{g}(n+1)) = 0\,. \tag{10}$$

Now, **g**(n+2) – **g**(n+1) is the change in the gradient as we move from **w**(n+1) to **w**(n+2). From Eq. (5), the gradient of E at **w**(n) can be found to be

$$\mathbf{g}(n) = \mathbf{A}\mathbf{w}(n) - \mathbf{b}\,. \tag{11}$$

Therefore,

$$\begin{aligned} 0 &= \mathbf{p}^{T}(n)((\mathbf{g}(n+2) - \mathbf{g}(n+1)) \\ &= \mathbf{p}^{T}(n)\mathbf{A}(\mathbf{w}(n+2) - \mathbf{w}(n+1)) \\ &= \mathbf{p}^{T}(n)\mathbf{A}\mathbf{p}(n+1)\alpha(n+1) \end{aligned} \tag{12}$$

or

$$\mathbf{p}^{\mathrm{T}}(n)\mathbf{A}\mathbf{p}(n+1) = 0\,. \tag{13}$$

When Eq. (13) holds for two vectors **p**(n) and **p**(n+1), these vectors are said to be conjugate.

After a line minimization along **p**(n), a point **w**(n+1) is reached, the next minimization direction is constructed using

$$\mathbf{p}(n+1) = -\mathbf{g}(n+1) + \beta(n)\mathbf{p}(n)\,. \tag{14}$$

There are various rules to determine β(n) in order to ensure conjugacy of **p**(n) and **p**(n+1); two alternate rules are the following:

- The Fletcher-Reeves formula [7]:

$$\beta(n) = \frac{\mathbf{g}^{\mathrm{T}}(n+1)\mathbf{g}(n+1)}{\mathbf{g}^{\mathrm{T}}(n)\mathbf{g}(n)}\,. \tag{15}$$

- The Polak-Ribiere formula [8]:

$$\beta(n) = \frac{\mathbf{g}^{\mathrm{T}}(n+1)[\mathbf{g}(n+1) - \mathbf{g}(n)]}{\mathbf{g}^{\mathrm{T}}(n)\mathbf{g}(n)}\,. \tag{16}$$

In order to find a particular value of η(n) in the update rule of Eq. (6) a line search is involved; that is to say, we have to find a value of η(n) for which $E(\boldsymbol{w}(n) + \eta\boldsymbol{p}(n))$ is minimized, given fixed values of **w**(n) and **p**(n). This η(n) is defined by

$$\eta(n) = \arg\min_{\eta}\{E(\mathbf{w}(n) + \eta\mathbf{p}(n))\} \tag{17}$$

The performance of the conjugate gradient method is greatly influenced by the accuracy of the line search.

2.2 Powell-Beale Restart Procedure

The conjugate gradient method can be improved by periodically resetting the search direction to the negative of the gradient, i.e. ,

$$\mathbf{p}(n+I) = -\mathbf{g}(n+1) \tag{18}$$

Since this procedure is ineffective, a restarting method that does not abandon the second derivative information is needed. One such reset method has been proposed by Powell [9]. For this technique, we will restart if there is very little orthogonality left between the current gradient and the previous gradient . This is tested with the following inequality

$$\left\|\mathbf{g}^T(n-1)\mathbf{g}(n)\right\| \geq 0.2\left\|\mathbf{g}(n)\right\|^2 \tag{19}$$

If this condition is satisfied, the search direction is reset to the negative of the gradient

3 Simulations on Real-World Applications

Neural Network Toolbox in Matlab R12 version 6 was used to stimulate the following data sets.

3.1 Gender Classification of Crabs

A 6-5-2 network was used where the six inputs correspond to species, frontal lip, rear width, length, width and depth and the two output nodes correspond to male and female. The training set consists of 120 vector pairs while the testing set consists of 80 vector pairs. The learning process was terminated when the mean squared error (MSE) reached $1*10^{-6}$ within 100 epochs. Table 1 shows the results of the simulations, which are the average of 10 trials. The learning curves in a typical run of both the Fletcher Reeves and Polak Ribiere, together with Powell restart are shown in Figure 1.

Table1. The simulation results for the Gender Classification of Crabs using Conjugate Gradient Methods

Algorithm	Epoch	Mean squared error	Gradient
Fletcher Reeves	0	0.64757	1.33668
	25	0.322316	0.766277
	50	0.311316	0.241607
	75	0.297807	0.704012
	100	0.159405	0.164871
Polak Ribiere	0	1.27289	1.34359
	25	0.345189	0.0459379
	50	0.313014	0.0458141
	75	0.242026	0.0391324
	100	0.199122	0.0287724
Powell-Beale	0	1.48478	0.269887
	25	0.333582	0.0363292
	50	0.320753	0.106848
	75	0.306246	0.0294416
	100	0.297566	0.10075

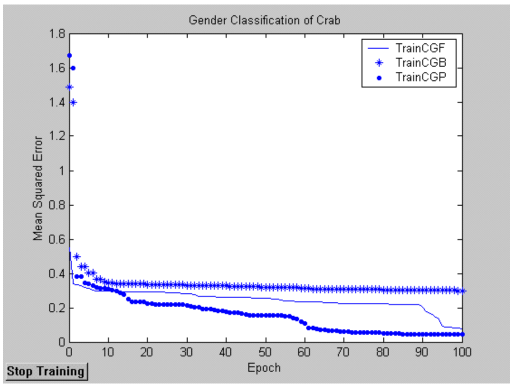

Figure 1. The learning progressions of the Gender Classification of Crabs Problem

Polak Ribiere is definitely better than the other algorithms escaping shallow local minima and providing fast training. It cab be observed that Polak Ribiere conjugate gradient starts to outperform the other methods right at the beginning of the learning and the MSE decreases sharply until it reaches 100 epochs. With the Fletcher Reeves and Powell restart, the decrease in MSE is rather gradual during the course of learning until it reaches 100 epochs. There were no cases of the solution getting stuck in a local minimum, indicating that the choice of initial weights is suitable.

3.2 Classification of Iris Plant

The data set consists of three different species, Setosa, Versicolor and Virginica. The network consists of 4 inputs-sepal length, sepal width and petal width. A 4-2-3 network was used where the output nodes correspond to the 3 classification classes. The training set consists of 99 vector pairs while the testing set consists of 50 vector pairs. For all simulations, the training has been continued until the MSE reached $1*10^{-6}$ within 100 epochs. The MSE after 25, 50, 75 and 100 epochs respectively are summarized in Table 2.

Table2. The simulation results for the Classification of Iris Plant using Conjugate Gradient Methods

Algorithm	Epoch	Mean squared error	Gradient
Fletcher Reeves	0	0.975378	0.20021
	25	0.347129	0.0467346
	50	0.323631	0.0510218
	75	0.30802	0.0762527
	100	0.289361	0.0192811
Polak Ribiere	0	1.4576	0.173127
	25	0.133615	0.0521216
	50	0.126588	0.0448276
	75	0.119296	0.0369579
	100	0.109502	0.00644071
Powell-Beale	0	0.350441(after 6 epochs)	0.12742
	25		
	50		
	75		
	100		

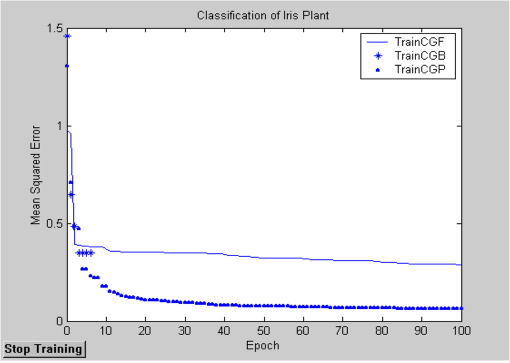

Figure 2. The learning progressions of the Classification of Iris Plant Problem

As can be observed, the Polak Ribiere has a remarkable advantage in accelerated convergence providing a reduction in MSE of up to 96.27 % after 100 epochs have been reached. The Fletcher Reeves conjugate gradient exhibits gradual convergence after about 4 epochs. In contrast, Polak Ribiere conjugate gradient develops a sharp decrease in MSE immediately at the start of the training process until 100 epochs have been reached. Powell restart could not converge to the required solution indicating that it gets stuck in a local minimum.

3.3 Human Face Recognition Problem

A 460-12-5 network was used where the output nodes correspond to the 5 classification classes. The training and testing sets consist of 45 images each. A detailed description of the data set can be found in Evans et al [1]. Similar to the previous problems, the training has been continued until the MSE reached $1*10^{-6}$ within 100 epochs. Table 3 shows the results of the simulations which are the average of 10 trials.

Table3. The simulation results for the Human Face Recognition Problem using Conjugate Gradient Methods

Algorithm	Epoch	Mean squared error	Gradient
Fletcher Reeves	0	0.494581	2.56718
	25	0.02415	0.135225
	50	0.00613019	0.058746
	75	0.001435	0.0329887
	100	0.00061473	0.0245373
Polak Ribiere	0	0.711677	2.00961
	25	0.0291612	0.168869
	50	0.06418198	0.0752522
	75	0.00192943	0.0341702
	100	0.000986186	0.0259714
Powell-Beale	0	0.685621	1.35497
	25	0.0306093	0.0871412
	50	0.0171283	0.0508304
	75	0.00985756	0.0293154
	100	0.00218934	0.139763

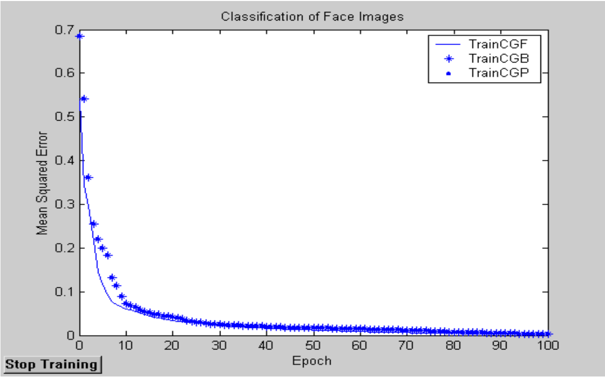

Figure 3. The learning progressions of the Human Face Recognition Problem

As can be observed, Fletcher Reeves, Polak Ribiere and Powell restart had similar performance. After 100 epochs is reached a reduction in MSE of up to almost 100 % was obtained for all three methods. A typical run of all the methods is shown is Figure 3. It can be observed that all the three methods are able to provide accelerated convergence right at the beginning of the learning and the MSE decreases sharply until 100 epochs is reached.

4. Conclusion

The conjugate gradient methods, namely, Fletcher Reeves conjugate gradient, Polak Ribiere conjugate gradient and Powell-Beale restart have proven to be very effective and superior in terms of convergence when tested and mutually compared on three real world application problems: gender classification of crabs, classification of iris plant and human face recognition problem. Conjugate gradient algorithms, which can be seen as error back propagation with momentum, were shown to be a good choice for feed forward network training. In particular, Polak Ribiere conjugate gradient shows promising results for training feed forward networks. The conjugate gradient methods have advanced convergence rates since they use second order information to calculate the new direction.

References

[1] Evans, D.J. Ahmad Fadzil M.H. & Zainuddin, Z. (1997). Accelerating backpropagation in human face recognition, *Proc. IEEE Int. Conf. on Neural Networks*, IEEE Press, 1347-1352.

[2] Rumelhart D.E., Hinton G.E., Williams R.J. (1986). Learning internal representations by error propagation., In Rumelhart D.E. McClelland J.L., editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, MIT Press, Cambridge, MA., 318-362.

[3] Leonard, J., & Kramer, M.A. (1990). Improvement to the back-propagation algorithm for training neural networks. *Computers and Chemical Engineering*, 14(3), 337-341.

[4] Jacobs, R. A. (1988). Increased rates of convergence through learning rate adaptation, *Neural Networks,* 1(4), 295-308.

[5] Fahlman S. E. (1989). Faster learning variation on backpropagation: an empirical study. In *Proceedings of the 1988 Connectionist Models*, p. 38-51, Kaufman.

[6] Riedmiller, M. & Braun, H. (1993). A direct adaptive method for faster backpropagation learning. The RPROP algorithm. In *Proceedings of the IEEE International Conference on Neural Networks (ICNN)*, Ruspini, H., ed., p. 586-591. San Francisco.

[7] Fletcher, R., & Reeves, C.M. (1964). Function minimization by conjugate gradients. *Computer Journal*, 7, 149-154.

[8] Polak, E. (1971). Computational methods in optimization. New York; Academic press.

[9] Powell, M. J. D.(1977). Restart procedures for the conjugate gradient method. *Mathematical Programming*, 12, 241-254.