Improving Performance of Back propagation Learning Algorithm

Harikrishna B Jethva[1] Dr. V. M. Chavda[2]
[1]Ph. D. Scholar,Department of Computer Science and Engineering
[1]Bhagwant University, Sikar Road Ajmer, Rajasthan
[2]SVICS, Kadi, Gujarat

Abstract— The standard back-propagation algorithm is one of the most widely used algorithm for training feed-forward neural networks. One major drawback of this algorithm is it might fall into local minima and slow convergence rate. Natural gradient descent is principal method for solving nonlinear function is presented and is combined with the modified back-propagation algorithm yielding a new fast training multilayer algorithm. This paper describes new approach to natural gradient learning in which the number of parameters necessary is much smaller than the natural gradient algorithm. This new method exploits the algebraic structure of the parameter space to reduce the space and time complexity of algorithm and improve its performance.

I. INTRODUCTION

The back-propagation (BP) training algorithm is a supervised learning method for multi-layered feed-forward neural networks. It is essentially a gradient descent local optimization technique which involves backward error correction of network weights. Despite the general success of back-propagation method in the learning process, several major deficiencies are still needed to be solved. The convergence rate of back-propagation is very low and hence it becomes unsuitable for large problems. Furthermore, the convergence behavior of the back-propagation algorithm depends on the choice of initial values of connection weights and other parameters used in the algorithm such as the learning rate and the momentum term.

Amari has developed natural gradient learning for multilayer perceptrons [18], which uses Quasi-Newton method [6] instead of the steepest descent direction. The Fisher information matrix is a technique used to estimate hidden parameters in terms observed random variables. It fits very nicely into Quasi-Newton optimization framework.

This paper suggests that a simple modification to the initial search direction, in the above algorithm i.e. changing the gradient of error with respect to weights, to improve the training efficiency. It was discovered that if the gradient based search direction is locally modified by a gain value used in the activation function of the corresponding node, significant improvements in the convergence rates can be achieved [24].

II. BACKPROPAGATION LEARNING ALGORITHM

An artificial neural network consist of input vector *x* and gives output *y*. when network has m hidden units, the output of hidden layer is $\varphi(w_\alpha \cdot x)$, $\alpha = 1,\ldots,m$ where w_α is an n dimensional connection weight vector from input to the α-th hidden unit, and φ is a sigmoidal output function. Let v_α be a connection weight from the α-th hidden unit to the linear output unit and let ζ be a bias. Then the output of the neural network is written as

$$y = \sum_{\alpha=1}^{m} v_\alpha \varphi(w_\alpha \cdot x) + \zeta \qquad (1)$$

Any perceptron is specified by the parameter $\{w_1,\ldots,w_\alpha; v\}$. We summarize them into a single m(n+1) dimensional vector θ. We call the space S consisting of all multilayer neurons. The parameter θ plays the role of a coordinate system of S.

The vector θ of dimension m(n+1) can represent a single neuron.

The output of a neuron is a random variable depends on input x. Hence the input output relation of the neuron having parameter θ is described by the conditional probability of output y on input x,

$$p(y|x,\theta) = \frac{1}{\sqrt{2\pi\sigma}} exp\left[-\frac{1}{2\sigma^2}\{y - f(x,\theta)\}^2\right] \qquad (2)$$

Where $f(x,\theta) = \sum v_\alpha \varphi(w_\alpha \cdot x)$ (3)

is the mean value of y given input x. Its logarithm is

$$l(y|x,\theta) = \frac{1}{2\sigma^2}\{y - f(x,\theta)\}^2 - \log(\sqrt{2\pi\sigma}) \qquad (4)$$

This can be regarded as the negative of the square of an error when y is a target value and $f(x,\theta)$ is the output of the network.

Hence, the maximization of the likelihood is equivalent to the minimization of the square error

$$l^*(y,x,\theta) = \frac{1}{2}\{y - f(x,\theta)\} \qquad (5)$$

The conventional on-line learning method modifies the current parameter θ_t by using the gradient (∇) of the loss function such that

$$\theta_{t+1} = \theta_t - \eta_t \nabla l^*(x_t, y^*_t; \theta_t) \qquad (6)$$

here η_t is a learning rate, and

$$\nabla l^*(x, y*, \theta) = \left\{\frac{\partial}{\partial\theta_i} l^*(x, y^*; \theta)\right\} \qquad (7)$$

is the gradient of the loss function l^* and y^*_t is the desired output signal given from teacher.

The steepest descent direction of the loss function $l^*(\theta)$ in a Riemannian space is given [18] by

$$-\tilde{\nabla} l^*(\theta) = -G^{-1}(\theta)\nabla l^*(\theta) \qquad (8)$$

Where G^{-1} is the inverse of a matrix $G = (g_{ij})$ called the Riemannian metric tensor. This gradient is called natural gradient of the loss function $l^*(\theta)$ in the Riemannian space.

III. NATURAL GRADIENT LEARNING ALGORITHM

In the multilayer neural network, the Riemannian metric tensor $G(\theta) = (g_{ij}(\theta))$ is given by the Fisher information matrix[18],

$$g_{ij}(\theta) = E\left[\frac{\partial l(y|x;\theta)}{\partial\theta i}\frac{\partial l(y|x;\theta)}{\partial\theta j}\right] \qquad (9)$$

where E denotes expectation with respect to the input output pair (x,y) given in Eq.(2).

The natural gradient learning algorithm updates the current θ_t by

All rights reserved by www.ijsrd.com

$$\theta_{t+1} = \theta_t - \eta_t \ddot{\nabla} l^*(\theta) \qquad (10)$$

IV. Adaptive Implementation of Natural Gradient Learning

The Fisher information $G(\theta)$ depends on the probability distribution of x which is usually unknown. Hence, it is difficult to obtain $G(\theta)$. Moreover, its inversion is costly. Here, we show an adaptive method of directly estimating $G^{-1}(\theta)$ [5].

Since the Fisher information of Eq. (9) can be rewritten, by using Eq. (4), as

$$G_t = E\left[\frac{\partial l(y|x;\theta t)}{\partial \theta t}\frac{\partial l(y|x;\theta t)'}{\partial \theta t}\right] = \frac{1}{4\sigma^4}E\{y - f(x,\theta)\}^2\}$$
$$E\left[\frac{\partial f(x;\theta t)}{\partial \theta t}\frac{\partial f(x;\theta t)'}{\partial \theta t}\right] = \frac{1}{4\sigma^2}E\left[\frac{\partial f(x;\theta t)}{\partial \theta t}\frac{\partial f(x;\theta t)'}{\partial \theta t}\right] \qquad (11)$$

where ' denotes transposition of a vector or matrix.

We have following recursive estimation of G^{-1}[23]

$$G_{t+1}^{-1} = (1 + \varepsilon t)G_t^{-1} - \varepsilon t G_t^{-1}\nabla ft(\nabla ft)^T G_t^{-1} \qquad (12)$$

Where ε_t is a small learning rate, $\nabla f = (\partial/\partial\theta)f$ and $f_t = f(x_t,\theta_t)$. Together with

$$\theta_{t+1} = \theta_t - \eta_t G_t^{-1}\nabla l(x_t, y_t, \theta_t) \qquad (13)$$

this gives the adaptive method of natural gradient learning. This is different from the Newton method, but can be regarded as an adaptive version of Gauss Newton method. Moreover, information geometry suggests the important geometric properties of hierarchical statistical model in general.

V. Experimental Results

We conducted an experiment for comparing convergence speeds between conventional Natural Gradient Learning (NGL) algorithm, and the Adaptive Natural Gradient learning (ANGL) algorithms.

We take XOR problem because it is not linearly separable problem. We use NN architecture with two hidden units and hyperbolic tangent transfer function between both the hidden units and output units.

The inputs and outputs are:

$X_0 = \begin{bmatrix}-1\\-1\end{bmatrix}$ $X_1 = \begin{bmatrix}-1\\1\end{bmatrix}$ $X_2 = \begin{bmatrix}1\\-1\end{bmatrix}$ $X_3 = \begin{bmatrix}1\\1\end{bmatrix}$

$Y_0 = -1$ $Y_1 = 1$ $Y_2 = 1$ $Y_3 = -1$

Respectively.

Thus the error for each pattern is

$$\epsilon_n = y_n - tanh(W_2 tanh(W_1 x_n + b_1) + b_2)^2 \qquad (14)$$

There are two hidden units and each layer has bias. Hence W_1 is a 2-by-2 matrix and W_2 is a 1-by-2 matrix.

The performance compared with sum squared error metric.

Neural network training algorithms are very sensitive to the learning rate. So we use step size $\eta\|\nabla J\|$ for NGL algorithm. An interesting point of comparison is the relative step size of this algorithm. For ANGL, the effective learning rate is the product of the learning rate η and the largest eigenvalue of the G^{-1}.

Figure 1, 2 shows the sum squared error of each learning epoch for NGL and ANGL. Table 1 show the parameters used in the three learning algorithms and some of the result of the experiment.

Parameter	NGL	ANGL
Hidden units	2	2
Learning rate	0.25	0.25
Adaption rate	N.A.	0.1
Learning Epoch When SSE < 0.02	10000	320
Final SSE	0.0817	3.55e-4
Final Learning Rate	1e-4	0.144

Table 1: The result of XOR Experiment And Parameter Used

VI. Conclusion

Natural Gradient Descent learning works well for many problems. Amari[18] had developed an algorithm to avoid local minima by following the curvature of a manifold in the parameter space of neuron. By using recursive estimate of the inverse of the Fisher information matrix of the parameters, the algorithm is able to accelerate learning in the direction of descent.

The experiment have shown that the performance of natural gradient algorithm improved by using adaptive gradient method of learning.

There are many areas of research in which this research can be applied, like speech recognition etc.

REFERENCES

[1] D. E. Rumelhart and J. L. McClelland, Parallel Distributed Processing. Cambridge, MA: MIT Press, 1986.

[2] D. O. Hebb, The Organization of Behavior. New York: John Wiley & Sons, 1949.

[3] D. J. C. MacKay, Information Theory, Inference, and Learning Algorithms. New York: Cambridge University Press, 2003.

[4] F. Rosenblatt, Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Washington DC: Spartan Books, 1962.

[5] H. Park, S. Amari, and K. Fukumizu, ―Adaptive natural gradient learning algorithms for various stochastic models,‖ Neural Networks, vol. 13, no. 7, pp. 755–764, 2000.

[6] James A. Freeman David M. Skapura, Neural Networks Algorithms, Applications, and Programming Techniques, Addison-Wesley Publishing Company (1991)

[7] Jinwook Go, Gunhee Han, Hagbae Kim Multigradient: A New Neural Network Learning Algorithm for Pattern Classification IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, VOL. 39, NO. 5, MAY 2001

[8] Kenji Fukumizu, Shun-ichi Amari Local Minima and Plateaus in Hierarchical Structures of Multilayer Perceptrons Brain Science Institute The Institute of Physical and Chemical Research (RIKEN) E-mail: ffuku,amarig@brain.riken.go.jp Oct 22, 1999

[9] Kavita Burse, Manish Manoria, Vishnu P. S. Kirar Improved Back Propagation Algorithm to Avoid Local Minima in Multiplicative Neuron Model World

All rights reserved by www.ijsrd.com

Academy of Science, Engineering and Technology 72 2010
[10] M. Abramowitz and I. A. Stegun, Eds., Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. Washington, DC: US Government Printing Office, 1972.
[11] M. Biehl and H. Schwarze, ―Learning by online gradient descent," Journal of Physics, vol. A, no. 28, pp. 643–656, 1995.
[12] N. Murata, ―A statistical study of on-line learning," in On-line Learning in Neural Networks, D. Saad, Ed., pp. 63–92. New York: Cambridge University Press, 1999.
[13] N. M. Nawi, M. R. Ransing, and R. S. RansingAn Improved Learning Algorithm based on the Conjugate Gradient Method for Back Propagation Neural NetworksInternational Journal of Applied Science, Engineering and Technology www.waset.org Spring 2006
[14] R. Rojas, Neural Networks, ch. 7. New York: Springer-Verlag, 1996.
[15] RIKEN Brain Science Institute(RIKEN BSI) Japan http://www.brain.riken.jp/
[16] R. A. Fisher, ―On the mathematical foundations of theoretical statistics," Philosophical Transactions of the Royal Society of London, vol. 222, pp. 309–68, 1922.
[17] S. Amari, ―Neural learning in structured parameter spaces — natural riemannian gradient," in Advances in Neural Information Processing Systems, M. C. Mozer, M. I. Jordan, and T. Petsche, Eds., vol. 9, p. 127. Cambridge, MA: The MIT Press, 1997.
[18] S. Amari, ―Natural gradient works efficiently in learning," Neural Computation, vol. 10, no. 2, pp. 251–276, 1998.
[19] S. Amari, H. Park, and T. Ozeki, Geometrical singularities in the neuromanifold of multilayer perceptrons, no. 14. Cambridge, MA: MIT Press, 2002.
[20] S. Amari and H. Nagaoka, Methods of Information Geometry, Translations of Mathematical Monographs, vol. 191. New York: Oxford University Press, 2000.
[21] Simon Haykin Neural Networks A comprehension foundation Pearson education seventh edition (2007)
[22] T. Heskes and B. Kappen, ―On-line learning processes in artificial neural networks," in Mathematical Foundations of Neural Networks, J. Taylor, Ed., pp. 199–233. Amsterdam, Netherlands: Elsevier, 1993.
[23] Todd K. Moon and Wynn Stirling Mathematical Methods and Algorithms for Signal Processing, Prentice Hall, 1999
[24] Weixing, Xugang, Zheng TANG Avoiding the Local Minima Problem in Backpropagation Algorithm with Modified Error Function IEICE TRANS. FUNDAMENTALS, VOL.E88–A, NO.12 DECEMBER 2005

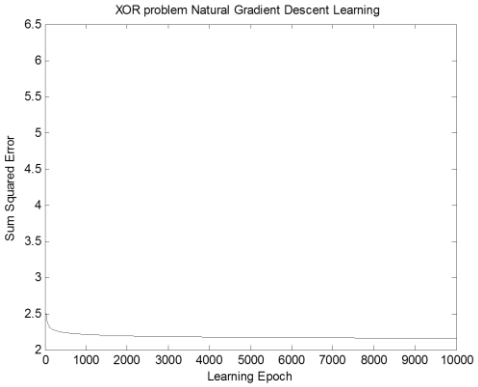

Fig. 1: The Sum Squared Error of NGL

.

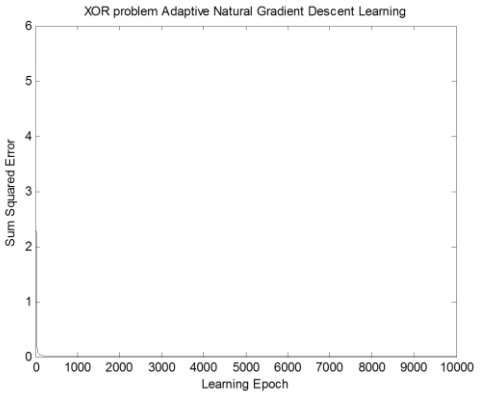

Fig. 2: The Sum Squared Error of ANGL.

All rights reserved by *www.ijsrd.com*