2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Impact of Ratio k on Two-layer Neural Networks with Dynamic Optimal Learning Rate

Zhang Tong, Student member, IEEE, and C. L. Philip Chen, Fellow, IEEE, Zhou Jin, member, IEEE,

Abstract—**Learning process is an important part in two-layer networks. It is imperative to search for an optimal learning rate to get a maximum error reduction in each learning step. Related literature has proposed various kinds of methods to find such an optimal learning rate in the past decades. In this paper, we proposed an improved dynamic optimal learning rate by adding an optimal ratio k. It is found that our improved dynamic optimal learning rate can generate a better result in learning processes. Meanwhile, we have proved the existence of the ratio $k$by giving it a proper math expression. Furthermore, we also applied the improved learning rate to solve inverse problem and compared the difference of the improved learning rate with the previous approach. It is observed that our proposed method performs better. Therefore, it can be concluded that our new method to search for dynamic optimal learning rate is valuable in the intelligence learning applications of neural networks, or it is effective in the aspect of tested problem at least.**

I. Introduction

THE two-layer neural networks (NN) were inspired from biological modeling of human's brain. Many literatures reported in designing artificial neural networks from a wide range of disciplines [1]. It has been applied to various fields, such as pattern matching, optimization, control, classification, and function approximation [2-5]. Generally, the networks include two phases. The first is to calculate the actual outputs from the input layers by applying the activation function. The second is to train the neuron weights by back propagation (BP). Adjusting the weight degree of neuron by using the descent gradient method can be achieved [6, 7]. And the general procedure of this method is to carry out continuous iterative of the two phrases by minimizing the total squared error of the actual outputs and desired outputs until reaching to stop conditions.

Classical BP methods are mainly to find the local optimal solution of optimized function in negative gradient direction. The learning rates were always fixed empirical values. However, there are mainly two disadvantages, one is the learning or convergence process is very slowly while the other is the learning process may lead to local minimum. However, the effects of learning rates on the training process are very huge in various kinds of learning applications and these differences can even lead to the failure of the process [8, 9]. Therefore, many researches are devoted to adjusting the learning rates of BP algorithms to improve the learning speed [7, 9-11].

For example, the learning rate was optimized by using derivative information. In [10], the authors introduced a dynamic changing of learning rates for a neuron set of input. Then, the dynamic learning rate was found in [10, 14] and was proven that must be greater than zero. In [13], we have proposed the new learning rate by adding a ratio $k, (0 < k < 1)$, which improved the efficacy of the training process.

In this paper, we propose a more precise function of this ratio k and it is proved effectively existing in any kind of learning processes. After multiply calculating, we find the proper math expression of ratio k from a differential equation. And, we have proved the math expression of ratio k has a unique solution, which ranges from 0 to 1. Finally, we present the performance of the improved optimal learning rate on solving the inverse problem.

The following parts of this paper are organized as below. In section II, we briefly review the two-layer neural network and dynamic optimal learning rate. Section III presents the idea of the new dynamic optimal learning rate with ratio k and the mathematical expression, along with its detailed proofs of ratio k. The experimental results are showed and analyzed by using the proposed learning method to solve reverse problems, compared with the previous learning rate at the same time in Section IV. Section V concluded the paper.

II. Preliminary

A. Two-layer Neural Network

Two-layer NN plays a crucial role in theory and practice of artificial intelligence fields. It includes interconnected processing elements, known as nodes or neurons which can work together efficiently. Fig. 1 shows the basic structure of a two-layer NN.
In the input and output layers,

$$\vec{r} = [r_1, r_2, \cdots, r_L]^T \in R^L \quad (1)$$

$$\vec{y} = [y_1, y_2, \cdots, y_Z]^T \in R^Z \quad (2)$$

where "T" is the transpose operator.
W stands for the weighting matrix,

$$W = [\overrightarrow{w_1}, \overrightarrow{w_2}, \cdots, \overrightarrow{w_Z}]^T \in R^{L\times Z} \quad (3)$$

Then, for the desired output, refer to Equation (4),

T. Zhang is with the Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau, China (e-mail: ztony86@gmail.com).
C. L. Philip Chen is with the Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau, China (phone: 853-8397-4950; fax: 853-2883-8314; e-mail: Philip.Chen@ieee.org).
J. Zhou is with the Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau, China (e-mail: ise.zhouj@gmail.com).
This work was supported by the Macau Science and Technology Development Fund under Grant No.008/2010/A1 and University of Macau Multi-Year Research Grants.

$$\vec{d} = [d_1, d_2, \cdots, d_Z]^T \in R^Z \tag{4}$$

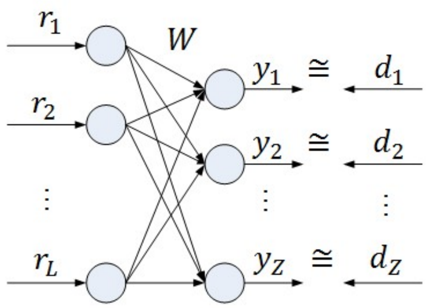

Fig. 1. Basic structure of two-layers ANN

According to inter relations of the nodes or neurons, we get the following equation,

$$y_Z = \sum_{l=0}^{L} r_l w_{Zl} = \vec{r}^T \overrightarrow{w_z} \tag{5}$$

In mathematical way, we assume both the input and output elements are notated in matrix, if the training dataset included P samples, we can get the input matrix as Equation (6)

$$R = [\underline{r}_1, \underline{r}_2, \dots, \underline{r}_P] \in \mathbb{R}^{L \times P} \tag{6}$$

The actual output can be obtained from Equation (7),

$$Y = R^T W \tag{7}$$

Matrix D stands for the desire output, which have a same dimension with Y.

Hence, the total square error (TSE) can be described as Equation (8), which is a function of W.

$$J(W) = \frac{1}{2P \times Z}(y_z - d_z)^T (y_z - d_z) \tag{8}$$

Then, the learning process can be summarized to reduce the value of $J(W)$ by updating the weighting matrix W.

B. Dynamic Optimal Learning Rate

The dynamic optimal learning rate was proposed in paper [9]. Compared with the classical approach by using gradient descent method, the authors construct an iterative process to train the weight W in a two-layer NN, which is expect to converge to the optimal one. Refer to Equation (9).

$$W_{t+1} = \mathrm{W}_t - \beta_t \frac{\partial J(W_t)}{\partial W_t}\bigg|_{\mathrm{W}_t} \tag{9}$$

where t is the iteration number and β_t is the dynamic optimal learning rate. Applying the chain rule, differential part in Equation (9) can be calculated as Equation (10),

$$\frac{\partial J(W)}{\partial W} = \frac{1}{P \cdot Z} R(R^T W - D) \tag{10}$$

Then, after solving the quadratic polynomial $a\beta^2 + b\beta$ to get the minimum solution by applying the theorem 1 in [5], we can obtain the optimal learning rate from Equation (11),

$$\beta_{opt} = -\frac{b}{2a} \tag{11}$$

where,

$$a = \frac{1}{2(P \cdot Z)^3} R^T R E_t E_t^T R^T R \tag{12}$$

$$b = \frac{1}{2(P \cdot Z)^3} R^T R E_t E_t^T R^T R \tag{13}$$

$$E = R^T W - D \tag{14}$$

The details of the training process can be referred to algorithm I in [8]

III. Analysis of Ratio k of Dynamic Optimal Learning Rate

In this part, the new dynamic learning rate with ratio k is shown. Then, we analyze its proper math expression.

A. New Dynamic Learning rate with Ratio k

Assume that the initial weighting matrix W_0 has been calculated at the beginning. The corresponding total square error was denoted as J_0. Then, the optimal learning rate β_0 can be obtained from theorem 1 in [10]. Instead of applying Equation (15) in [10], a ratio k will be added to β_0. So the W_1 can be got from Equation (15),

$$W_1 = W_0 - k\beta_0 \frac{\partial J(W_0)}{\partial W_0} \tag{15}$$

where,

$$\beta_0 = \frac{PZ(Tr[E_0{}^T R^T R E_0])}{Tr[R^T R E_0 E_0{}^T R^T R]} \tag{16}$$

Then, the error function is,

$$\begin{aligned} E_1 &= R^T W_1 - D \\ &= E_0 - k\beta_0 R^T R E_0 (PZ)^{-1} \end{aligned} \tag{17}$$

For the next iteration, we can get,

$$E_2 = E_0 - k\beta_0 \frac{R^T R E_0}{PZ} - \beta_1 \frac{R^T R E_0}{PZ} + k\beta_0 \beta_1 \frac{R^T R R^T R E_0}{(PZ)^2}$$

The optimal learning rate β_1 in the next generation can be calculated as Equation (18)

$$\begin{aligned} \beta_1 &= \frac{PZ(Tr\{[E_0 - k\beta_0 R^T R E_0 (PZ)^{-1}] R^T R [E_0 - k\beta_0 R^T R E (PZ)^{-1}]}{Tr\left[R^T R E_1 E_1{}^T R^T R\right]} \\ &= \frac{PZ\left\{Tr\left[E_0{}^T R^T R E_0 - 2(PZ)^{-1} k\beta_0 E_t{}^T R^T R R^T R E_t + (PZ)^{-2} k^2 \beta_0{}^2 E_t{}^T R^T R R^T R R^T R E_t\right]\right\}}{Tr\left[R^T R E_{t+1} E_{t+1}{}^T R^T R\right]} \\ &= \frac{PZ \cdot Tr[E_0{}^T R^T R E_0] - 2k\beta_0 Tr[E_0{}^T R^T R R^T R E_0] + (PZ)^{-1} k^2 \beta_0{}^2 Tr[E_0{}^T R^T R R^T R R^T R E_0]}{Tr[E_0{}^T R^T R R^T R E_0] - (PZ)^{-1} 2k\beta_0 Tr[E_0{}^T R^T R R^T R R^T R E_0] + (PZ)^{-2} k^2 \beta_0{}^2 Tr[E_0{}^T R^T R R^T R R^T R R^T R E_0]} \end{aligned} \tag{18}$$

For the sake of writing conveniently, we denote five new intermediate variables $P_0, P_1, P_2, P_3, and\ P_4$, expressed as Equation (19) to Equation (23) as below,

$$P_0 = Tr[{E_0}^T E_0] \tag{19}$$

$$P_1 = Tr[{E_0}^T R^T R E_0] \tag{20}$$

$$P_2 = Tr[{E_0}^T R^T R R^T R E_0] \tag{21}$$

$$P_3 = Tr[{E_0}^T R^T R R^T R R^T R E_0] \tag{22}$$

$$P_4 = Tr[{E_0}^T R^T R R^T R R^T R R^T R E_0] \tag{23}$$

Then, we get Equation (24) from (18).

$$\beta_1 = \frac{PZ \cdot P_1 - 2k\beta_0 P_2 + (PZ)^{-1} k^2 {\beta_0}^2 P_3}{P_2 - (PZ)^{-1} 2k\beta_0 P_3 + (PZ)^{-2} k^2 {\beta_0}^2 P_4} \tag{24}$$

The total square error $J(W_2)$ can also be calculated as Equation (25),

$$J(W_2) = \frac{1}{2P \cdot Z} Tr\left(Q_1 {Q_1}^T\right) \tag{25}$$

Where,

$$Q_1 = \quad E_0 - k\beta_0 \frac{R^T R E_0}{PZ} - \beta_1 \frac{R^T R E_0}{PZ} + \ k\beta_0\beta_1 \frac{R^T R R^T R E_0}{(PZ)^2}$$

$${Q_1}^T = {E_0}^T - k\beta_0 \frac{{E_0}^T R^T R}{PZ} - \beta_1 \frac{{E_0}^T R^T R}{PZ} + \ k\beta_0\beta_1 \frac{{E_0}^T R^T R R^T R}{(PZ)^2}$$

In order to find the minimum of $J(W_2)$, we differentiate $J(W_2)$ with respect to k and let it equal to zero, then we can solve the equation. After multiply calculating from MAPLE or Mathematica, we find three roots (Equation 26-28) of the differential equation. However, only the root of equation (26) is ranged from 0 to 1.

$$k_1 = \frac{(P_1P_4 - P_2P_3 - Q_2) \cdot PZ}{4(P_2P_4 - {P_3}^2)\beta_0} \tag{26}$$

$$k_2 = \frac{(P_1P_4 - P_2P_3 + Q_2) \cdot PZ}{4(P_2P_4 - {P_3}^2)\beta_0} \tag{27}$$

$$k_3 = \frac{PZ\sqrt[3]{-12P_2P_3P_4 + 4P_1{P_4}^2 + 8{P_3}^3 + Q_2}}{8\left(P_2P_4 - {P_3}^2\right){\beta_0}^2} - \frac{2PZ\left(P_2P_4 - {P_3}^2\right)}{2\beta_0\sqrt[3]{-12P_2P_3P_4 + 4P_1{P_4}^2 + 8{P_3}^3 + Q_2}} + \frac{PZ \cdot P_3}{8\beta_0 P_4} \tag{28}$$

Here, Q_2 is an intermediate variable and

$$Q_2 = \sqrt{4{P_2}^3P_4 - 3{P_2}^2{P_3}^2 - 6P_1P_2P_3P_4 + {P_1}^2{P_4}^2 + 4P_1{P_3}^3}$$

B. *Math Expression of Ratio k*

Firstly, we define a function $\mathcal{F}$ with respect to k,

$$\mathcal{F}(k) = \ J(W_2) - \ J(\widehat{W}_2)$$

Obviously, if k=1, there is no difference between $J(W_{t+2})$ and $J(\widehat{W}_{t+2})$, i.e. $\mathcal{F}(1) = 0$. We assume that when k is satisfied with the Equation (26), we get the minimum of $\mathcal{F}(k)$. It should satisfy the following two conditions,

$$\mathcal{F}(k) < 0$$

$$\mathcal{F}''(k) > 0$$

We sub Equation (26) to $J(\widehat{W}_2)$ and $J'(\widehat{W}_2)$ respectively. Then, we have,

$$J(\widehat{W}_2) = \frac{{P_2}^3 - P_0P_2P_4 - 2P_1P_2P_3 + {P_1}^2P_4 + P_0{P_3}^2}{2PZ(P_2P_4 - {P_3}^2)} \tag{29}$$

$$J(W_2) = \frac{1}{2P^3Z^3\left(P_2P_4{P_3}^2 - 4PZ\beta_0P_3 + 4{\beta_0}^2P_4\right)} \left(P^4Z^4P_0P_2 - P^4Z^4{P_1}^2 + 4P^3Z^3\beta_0P_1P_2 - 4P^3Z^3\beta_0P_0P_3 - 12P^2Z^2\beta_0{P_2}^2 + 8P^2Z^2\beta_0P_1P_3 + 4P^2Z^2{\beta_0}^2P_0P_4 + 16PZ{\beta_0}^3P_2P_3 - 16PZ{\beta_0}^3P_1P_4 + 16{\beta_0}^4P_2P_4 - 16{\beta_0}^4{P_3}^2\right) \tag{30}$$

So,

$$\mathcal{F}(k) = -\frac{\left(P^2Z^2{P_2}^2 - 4{\beta_0}^2P_2P_4 + 2PZ\beta_0P_1P_4 - 2PZ\beta_0P_2P_3 - P^2Z^2P_1P_3 + 4{\beta_0}^2{P_3}^2\right)^2}{2P^3Z^3\left(P_2P_4 - {P_3}^2\right)\left(P^2Z^2P_2 - 4PZ\beta_0P_3 + 4{\beta_0}^2P_4\right)} \tag{31}$$

Reviewing the previous Equations (19-23), it is obviously that the numerator of Equation (31) is non-negative. For ${E_0}^T R^T R$ and $R^T R E_0$, ${E_0}^T R^T R R^T R$ and $R^T R R^T R E_0$ are both real symmetric matrices.

Let $A = {E_0}^T R^T R$, $A^T = R^T R E_0$, $B^T = {E_0}^T R^T R R^T R$, $B = RTRRTRE0$. Then, we have the following proof by applying Cauchy–Schwarz inequality,

$$\begin{aligned} P_3 &= Tr[A^T B] \\ &= Tr[{E_0}^T R^T R R^T R R^T R E_0] \\ &\le (Tr[A^T A])^{\frac{1}{2}} (Tr[B^T B])^{\frac{1}{2}} \\ &= {P_2}^{\frac{1}{2}} {P_4}^{\frac{1}{2}} \end{aligned}$$

That is ${P_3}^2 \le P_2P_4$, i.e. $P_2P_4 - {P_3}^2 \ge 0$. But the input samples cannot be one. Therefore, the equality cannot holds.

For $P^2Z^2P_2 - 4PZ\beta_0P_3 + 4{\beta_0}^2P_4$ is a quadratic function of β_0.

$$\begin{aligned} \Delta &= 16P^2Z^2{P_3}^2 - 16P^2Z^2P_2P_4 \\ &= -16P^2Z^2\left(P_2P_4 - {P_3}^2\right) \\ &< 0 \end{aligned}$$

For all value of β_0, $\mathcal{F}(k)$ is negative. In other words, the assumption of the existence of ratio k is true, and the proper math expression is Equation (26).

IV. Experimental Results

In order to show the impact on TSE of ratio k in a learning process, we generalize the following experiment with random input, weight matrix and output. Then, two experiments were carries out by applying algorithm I in [10] and algorithm 2 in

[13].

Random Test Problem:

Input layer,

$$R = \begin{bmatrix} 0.5201 & -0.7982 & -0.7145 & -0.5890 \\ -0.0200 & 1.01869 & 1.35139 & -0.2938 \\ -0.0348 & -0.1332 & -0.2248 & -0.8479 \end{bmatrix}$$

The desire output D,

$$D = \begin{bmatrix} -1.1201 & -1.2571 \\ 2.5260 & -0.8655 \\ 1.6555 & -0.1765 \\ 0.307535 & 0.79142 \end{bmatrix}$$

The initial weighting matrix W,

$$W = \begin{bmatrix} -1.332 & 0.33351 \\ -2.3299 & 0.39135 \\ -1.4491 & 0.45168 \end{bmatrix}$$

Then we find the performance of TSE with different ratio , which can be illustrated in Fig. 2,

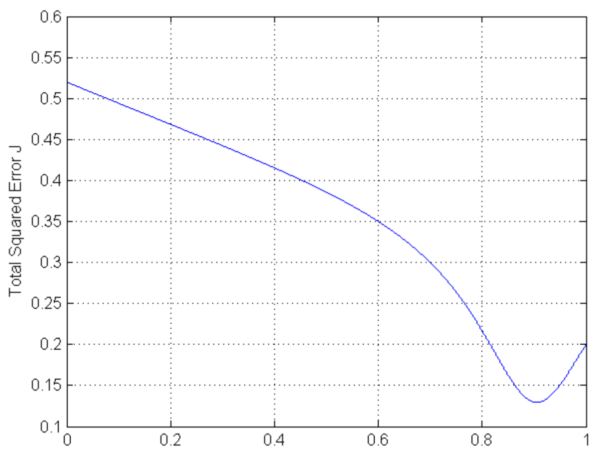

Fig. 2. The TSE value associate with different ratio k.

From the calculation

$$P_0 = 36.8982$$
$$P_1 = 108.2793$$
$$P_2 = 410.2772$$
$$P_3 = 1.6671 \times 10^3$$
$$P_4 = 6.9155 \times 10^3$$

Then, the ratio k can be obtained from Equation (26),

$$k = 0.9056$$

Solving Inverse problem:

Suppose we have a convolution equation as Equation (32),

$$\mathrm{g(x)} = \int_0^1 k(x - x')\, f(x') dx, 0 < x < 1 \qquad (32)$$

where $k(x)$ is a Gaussian kernel function,

$$k(x) = \frac{1}{\gamma\sqrt{2\pi}} e^{-\frac{x^2}{2\gamma^2}}, \gamma = 0.05 \qquad (33)$$

$f(x)$ is the initial function presented in Equation (34) .

$$f(x) = \begin{cases} 0.75 & 0.1 < x < 0.25 \\ 0.25 & 0.3 < x < 0.32 \\ sin^4(2\pi x) & 0.5 < x < 1 \\ 0, & \text{other wise} \end{cases} \qquad (34)$$

It can be shown in Fig. 3,

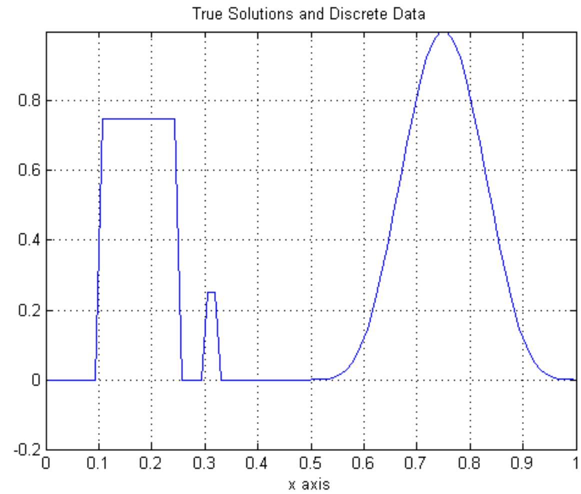

Fig. 3. The initial function f.

The kernel function can be processed in a discrete way and we can get the following equation,

$$[K]_{ij} = \frac{h}{\gamma\sqrt{2\pi}} e^{-\frac{((i-j)h)^2}{2\gamma^2}}, \gamma = 0.05, 1 \le i, j \le n \qquad (35)$$

Then, we plot the surface graphics of matrix K as in Fig. 4,

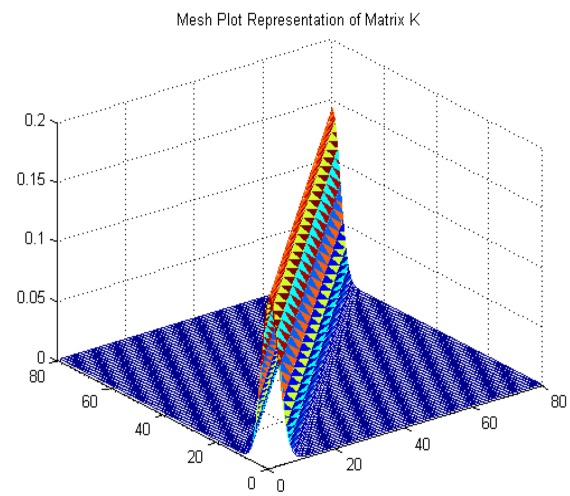

Fig. 4. Surface graphics of matrix K

We can apply the NN model to approximate the original equation as Equation (31), and solve the following inverse equation to get fitted curve in Fig. 4. In other words, we aim to approximate the distribution of K by a function from f as closely as possible [15, 16].

$$f = K^{-1}d \qquad (36)$$

The results can be shown as in Fig. 5.

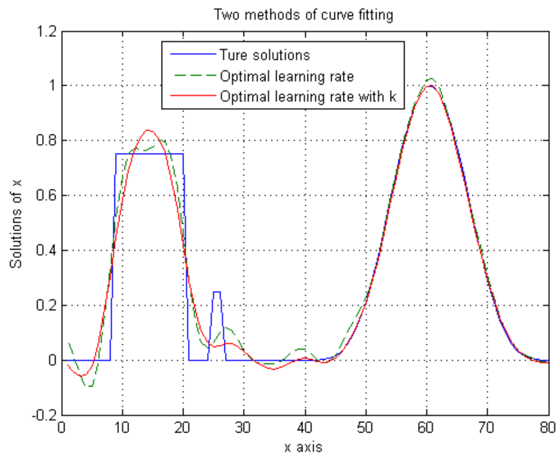

Fig. 5. The fitted curves of initial function by comparing the two kinds of optimal learning rate.

The TSE of our approach is 0.0041 with 6 iterations, while the previous method is 0.0096 with 10 iterations. It can be summarized that our approach is a better solution with less iteration in solving this inverse problem.

V. Conclusion

Based on the optimal learning rates in previous work, we proposed a new dynamic optimal learning rate with ratio k. And we can contribute to the prior work by giving the ratio k a proper math expression. After multiple calculations, three roots of the ratio k were obtained. Then, we analyzed the rationality of these solutions and further precisely proving its range is between 0 and 1.

Besides, our new method performs better both in the convergence speed and training accuracy in two experiments. One experiment is designed of using randomly generated training data. The other is designed of solving inverse function in kernel space. Our proposed method is preferably fitting and stability for the testing case. Therefore, we can draw the conclusion that the new dynamic optimal learning rate with ratio k is highly efficient, and it performs better than the previous work in neural network applications.

Our future work will focus on solving much broader range of related applications and compare the efficiency with the other methods (standard BP, Levenberg–Marquardt algorithm (LMA), and other gradient descent methods).

References

[1] Kemal Akkaya, Mohamed Younis, A survey on routing protocols for wireless sensor networks, Ad Hoc Networks, Volume 3, Issue 3, May 2005, Pages 325-349

[2] Phung, S.L.; Bouzerdoum, A., "A Pyramidal Neural Network For Visual Pattern Recognition," *IEEE Trans. Neural Networks,* vol.18, no.2, pp.329, 343, March 2007.

[3] Ching-Chih Tsai; Hsu-Chih Huang; Shui-Chun Lin, "Adaptive Neural Network Control of a Self-Balancing Two-Wheeled Scooter," *IEEE Trans. Industrial Electronics*, vol.57, no.4, pp.1420-1428, April 2010.

[4] Tong Zhang, Sik Chung Tam, Chi-Hsu Wang, C. L. Philip, Chen, "On the classification of cancer cell gene via Expressive Value Distance (EVD) algorithm and its comparison to the optimally trained ANN method," Fuzzy Systems (FUZZ), 2011 IEEE International Conference on , vol., no., pp.2199,2204, 27-30 June 2011.

[5] Jianbo Yu, Shijin Wang, Lifeng Xi, Evolving artificial neural networks using an improved PSO and DPSO, Neurocomputing, Volume 71, Issues 4–6, January 2008

[6] Shian-Tang Tzeng, "Design of fuzzy wavelet neural networks using the GA approach for function approximation and system identification", Fuzzy Sets and Systems, Volume 161, Issue 19, 1 October 2010, Pages 2585-2596.

[7] G. O. Young, "Synthetic structure of industrial plastics (Book style with paper title and editor)," in *Plastics*, 2nd ed. vol. 3, J. Peters, Ed. New York: McGraw-Hill, 1964, pp. 15–64.

[8] Magoulas G D, Vrahatis M N, Androulakis G S. Improving the convergence of the back propagation algorithm using learning rate adaptation methods. Neural Computation, 1999, 11(7): 1769-1796.

[9] Yunong Zhang, Dongsheng Guo, Zhan Li, "Common Nature of Learning Between Back-Propagation and Hopfield-Type Neural Networks for Generalized Matrix Inversion With Simplified Models," *IEEE Trans. Neural Networks and Learning Systems*, vol.24, no.4, pp.579,592, April 2013.

[10] Wang C H, Liu H L, Lin C T. Dynamic optimal learning rates of a certain class of fuzzy neural networks and its applications with genetic algorithm. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 2001, 31(3): 467-475.

[11] Xiao-Hu Yu, Guo-An Chen, and Shi-Xin Cheng, "Dynamic learning rate optimization of the back propagation algorithm," *IEEE Trans. Neural Networks*, vol. 6, pp. 669–677, May 1995.

[12] Ming-Jen Kuo, Tsung-Chih Lin, "Dynamical optimal training for behavioral modeling of nonlinear circuit elements based on radial basis function neural network," Electromagnetic Compatibility and 19th International Zurich Symposium on Electromagnetic Compatibility, 2008. APEMC 2008.

[13] Tong Zhang, C. L. Philip Chen, Chi-Hsu Wang, and Sik Chung Tam, "A New Dynamic Optimal Learning Rate for A Two-Layer Neural Network", System Science and Engineering (ICSSE), 2012 International Conference on, Dalian, China, June 30-July 2, 2012.

[14] C.H. Wang, C.S. Cheng, T. T. Lee, "Dynamical optimal training for interval type-2 fuzzy neural network (T2FNN)," *IEEE Trans. Systems Man and Cybernetics: Part B* 34 (3), 1462–1477,2004.

[15] Mendelson S, Neeman J., "Regularization in kernel learning". The Annals of Statistics, 2010, 38(1): 526-565.

[16] Wang Shaowei, Wan Shanming, "Identify PMSM's Parameters by Single-Layer Neural Networks with Gradient Descent," Electrical and Control Engineering (ICECE), 2010 International Conference on , vol., no., pp.3811,3814, 25-27 June 2010.