Journal of Multidisciplinary Modeling and Optimization 1(1) (2018), 14-22.

Training Feed Forward Neural Network with Modified Fletcher-Reeves Method

Yoksal A. LAYLANI[1], Khalil K. ABBO[2] and Hisham M. KHUDHUR[2]

[1] Department of Mathematics,
Kirkuk University, Kirkuk, Iraq
yoksalalattar@yahoo.com

[2] Department of Mathematics,
Mosul University, Mosul, Iraq
kh_196538@yahoo.com
hisham892012@gmail.com

Abstract *— In this research, a modified Fletcher-Reeves (FR) conjugate gradient algorithm for training large scale feed forward neural network (FFNN) is presented. Under mild conditions, we establish that the proposed method satisfies the sufficient descent condition, and it is globally convergent under Wolfe line search condition. The evidence which is provided by experimental results showed that our proposed method is preferable and superior to the classic methods.*

Keywords: Unconstrained optimization problem, Conjugate gradient method, Wolfe condition, Globally convergent ..
Mathematics Subject Classification: 65K10, 80M50, 90C26.

1 Introduction

Multi layered feed forward neural networks have received a great deal of attention in many research and applications areas [1, 2, 3]. A critical factor in the use of this technology is the training. Off-line training of multilayer perceptions and FFNs has progressed considerably since the development of the batch back – propagation algorithm (BP) [4]. It is well known that training a neural network problem is highly consistent with the unconstrained optimization theory [5]. More analytically, training a neural network problem' can be formulated as the minimization of the error function E(W) that depends on the connection weights W of the network, defined as the sum of squares of the error in the outputs[6] i.e. the operation of the (FFN) is usually based on the following equations :

$$net_j^l = \sum_{i=l}^{N_{l-1}} w_{i,j}^{l-1} x_j^{l-1} + b_j^l, \quad O_j^l = f(net_j^l) \tag{1}$$

Where $f(net_j^l)$ is the activation function , net_j^l is the sum of the weight inputs for the $j-th$ node in the $l-th$ layer $(j = 1,2,\dots,NI)$, $w_{i,j}$ is the weight from the $i-th$ neuron to the $j-th$ neuron at the $l-1, l-th$ layer ,respectively, b_j^l is the bias of the $j-th$ neuron at the $l-th$ layer and x_j^l is the output of the $j-th$ neuron which be-

longs to the $l-th$ layer. The weight of training a neural network problem' iteratively adjusts in order to minimize the difference between the actual output of the network and the desired output of the training set [7]. Actually finding such minimum is equivalent to find an optimal minimization of the error function which defined by:

$$E(w) = \frac{1}{2}\sum_{j=1}^{P}\sum_{i=1}^{M}(O_i^{(j)} - T_i^{(j)})^2 \quad (2)$$

The variables O_i and T_i are the desired and the actual output of the $i-th$ neuron, respectively. The index j denotes the particular learning pattern. The vector w is consist of all weights in the network. Back propagation (BBP) algorithm is the most widely used to train multilayer feed forward neural networks. The standard back propagation algorithm adjusts the weight vector w using steepest descent with respect to E such that :

$$w_{k+1} = w_k - \alpha_k d_k, \qquad d_k = \nabla E(w_k) \quad (3)$$

Where the constant α is the learning rate belongs to the interval (0,1) and w_k is a vector representing the weights at iteration (epoch) step k. Since the steepest descent method has slow convergence rate, and since the search for the global minimum often becomes trapped at a poor local minimum, then this what implies that the back propagation algorithm takes unendurable time to adapt the weights between the units in the network. For this reason, many researches proposed to improve this algorithm see [8, 9, 10, 11].

This paper organized as following. In section 2, we present our proposed modified Fletcher-Recues conjugate gradient training algorithm and in section 3, we present its global convergence analysis. The experimental results reports' in section 4. Finally, section 5 presents our concluding remarks.

2 The Methods of Conjugate Gradient (CG)

The linear combination of negative gradient vector is the basic idea for determining the search direction in conjugate gradient methods at the current iteration with the previous search direction that is:

$$d_{k+1} = \begin{cases} -g_{k+1} & \text{if } k = 0 \\ -g_{k+1} + \beta_k d_k & k \geq 1 \end{cases} \quad (4)$$

Where $g_k = \Delta E(W_k)$ and β_k is the (CG) update parameter. The first CG algorithm for non-convers problems was proposed by Fletcher and Reeves (FR) in 1969 [12] which defined β_k as

$$\beta^{FR} = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k} \quad (5)$$

The FR method's numerical performance is somewhat erratic [13]. It is efficient sometimes, but it is over all slower. Powell [14] gives an argument showed that the FR method with exact line searches $(g_{k+1}^T d_k = 0)$, under some circumstances, will produce very small displacements i.e. In some cases $g_{k+1}^T g_k \approx g_h^T g_h$ which leades to d_{k+1} become linear combination with the d_k which is the main drawbacks for the FR method.

In the convergence analysis and implementations of CG method, one often requires the inexact line search (learning rate) as the Wolfe line search. The standard Wolfe line search requires α_k in equation (3) satisfies the following conflations

$$E(W_n + \alpha_k d_k) \le E(W_h) + \rho\, \alpha_k\, g_k^T d_k \quad (6)$$

$$g_k^T d_k \ge \sigma g_k^T d_k\,, \quad (7)$$

where $0 < \rho < \sigma < 1$.

2.1 Modified FR Method

In the following we suggest a modification to the FR method to avoid the drawback $\left(g_{k+1}^T g_k \approx g_h^T g_h\right)$ of FR algorithm. We can define the new CG update parameter β_{k+1}^{MFR} as follows:

$$\beta_{k+1}^{MFR} = \frac{g_{k+1}^T g_{k=1}}{g_{k+1}^T g_k + M\,|d_k^T\, g_{k+1}|}\ , M > 1 \quad (8)$$

Therefore the new search direction can be written as

$$d_{k+1} = \begin{cases} -g_{k+1} & \text{if } k = 0 \\ -g_{k+1} + \beta^{MFR} d_k & k \ge 1. \end{cases} \quad (9)$$

We can prove that our proposed formula β^{MFR} satisfies the sufficient descent condition.

Theorem 1. *Let d_{u+1} defined by (8) and (9) then d_{u+1} satisfies the sufficient descent i.e.* $d_{k+1}^T\, d_{k+1} \le -c\,|g_{k+1}| + k \ge 0$.

Proof :

If k=0 then $d_1 = -g_1$ therefore

$$d_1^T\, g_1 = -|g_1|^2 < 0$$

For ≥ 1 , from (8) and (9)we have

$$\begin{aligned} d_{k+1}^T\, g_{k+1} &= -g_{k+1}^T\, g_{k+1} + \frac{g_{k+1}^T\, g_{k+1}}{g_k^T\, g_k + M|d_k^T\, g_{k+1}|}\, d_k^T\, g_{k+1} \\ &\le -|g_{k+1}|^2 + \frac{|g_{k+1}|^2}{|g_k|^2 + M|d_k^T\, g_{k+1}|}\, |d_k^T\, g_{k+1}| \\ &\le \left(-1 + \frac{|d_k^T\, g_{k+1}|}{M|d_k^T\, g_{k+1}|}\right) |g_{k+1}|^2 \\ &= \left(-1 + \frac{1}{M}\right) |g_{k+1}|^2 \end{aligned}$$

Since M>1 , we obtain

$d_k^T\, g_{k+1} \le c\,|g_{k+1}|^2$.

2.2 Modified Fletcher – Reeves conjugate gradient algorithm (MER)

Step 1: Initiate W_1 , $0 < \rho < \sigma < 1$, $E_G\,|g_{k+1}|^2$ and k_{max} ; set k=0.

Step 2: Calculate the error function value E_{k+1} and its gradient g_{k+1}.

Step 3: If $(E_{k+1} \le E_c)$ or $(|g_{k+1}|^2\ \varepsilon_1)$ return $W^{\infty} = W_{u+1}$ and $E_* = E_{u+1}$.

Step4: Compute the descent direction d_{u+1} using equations (8) and (9).

Step5: Compute the learning rate α_{k+1} using the standard Wolfe line search condition (6) and (7).

Step6: Update the weight $W_k = W_k + \alpha_k\, d_k$; set k=k+1.

Step 7: If $k > k_{n+m}$ return to 1; Error goal; not go to Step 2.

2.3 Global Converges Analysis

In order to establish the global converges result for our intended method, we will impose some assumptions on the error function E as follow:

Assumption 1: The level set $\mathcal{L} = \{w \in \mathbb{R}^2 \mid E(w) \leq E(w_0)\}$ is bounded.

Assumption 2: In some neighborhood $N \in \mathcal{L}$, E is differentiable and its gradient g is Lipchitz continuous, more specifically, there exists a positive constant L>0 such that

$$\|g(w) - g(\breve{w})\| \leq L \|w - \breve{w}\|, \forall w, \breve{w} \in N.$$

Since $\{E(w_k)\}$ is a decreasing sequence, it is clear that the sequence $\{w_k\}$ is contained in $\mathcal{L}$. In addition it follows directil from Assumptions 2.1 and 2.2 that there exist constrants B and M, such that

$$\|w - \breve{w}\| \leq B, \forall w, \breve{w} \in \mathcal{L}$$
$$\|g(w)\| \leq M, \forall w \in \mathcal{L}$$

Touati –Ahmed and story in [15] show that any conjugate gradient method with $0 < \beta \leq \beta^{FR}$ is globally convergent.

3 Experiments and Results

A computer simulation has been developed to study the performance of the following algorithms.

1- FR: Conjugate gradient back propagation with Fletcher-Reeves updates.
2- YH: New training algorithm.

The simulations have been carried out using MATLAB (7.6). The performance of the MSBP has been evaluated and compared with batch versions of the above algorithm. By using the initial weights, initialized by the Nguyen – Widrow method [19] , the algorithms were tested and received the same sequence of input patterns. The network weight's updates only when the entire set of patterns to be learned has been presented. For each of the test problems, a table summarizing the performance of the algorithms for simulations that reached solution is presented. The reported parameters are min the minimum number of epochs for 50 simulation, mean the mean value of epochs for 50 simulation, Max the maximum number of epochs for 50 simulation, Tav the average of total time for 50 simulation and Succ, the succeeded simulations out of (50) trails within error function evaluations limit. If an algorithm fails to converge within the above limit considered that it fails to train the FFNN, but its epochs are not included in the statically analysis of the algorithm, one gradient and one error function evaluations are necessary at each epoch.

1- Problem (XOR Problem)

The initial problem that we encountered with is the XOR Boolean function problem. It considers as a classical problem for the FFNN training. The XOR function maps two binary inputs to a single binary output. This function is not linearly separable. The network architectures for this binary classification problem consist of one hidden layer with 3 neurons and an output layer of one neuron. The termination criterion is set to $\varepsilon_2 \leq 0.002$ within the limit of 1000 epochs, and table (1) summarizes the result of all algorithms i.e. for 50 simulations the minimum epochs for each algorithm are listed in the first column (Min), the maximum epochs for each algorithm are listed in the second column, third column contains (Mean) the mean value of epochs and (Tav) is the average of time for 50 simulations and last columns contain the percentage of succeeds of the algorithms in 50 simulations.

The results of the simulations presented in table (1),figure (1)and(2) where the vertical axis represent error and horizontal axis represent the number of epochs.

Table 1: Results of simulations for the XOR function

Algorithms	min	max	Mean	Tav	succ
FR	4	48	9.5	0.4125	100%
YH	3	39	8.26	0.39016	100%

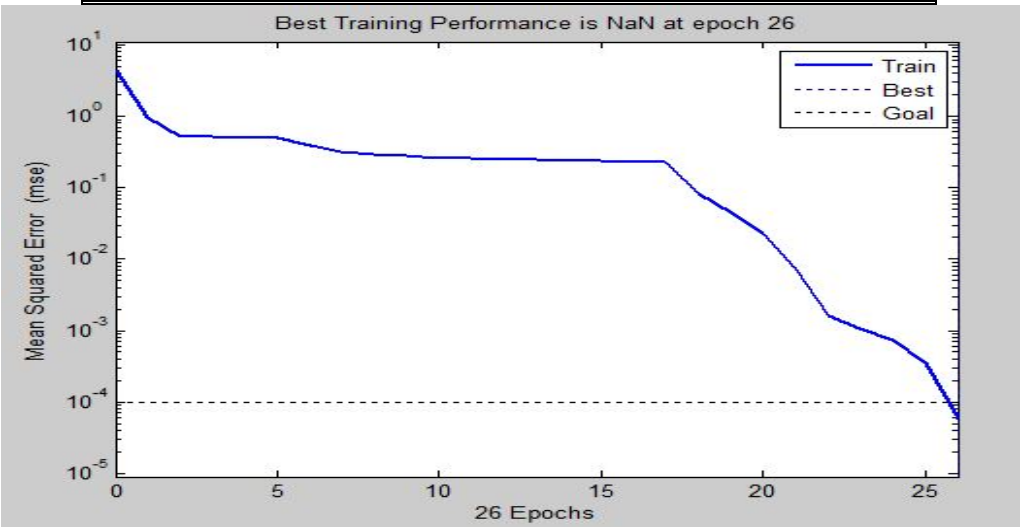

Figure 1: Mean Squared Error (MSE) for FR

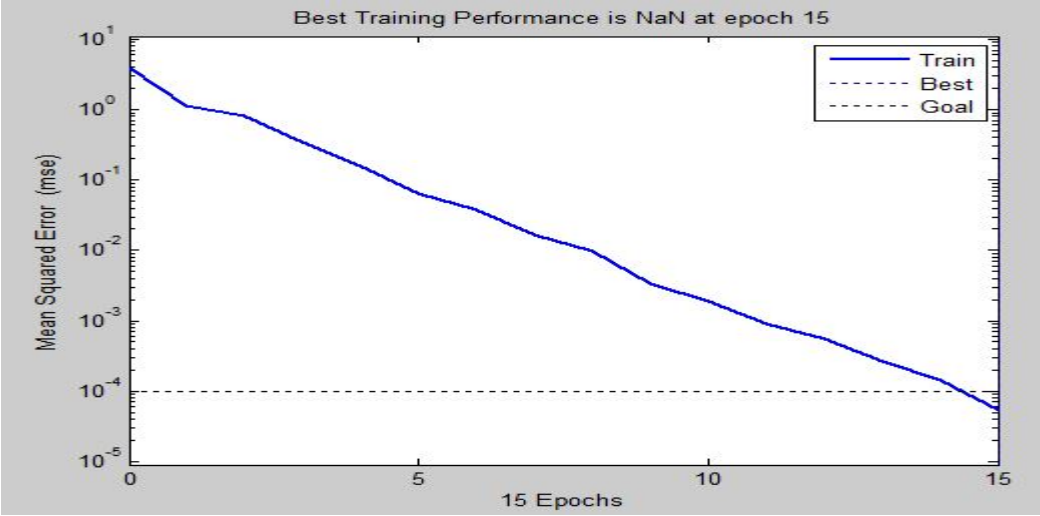

Figure 2: Mean Squared Error (MSE) for YH

2- Function Approximation Problem

The second problem we have considered is the approximation of continuous function,

$f(x) = \cos(\pi x) + 0.1\, rand(sinx)$,

where $x = -1{:}\,0.005{:}\,1$. This problem takes one real input to a single real output. The selected architecture of the FFNN is one neuron in input lager, ten neuron in hidden layer and one neuron in output neuron, with sigmoid function in hidden neuron's and a linear function in output neuron. The error goal has been let to 0.001 and the maximum epochs

to 1000. The results of the simulations presented in table (2), figure (3) and (4) where the vertical axis represent error and horizontal axis represent the number of epochs.

Table 2: Results of simulations for the Function Approximation Problem

Algorithms	min	max	Mean	Tav	succ
FR	14	44	26.34	0.5837	100%
YH	16	36	25.8	0.55132	100%

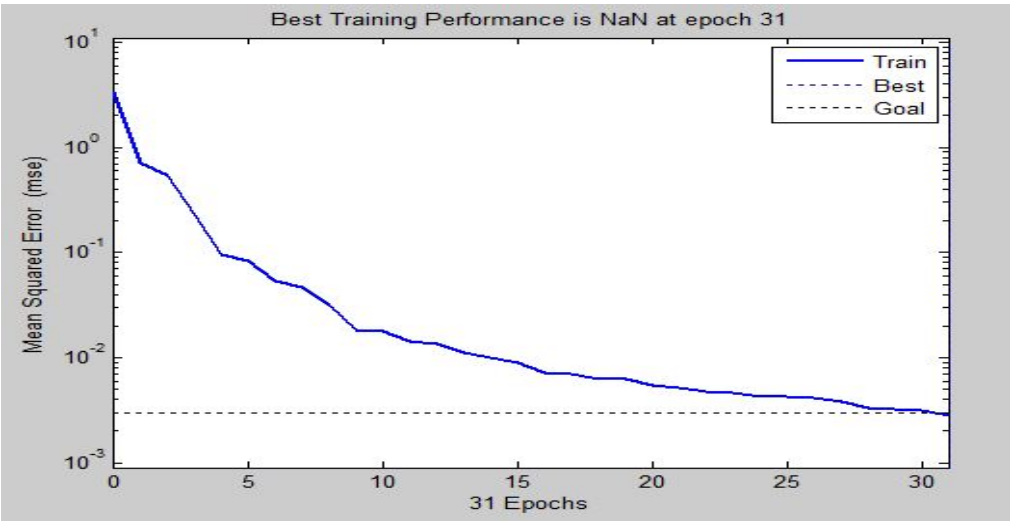

Figure 3: Mean Squared Error (MSE) for FR

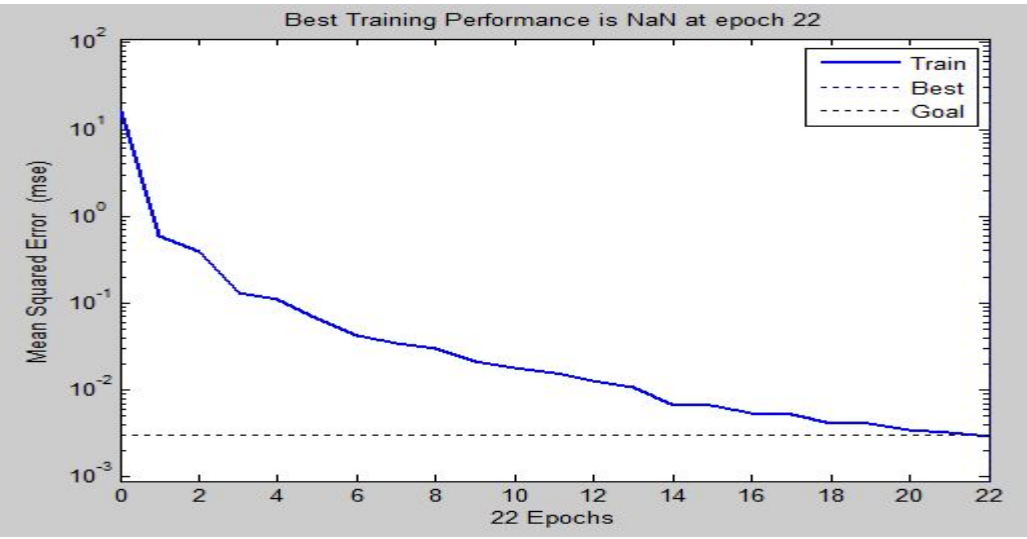

Figure 4: Mean Squared Error (MSE) for YH

3- SPECT Heart Problem

This dataset consist of data instances which derived from cardiac single proton Emission Computed Tomography (SPECT) images from the University of Colorado. Also, it is a binary classification task, where patients' heart images are classified as nor-

mal is abnormal. The class distribution has 55 instances of the abnormal class 20.6% and 212 instances of the normal class (79.4%). There have been selected 80 instances for the training process and the remainder 187 for testing the neural networks generalization capability. The network architecture for this medical classification problem constitute of 1 hidden layer with 6 neurons and an output layer of 2 neurons. The termination criterion is set to $E_{rr} \leq 0.1$ within the limit of 1000 epochs. The results of the simulations presented in table (3), figure (5) and (6) where the vertical axis represent error and horizontal axis represent the number of epochs

Table 3: Results of simulations for the SPECT Heart Problem

Algorithms	min	max	mean	Tav	succ
FR	13	77	27.14	0.57716	100%
YH	15	40	24.12	0.5522	100%

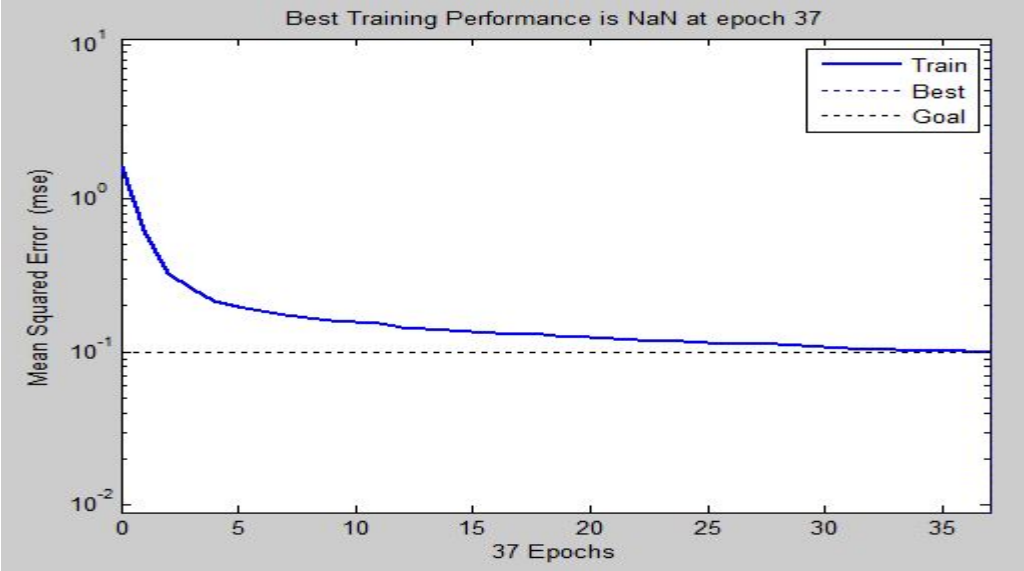

Figure 5: Mean Squared Error (MSE) for FR

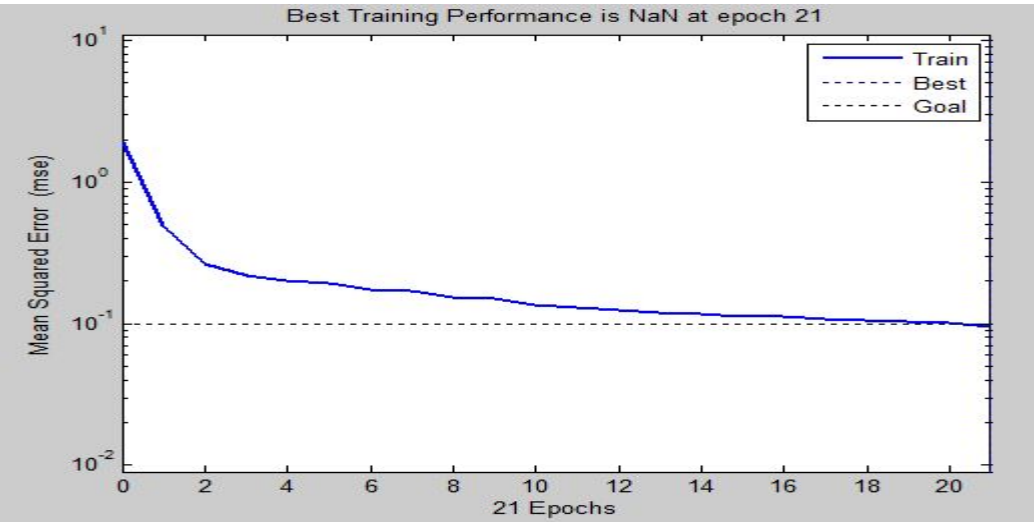

Figure 6: Mean Squared Error (MSE) for YH

4 Conclusions

The Fletcher-Reeves performs $g_{k+1}^T g_k \approx g_h^T g_h$ leads to d_{k+1}become linear combination with the d_k which is the main drawbacks for the Fletcher-Reeves method. Then this lead loss of descent property for Fletcher-Reeves method to overcome to this problem we developed the modified Fletcher-Reeves algorithm by adding $M\left|d_k^T\ g_{k+1}\right|$ to the denominator to a void this problem, where $M > 1$. Experimental results provide evidence that our proposed method is preferable and superior to the Fletcher-Reeves algorithm.

References

[1] C. Bishop and CM. Bishop, Neural networks for pattern recognition, Oxford University Press, 1995.

[2] J. A. Suykens and J. Vandewalle, Least squares support vector machine classifiers, Neural Process. Lett. 9 1999, 293-300.

[3] A. Hmich, A. Badri and A. Sahel, Automatic speaker identification by using the neural network, International Conference on IEEE Multimedia Computing and Systems (ICMCS), 2011.

[4] D. E. Rumelhart and J. L. McClell, Explorations in the Microstructure of Cognition, (Volume 1: Foundations). MIT press, 1988.

[5] I. Livieris and P. Pintelas, Performance evaluation of descent CG methods, for neural networks training, In Proceedings of "9th Hellenic European Research on Computer Mathematics and it is Applications Conference (HERCMA)", 2009.

[6] D. E. Rumelhart G. E. Hinton and R. J. Williams, Learning representations by back-propagating errors, Nature, 323 1986, 533.

[7] R. Battiti, First-and second-order methods for learning: between steepest descent and Newton's method, Neural Comput., 4 1992, 141-166.

[8] K. K. Abbo and H. H. Mohammed, Conjugate Gradient Algorithm Based on Aitken's Process for Training Neural Networks, AL-Rafidain Journal of Computer Sciences and Mathematics. 11 2014, 39-51.

[9] I. E. Livieris and P. Pintelas, A survey on algorithms for training artificial neural networks, University of Patras, Department of Mathematics, Educational Software Development Laboratory, University of Patras, GR-265 4 2008.

[10] I. E. Livieris and P. Pintelas, An advanced conjugate gradient training algorithm based on a modified secant equation, ISRN Artificial Intelligence 2011.

[11] I. E. Livieris, D. G. Sotiropoulos and P. Pintelas, On descent spectral CG algorithms for training recurrent neural networks, In Proceedings of "13th Panhellenic Conference on. IEEE", 2009.

[12] R. Fletcher and C. M. Reeves, Function minimization by conjugate gradients, The Computer Journal, 7 1964, 149-154.

[13] J. C. Gilbert and J. Nocedal, Global convergence properties of conjugate gradient methods for optimization, SIAM Journal on Optimization, 2 1992, 21-42.

[14] M. J. D. Powell, Restart procedures for the conjugate gradient method, Math. Programm., 12 1977, 241-254.

[15] D. Touati-Ahmed and C. Storey, Efficient hybrid conjugate gradient techniques, J. Optimiz. Theory App., 64 1990, 379-397.