Kirkuk University Journal /Scientific Studies (KUJSS)
Volume 10, Issue 3, September 2015 , p.p(230-241)
ISSN 1992 - 0849

New Scaled Conjugate Gradient Algorithm for Training Artificial Neural Networks Based on Pure Conjugacy Condition

[1]Khalil K. Abbo , [2]Hind H. Mohamed
[1,2]Department Of Math/College Of Computer Sci. And Math/University Of Mosul
[1]kh_196538@yahoo.com , [2]hinm@yahoo.com

Received date: 1 / 4 / 2015 **Accepted date: 8 / 3 / 2015**

ABSTRACT

Conjugate gradient methods constitute excellent neural network training methods characterized by their simplicity efficiency and their very low memory requirements. In this paper, we propose a new scaled conjugate gradient neural network training algorithm which guarantees descent property with standard Wolfe condition. Encouraging numerical experiments verify that the proposed algorithm provides fast and stable convergence.

Keywords: *Feed-forward Neural networks, training algorithms.*

خوارزمية جديدة متدرجة من نوع ذوات الانحدار المترافق لتدريب الشبكات العصبية الاصطناعية المستندة على شرط الترافق الصرف

خليل خضر عبو [1] , هند حسام الدين محمد[2]
[1,2]قسم الرياضيات / كلية علوم الحاسوب والرياضيات / جامعة الموصل
[1]kh_196538@yahoo.com , [2]hinm@yahoo.com

تاريخ قبول البحث: 8 /3/ 2015 **تاريخ استلام البحث: 1 / 4 / 2015**

الملخص

طرق التدرج المترافق تمثل طرق تدريب ممتازة للشبكات العصبية وتتميز ببساطتها، كفاءتها العددية وتطلبها ذاكرة منخفضة جدا. تم في هذا البحث اقتراح خوارزمية جديدة من نوع تدرج مترافق بمعلمة لتعليم الشبكة العصبية هذه الخوارزمية تحقق خاصية الانحدار باستخدام شرط ولف الاعتيادية. النتائج العددية لهذه الخوارزمية مشجعة من ناحية السرعة والاستقرارية مقارنة بالطرق الاخرى في هذا المجال.

كلمات دالة: شبكات عصبية ذات التغذية الامامية ، خوارزميات التعليم.

1.INTRODUCTION

Learning systems, such as multilayer feed-forward neural networks (FNN) are parallel computational models comprised of densely interconnected, adaptive processing units, characterized by an inherent propensity for learning from experience and also discovering new knowledge. Due to their excellent capability of self-learning and self-adapting, they have been successfully applied in many areas of artificial intelligence [1,2,3]and are often found to be more efficient and accurate than other classification techniques [4]. The operation of a FNN is usually based on the following equations:

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

where net_j^l is the sum of the weight inputs for the j-th node in the l -th layer (j=1,2,…, N_l), $w_{i,j}$ is the weights 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 belongs to the l -th layer, $f(\cdot)$ is the activation function and O_j^l is the output of the nod j at the output layer.

Recently many learning algorithms for feed-forward neural networks has been discovered [4,5,6]. Several of these algorithms are based on a known method in optimization theory known as the gradient descent algorithm. They usually have a poor convergence rate and depend on parameters which have to be specified by the user, since there is no theoretical basis for choosing them exists. The values of these parameters are often crucial for the success of the algorithm. For example the Standard Back Propagation(SBP) algorithm [7] which often behaves very badly on large-scale problems and which success depends of the user dependent parameters learning rate.

The problem of training a neural network is iteratively adjusting its weights, in order to minimize the difference between the actual output of the network and the desired output of the training set. Actually finding such minimum is equivalent to minimization of the error function which defined by:

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

The variables T_i and O_i are the desired (target) and the actual output of the i-th neuron, respectively. The index j denotes the particular learning pattern. The vector W is composed of all weights in the net[4].

From an optimization point of view learning in a neural network is equivalent to minimizing a global error function, which is a multivariate function that depends on the weights in the network. This perspective gives some advantages in the development of effective learning algorithms because the problem of minimizing a function is well known in other fields of science, such as conventional numerical analysis [8]. Since learning in realistic neural network applications often involves adjustment of several thousand weights only optimization methods that are applicable to large-scale problems, are relevant as alternative learning algorithms. The general opinion in the numerical analysis community is that only one class of optimization methods exists that are able to handle large-scale problems in an effective way. These methods are often referred to as the Conjugate Gradient(CG) Methods[8]. Several conjugate gradient algorithms have recently been introduced as learning algorithms in neural networks [5,9,10].

2.CONJUGATE GRADIENT METHDS

Conjugate gradient methods are probably the most famous iterative methods for efficiently training neural networks due to their simplicity, numerical efficiency and their very low memory requirements. These methods generate a sequence of weights $\{w_k\}$ using the iterative formula.

$$w_{k+1} = w_k + \alpha_k d_k \qquad (3)$$

where k is the current iteration usually called epoch, $w_1 \in R^n$ is a given initial point, $\alpha_k > 0$ is the learning rate and d_k is a descent search direction (by Descent, we mean $g_k^T d_k < 0 \ \forall k$) defined by

$$d_{k+1} = -g_{k+1} + \beta_k d_k \ , \ d_1 = -g_1 \qquad (4)$$

Conjugate gradient methods differ in their way of defining the multiplier β_k. The most famous approaches were proposed by Fletcher–Reeves (FR) and Polak–Ribere (PR) :

$$\beta^{FR} = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k} \ , \ \beta^{PR} = \frac{g_{k+1}^T y_k}{g_k^T g_k} \qquad (5)$$

The conjugate gradient methods using β^{FR} update were shown to be globally convergent [8]. However the corresponding methods using β^{PR} or β^{HS} update are generally more efficient ever without satisfying the global convergence property. In the convergence analysis and implementations of CG methods, one often requires the inexact line search such as the Wolfe line search. The standard Wolfe line search requires α_k satisfying:

$$\mathrm{E}\,(\mathrm{w}_{\mathrm{k}} + \alpha_k d_k) \le \mathrm{E}\,(\mathrm{w}_{\mathrm{k}}) + \rho \alpha_k g_k^T d_k \qquad (6)$$

$$\mathrm{g}(\mathrm{w}_{\mathrm{k}} + \alpha_K d_k)^{\mathrm{T}} d_k \ge \sigma \mathrm{g}_{\mathrm{k}}^{\mathrm{T}} \mathrm{d}_{\mathrm{k}} \qquad (7)$$

or strong Wolfe line search:

$$\mathrm{E}\,(\mathrm{w}_{\mathrm{k}} + \alpha_k d_k) \le \mathrm{E}\,(\mathrm{w}_{\mathrm{k}}) + \rho \alpha_k g_k^T d_k \qquad (8)$$

$$\left| g_{k+1}^T d_k \right| \ge \sigma g_{\mathrm{k}}^{\mathrm{T}} d_{\mathrm{k}} \qquad (9)$$

where $0 < \rho < \sigma < 1$. Moreover, an important issue of CG algorithms is that when the search direction (4) fails to be descent directions we restart the algorithm using the negative gradient direction to grantee convergence . A more sophisticated and popular restarting is the Powell restart [11].

$$\left| g_{k+1}^T g_k \right| \ge 0.2 \left\| g_{k+1} \right\|^2 \qquad (10)$$

where $\| \; \|$ denotes to the Euclidean norm. Other important issue for the CG methods is that the search directions generated from equation (4) are conjugate if the objective function is convex and line search is exact i.e:

$$d_i^T G d_j = 0 \,, \forall \; i \ne j \qquad (11)$$

where, G is the Hessian matrix for the objective function . the conjugacy condition given in (11) can be replaced [12] to the following equation:

$$d_{k+1}^T y_k = 0 \qquad (12)$$

which is called pure conjugacy. [13] show that if α_{k} is not exact the condition in (12) can be written as

$$d_{k+1}^T y_k = -t\, g_{k+1}^T s_k \,, \quad t > 0, \quad s_{\mathrm{k}} = w_{\mathrm{k+1}} - w_{\mathrm{k}} \qquad (13)$$

for general objective function with inexact line search.

Recently Abbo and Mohammed in [6] suggested a new CG algorithm β^{NA} for training the FFNN defined as:

$$\beta^{NA} = \frac{\gamma g_k^T g_k + g_{k+1}^T y_k}{y_k^T d_k}, \; 0 < \gamma << 1 \qquad (14)$$

3.SCALED CONJUGATE GRADIENT ALGORITHMS (SCG)

This type of algorithms assumes more general form of CG search direction. It generates a sequence w_k of approximations to minimum w^* of E, in which

$$w_{k+1} = w_k + \alpha_k d_k \qquad (15)$$

$$d_{k+1} = -\theta_{k+1} \; g_{k+1} + \beta_k \; d_k \qquad (16)$$

where, $g_k = \nabla E(w_k)$, α_k is selected to minimize $E(w)$ along the search direction d_k and β_k is a scalar. The iterative process is initialized with an initial point w_1 and $d_1 = -g_1$.

Observe that if $\theta_{k+1} = 1$, we get the classical CG algorithm according to the value of β_k. On the other hand, if $\beta_k = 0$, then we get another class of algorithms according to the selection of the parameter θ_{k+1}. There are two possibilities for θ_{k+1}: a positive scalar or positive definite matrix. If $\theta_{k+1} = 1$, then we have the steepest descent algorithm. If $\theta_{k+1} = H_{k+1}$, or an approximation of it, then we get Newton or quasi-Newton (QN) algorithms, respectively [14]. Therefore, we see that in the general case, when $\theta_{k+1} \neq 0$ is selected in a quasi-Newton manner, and $\beta_k \neq 0$, (16) represents a combination between the QN and CG methods.

Different scaled CG methods introduced [14], for example scaled Fletcher–Reeves (SFR) and scaled Polak–Ribere (SPR) :

$$\beta_k^{SFR} = \frac{\theta_{k+1} g_{k+1}{}^{\mathrm{T}} \; g_{k+1}}{\theta_k g_k^T \; g_k}, \quad \beta_k^{SFR} = \frac{\theta_{k+1} y_k{}^{\mathrm{T}} \; g_{k+1}}{\theta_k g_k^T \; g_k} \qquad (17)$$

$$\theta_{k+1} = \frac{s_k^T s_k}{s_k^T y_k} \qquad (18)$$

The main object of this work is to find a new and efficient scaled conjugate gradient method with search direction d_{k+1} having the simple form (16). For this purpose, we use the pure conjugacy condition (11) and β^{NA} .

3.1.New Scaled CG Method(Say N1SCG)

Abbo and Mohammed in [6] suggested a new CG algorithm β^{NA}based on the Aitken's process, in this section we try to generalize the method to more general form known as scaled conjugate gradient methods. Consider the search direction of the form :

$$d_{k+1} = -\theta_{k+1} g_{k+1} + \beta_K^{NA} d_k \qquad (19)$$

If we multiply both sides of the equation (19) by y_k we get :

$$d_{k+1}^T y_k = -\theta_{k+1} g_{k+1}^T y_k + \beta_K^{NA} d_k^T y_k \qquad (20)$$

By using the pure cojugacy condition (12) we get:

$$-\theta_{k+1} g_{k+1}^T y_k + \beta_K^{NA} d_k^T y_k = 0 \qquad (21)$$

then

$$\theta_{k+1} = \frac{\gamma g_k^T g_k + g_{k+1}^T y_k}{y_k^T g_{k+1}}$$

to avoid division to zero we can define θ_{k+1} as

$$\theta_{k+1} = \begin{cases} 1 + \dfrac{\gamma g_k^T g_k}{\left| y_k^T g_{k+1} \right|} & ; if \ y_k^T g_{k+1} \neq 0 \\ 1 & ; otherwise \end{cases} \qquad (22)$$

Then the search direction for the new scaled conjugate gradient (N1SCG) algorithm can be written as:

$$d_{k+1} = -\theta_{k+1} g_{k+1} + \beta_k^{NA} d_k \qquad (23)$$

We summarize our scaled conjugate gradient (N1SCG) algorithm as;

Algorithm (N1SCG)

Step1. Initialization: Select $w_1 \in R^n$, $0 < \gamma << 1$, gol= E_G , $\varepsilon > 0$ and

$K_{\max}$ (maximum number of epochs) and the parameters

$0<\rho\leq\sigma<1$. Compute $E(w_1)$ and $g_1=\nabla E(w_1)$. Consider $d_1=-\gamma\, g_1$ and set $\alpha_1=1$.

Step2. Test for continuation of iterations. IF $(E_k<E_G) or \|g_k\|<\varepsilon$,set $w^*=w_k$ and $E^*=E_k$, then stop. Else go to Step 3.

Step3. Line search. Compute α_k satisfying the Wolfe line search conditions (6) and (7) and update the variables $w_{k+1}=w_k+\alpha_k\, d_k$. Compute E_{k+1}, g_{k+1}, $s_k=w_{k+1}-w_k$ and $y_k=g_{k+1}-g_k$.

Step4. θ_k parameter computation. Compute θ_k from equation(22).

Step5. β_k^{NA} conjugate gradient parameter computation. Compute β_k^{NA} from equation(14)

Step6. Direction computation. Compute $d_{k+1}=-\theta_{k+1}\ g_{k+1}+\beta_k^{NA}\, d_k$.

Step7. $k=k+1$, go to step(2).

3.2. The Descent Property Of The Suggested Algorithm

In this section, we shall show our new conjugate gradient (N1SCG) algorithm satisfies the descent property with standard Wolfe conditions as stated in the following theorem:

Theorem(3.1)

Consider the N1SCG method where the learning rate α_k satisfies the standard Wolfe conditions equation (6) and (7) and if $\gamma\, g_k^T\, g_k\geq g_{k+1}^T\, y_k$ then

$$d_k^T\, g_k<0 \tag{24}$$

Proof :

For $k=1$ we have $d_1=-g_1$, then $d_k^T\, g_k=-\|g_k\|<0$. Now, from equations (14), (19) and (22) we have :

$$d_{k+1}=-(1+\frac{\gamma\, g_k^T\, g_k}{|y_k^T\, g_{k+1}|})g_{k+1}+\frac{\gamma\, g_k^T\, g_k+g_{k+1}^T\, y_k}{d_k^T\, y_k}d_k$$

notice that, by Wolfe condition (6) and (7), $y_k^T d_k > 0$ therefore:

$$y_k^T d_k = g_{k+1}^T d_k - g_k^T d_k \ge \sigma g_k^T d_k - g_k^T d_k = (\sigma - 1) g_k^T d_k$$

hence $\dfrac{1}{y_k^T d_k} \le \dfrac{1}{(\sigma - 1) g_k^T d_k}$ then

$$d_{k+1}^T g_{k+1} \le -(1 + \frac{\gamma g_k^T g_k}{\left| y_k^T g_{k+1} \right|}) g_{k+1}^T g_{k+1} + \frac{\gamma g_k^T g_k + g_{k+1}^T y_k}{(\sigma - 1) g_k^T d_k} d_k^T g_{k+1}$$

again by the second Wolfe condition (7) with $(\sigma - 1) = -(1 - \sigma)$ and $-g_{k+1}^T d_k \le -\sigma g_k^T d_k$ then,

$$d_{k+1}^T g_{k+1} \le -(1 + \frac{\gamma g_k^T g_k}{\left| y_k^T g_{k+1} \right|}) g_{k+1}^T g_{k+1} + \frac{\gamma g_k^T g_k + g_{k+1}^T y_k}{(1 - \sigma) g_k^T d_k} (-\sigma) g_k^T d_k$$

$$\therefore d_{k+1}^T g_{k+1} \le -(1 + \frac{\gamma g_k^T g_k}{\left| y_k^T g_{k+1} \right|}) g_{k+1}^T g_{k+1} - \frac{(\gamma g_k^T g_k + g_{k+1}^T y_k)\sigma}{(1 - \sigma)}$$

$$\le -(1 + \frac{\gamma g_k^T g_k}{\left| y_k^T g_{k+1} \right|}) g_{k+1}^T g_{k+1} - \frac{(\gamma g_k^T g_k + \left| g_{k+1}^T y_k \right|)\sigma}{(1 - \sigma)}$$

$$= -(\left| g_{k+1}^T y_k \right| + \gamma g_k^T g_k)\left[\frac{g_{k+1}^T g_{k+1}}{\left| g_{k+1}^T y_k \right|} + \frac{\sigma}{1 - \sigma} \right]$$

therefore, $d_{k+1}^T g_{k+1} < 0$■.

4.EXPERIMENTAL RESULTS

In this section, we will present experimental results in order to evaluate the performance of our proposed N1SCG in two problems the iris problem and continuous function approximation problem. The implementation code was written in Matlab 7.9 based on the SCG code of Birgin and Martınez [15]. All methods are implemented with the line search proposed in CONMIN [16] which employs various polynomial interpolation schemes and safeguards in satisfying the strong Wolfe line search conditions. The heuristic parameters were set as ρ= 10^{-4} and σ= 0.5 as in [10]. All networks have received the same sequence of input patterns and the initial weights were generated using the Nguyen-Widrow method [17].The results have been averaged over 500 simulations.

4.1.Training Performance

The cumulative total for a performance metric over all simulations does not seem to be too informative, since a small number of simulations can tend to dominate these results. For this reason, we use the performance profiles proposed by Dolan and More [18] to present perhaps the most complete information in terms of robustness, efficiency and solution quality. The performance profile plots the fraction P of simulations for which any given method is within a factor x of the best training method. The horizontal axis (x) of each plot shows the percentage of the simulations for which a method is the fastest (efficiency), while the vertical axis(p) gives the percentage of the simulations that the neural networks were successfully trained by each method (robustness). The reported performance profiles have been created using the Libopt environment [19] for measuring the efficiency and the robustness of our method in terms of computational time (CPU time) and epochs. The curves in the following figures have the following meaning:

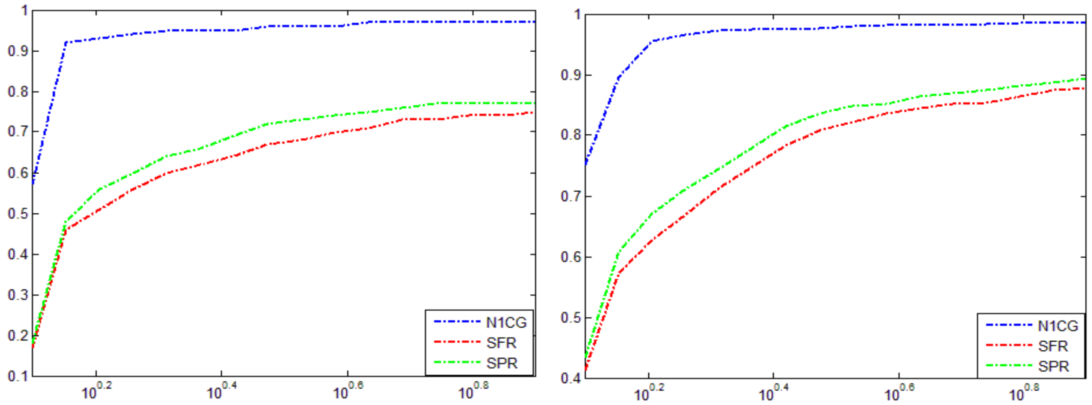
- “SPR” stands for the Scaled Polak-Ribiere CG method.
- “SFR” stands for the Scaled Fletcher-Reever CG method.
- “N1SCG” New Scaled proposed algorithm.

4.1.1. Iris Classification Problem

This benchmark is perhaps the most best known to be found in the pattern recognition literature [20]. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. The network architectures constitute of 1 hidden layer with 7 neurons and an output layer of 3 neurons. The training goal was set to $E_G \leq 0.01$ within the limit of 1000 epochs and all networks were tested using 10-fold cross-validation [10].

Figure (1) presents the performance profiles for the iris classification problem, regarding both performance metrics. N1SCG illustrates the best performance in terms of efficiency and robustness, significantly out-performing the scaled training methods SPR and SFR.

Furthermore, the performance profiles show that N1SCG is the only method reporting excellent (100%) probability of being the optimal training method.



(a): Performance based on CPU time **(b):** performance based on epoch

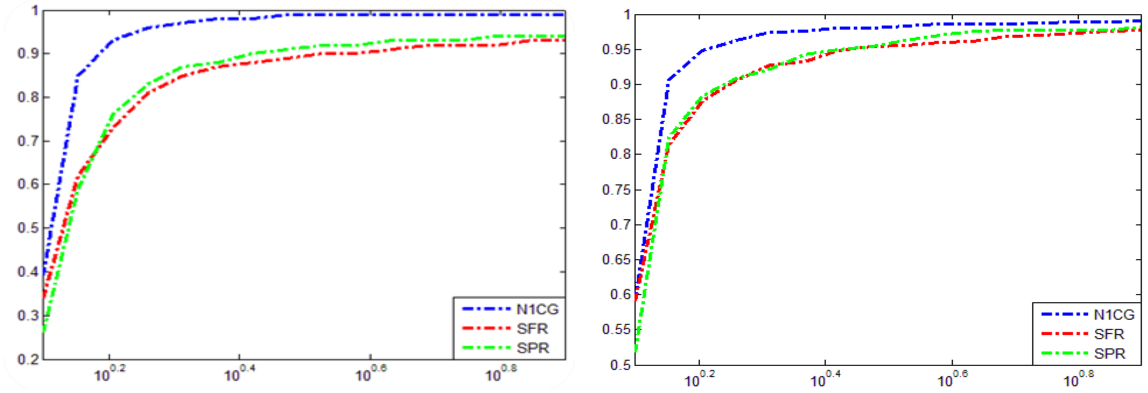
Figure (1): Log10 scaled performance profiles for the iris classification problem.

4.1.2.Continuous Function Approximation

The second test problem is the approximation of the continuous trigonometric function f(x)=sin(2πx)+0.1*rand([-1,1]) where x∈[-1,1].

The network architecture for this problem is 1-15-1 FNN, the network is trained until the sum of the squares of the errors becomes less than the error goal 0.001. The network is based on hidden neurons of logistic activations with biases and on a linear output neuron with bias.

Figure (2) shows the performance profiles for the continuous function approximation problem, investigating the efficiency and robustness of each training method. Clearly, our proposed method N1SCG significantly out-performs the scaled conjugate gradient methods SPR and SFR since the curves of the former lie above the curves of the latter, regarding both performance metrics. More analytically, the performance profiles show that the probability of N1SCG to successfully train a neural network within a factor 3.41 of the best solver is 94%, in contrast with SPR and SFR which have probability 84.3% and 85%, respectively.



(c): Performance based on CPU time **(d):** Performance based on epoch

Figure (2): Log10 scaled performance profiles for the function approximation problem.

5. CONCLUSIONS

It can be seen that if the scaling parameter θ contains two positive Terms a void the small multiplier for the gradient vector and hence Maintain the descent property and performance better than with scaling parameter with only one.

REFERENCES

[1] M. Bishop (1995).' ***Neural Networks for Pattern Recognition'***. Oxford.

[2] S. Haykin (1994).' ***Neural Networks: A comprehensive Foundation'.*** Macmillan College Publishing Company, New York.

[3] J. Hertz , A. krogh and R. Palmer (1991). ***'Introduction to the Theory of Neural Computation'*** Addison- Wesley, reading MA.

[4] I. Livieris and P. Pintelas (2011). ***'An advanced conjugate gradient training algorithm based on a modified secant equation'***. Technical Report No. TR11-03'. University of patras. Dep.of Math. GR-265 04 patras , Greece.

[5] R. Battiti. (1992). ***'1st and 2nd order method for learning between steepest descent and Newton method'***. Neural Comp. 4(2).

[6] K. Abbo and H. Mohammed (2014).***'Conjugate gradient algorithm based on Aitken's process for training neural networks'***.Raf. J. of Comp. and Math. Vol(11),No(1).

[7] D. Rumelhart, G. Hinton.and R. Wiliams (1986). ***'Learning representations by back-propagating errors'***, Nature, vol. 323.

[8] A. Andreas and S. Wu (2007). ***'Practical Optimization Algorithms and Engineering Applications***'. Springer Sci. and Business Media, LLC New York.

[9] K. Abbo and H. Mohammed (2013). ***'Improving the learning rate of the Back propagation by Aitken process'***. Iraqi J. Of Statistical Sci. 23.

[10] I. Livieris and P. Pintelas(2009).' ***Performance evaluation of descent CG methods for neural network training'***. In E.A. Lipitakis , editor,9th Hellenic European Research on Computer Mathematics & its Applications Conference (HERCMA'09).

[11] M. Powell(1977). ***'Restart procedure for the conjugate gradient Methods'***. Mathematics programming, Vol.(12).

[12] Y. Dai and L. Liao (2001).***'New conjugacy conditions and related non-linear CG methods.*** Appl. Math optim.(43). New York.

[13] G. Li , C. Tang and Z. Wei (2007). ***'New conjugacy condition and related new conjugate gradient methods for unconstrained optimization'***. J. of Computational and Applied Mathematics, 202.

[14] N.Andrei (2007).' ***Numerical comparison of conjugate gradient algorithms for unconstrained optimization'***. Studies in Informatics and Control, 16.

[15]G. Birgin and J. Mart´ınez (1999).' ***A spectral conjugate gradient method for unconstrained optimization'***. Applied Math. And Optim.43

[16] D. Shanno and K. Phua (1976).' ***Minimization of unconstrained multivariate functions'***. ACM Transaction son Mathematical Software, 2.

[17] D. Nguyen and B. Widrow (1990). ***'Improving the learning speed of 2-layer neural network by choosing initial values of adaptive Weights'***. Biological Cybernetics 59.

[18] E. Dolan and J. Mor´e (2002).' ***Benchmarking optimization software with performance profiles***'. Math. Programming 91.

[19] C. Gilbert and X. Jonsson.(2007). ***'LIBOPT-An environment for testing solvers on heterogeneous collections of problems – version 1'.*** CoRR, abs/cs/0703025.

[20] P.Murphy and D.Aha (1994).' ***UCI repository of machine learning databases. Irvine***, CA: University of California, Dep. Of Information and Computer Sci.

AUTHOR

Khalil K. Abbo: obtained his B.S.c and M.sc. in mathematics from Deportment of mathematics, College of science Mosul University. and Ph.D in numerical networking from Dept of Math, College of computer Sciences and mathematics. Dr. Abbo interest research area is numerical Optimization. Now Dr. Abbo is a senior Lecturer in same Dept. As well as he is the Dear of College of Information Technology in Telafar University. Dr. Abbo has published about 28 and scientific papers in word wide Journals and Supervised 2 M.Sc. Students and Ph.D. student. Currently Dr.Abbo supervising 3 M.Sc. students and 2 Ph.D. students and involved in several Researches.