Indian Journal of Computer Graphics and Multimedia (IJCGM)
ISSN: 2582-8592 (Online), Volume-3 Issue-2 August 2023

Effectiveness of MATLAB and Neural Networks for Solving Nonlinear Equations by Repetitive Methods

Mona A. Elzuway, Hend M. Farkash, Amani M. Shatshat

***Abstract**: Finding solutions to nonlinear equations is not only a matter for mathematicians but is essential in many branches such as physics, statistics, and others. However, some of the nonlinear equations in numerical analysis require a lot of complex calculations to achieve convergence. This leads to many arithmetic errors and is consumed a great effort to solve them. Hence, researchers in numerical analysis use computer programs to find approximate solutions. This study used Matlab and Artificial Neural Networks and applied two different numerical analysis methods. The results from training artificial neural networks by utilizing the Backpropagation algorithm and MATLAB have been compared. The importance of this study lies in shedding light on the capabilities of Matlab and its strength in the field of methods for solving mathematical series, and helps students in mathematics in solving complex equations faster and more accurately, also studying the utilization of Artificial Neural Network algorithms in solving these methods, and clarifying the difference between them and programming Ordinary Matlab and comparing them with ordinary mathematical methods. The findings revealed that Traditional methods need more effort. MATLAB helps. On the other hand, solving numerical analysis problems is easier, faster, more accurate, and more effective. Furthermore, in the case of the Matlab application, the Newton method gave faster and less in the number of steps. Additionally, in training, the neural network based on the Newton method gave results faster depending on the Bisection method.*

***Keywords**: Nonlinear Equations, Repetitive Methods, Matlab, Artificial Neural Networks, Newton-raphson, Bisection.*

I. INTRODUCTION

Solving nonlinear equations in numerical analysis, which the calculator solves, takes a long time and long term in the solution of complex calculations that lead to many computational errors, no matter how much he is doing the calculations and is consuming a huge effort in solving them. However, the computer gives a very accurate result and provides the optimal solution.

Manuscript received on 01 July 2023 | Revised Manuscript received on 14 August 2023 | Manuscript Accepted on 15 August 2023 | Manuscript published on 30 August 2023.
*Correspondence Author(s)

Mona A. Elzuway, The College of Electrical and Electronic Technology, Benghazi, Libya. E-mail: mona.elzuway@ceet.edu.ly, ORCID ID: 0009-0001-7710-6124

Hend M. Farkash, The College of Electrical and Electronic Technology, Benghazi, Libya. E-mail: Hend.mf_ceet@ceet.edu.ly, ORCID ID: 0009-0009-5672-9873

Amani M. Shatshat, The College of Electrical and Electronic Technology, Benghazi, Libya. E-mail: Amani.milad@ceet.edu.ly, ORCID ID: 0009-0002-2728-6996

Therefore utilizing programs such as Matlab solves the problem of time effort for the user by knowing specific codes to deal with the program or the so-called code [1]. Numerical analysis in practical life has numerous applications. The fame of numerical analysis and its enormous growth is currently one of the most important pieces of evidence that its applications continue to be the source of mathematical innovation. Thus, the numerical analysis is considered the numerical destination for the wide analysis requirements applications and the data field. The conventional thinking of some applications by known numerical methods does not give an exact answer to computer programs, and numerical analysis involves studying the data given and calculating the required practical results[1].In some Literature Reviews, entitled comparative of Bisection, Newton-Raphson and Secant Methods of Root- Finding Problems, The authors compare the rate of performance, viz-aviz, the rate of convergence of Bisection method, Newton-Raphson method and the Secant method of root-finding. The software Mathematica 9.0 was used to find the root of the function. It was concluded that the Secant method is the most effective scheme of the three methods considered. [2] Moreover, In another previous study, the comparative study of bisection and newton-raphson methods of root-finding problems has been proposed. The researcher has presented two numerical techniques of root-finding problems of nonlinear equations with the assumption that a solution exists, the rate of convergence of the Bisection method and the Newton-raphson method of root-finding has also been discussed. The software package MATLAB 7.6 was used to find the root of the function. The author concludes that Newton's method is formally the most effective of the methods compared with the Bisection method in terms of an order of convergence. [3] The importance of the study is to help solve the problems of numerical analysis and reduce the error rate, as this is the main objective of numerical analysis. Furthermore, it also shows the possibility of using the mechanism in solving equations and mathematical problems with the least effort and the shortest period of the time, such as the effectiveness of Matlab and Artificial Neural Networks. An expression of the form f(x) = $a_0 + a_1 x + a_2 x^2$ + …… + $a_n x^n$, where a_0 are constants($a_n \neq 0$) and n is a positive integer. if f(x) contains some other functions as trigonometric, logarithmic, exponential etc. then, f(x) = 0 is called a transcendental equation. The value of _ of x which satisfies f(x) = 0 is called a root of f(x) = 0, and a process to find out a root is known as root finding [4].

They are equations of one or more quadratic or exponential, logarithmic or trigonometric functions, where two different equations were taken. The application of these equations on two different methods of numerical analysis, namely, the method of BISECTION and NEWTON - RAPHSON method, in addition to the training of neural networks to solve equations of this type and we used in the FORWARD FEEDING NETWORK applied to the back propagation algorithm. Known methods that use direct formulas that lead to the solution of the equation an exact solution are methods that apply only to a minimal number of equations. We will use some numerical methods called iterative methods and usualy lead to finding the roots of the real (called the root of the equation, (f (x) = 0 F (a) = 0).
We have an initial approximate value of x_0 and then repeat the formula several times to get values closer to the exact root .

II. MATERIALS AND METHODS

This study uses two iterative methods in numerical analysis on two different models of the nonlinear equations:

1. $f(x) = 3(x+1)(x-\frac{1}{2})(x-1)$

a=[-2,1.5] b=[-1.25,2.5]

2. $f(x) = x\cos x - 2x^2 + 3x - 1$

a=[0.2,0.3] b=[1.2,1.3]

There are several iterative methods used in numerical analysis to solve nonlinear equations, The following were used in this study:

1. Bisection Method

This method is used for solving the nonlinear equation $f(x) = 0$, where f is a continuous function. Two initial value is required to start the procedure. This method is based on the Intermediate value theorem [4].

Let a function $f(x) = 0$, continuous on an interval $[a,b]$, such that $f(a)\,f(b) < 0$, then, the function $f(x) = 0$ has at least a root or zero in the interval $[a,b]$. The method calls for a repeated halving of subintervals of $[a,b]$ containing the root. Thus, the root always converges, though very slow in converging [5].

Algorithm:
Input:
i. *f(x)* is the given function
ii. *a; b* the two numbers such that *f*(a)*f*(b) <0
Output:
An approximatin of the root of *f(x)* = 0 in [*a; b*], for k = 0; 1; 2; 3…….. do until satisfied.
i. $c_k = 1/2(a_k+b_k)$
ii. Test if c_k is target root. if so, stop.
iii. If c_k is not the target root, test if $f(c_k)f(a_k) < 0$. if so, set $b_{k+1} = c_k$ and $a_{k+1} = c_k$.
Otherwise ck= $b_k + 1 = b_k$
End

2. Stopping Criteria for Bisection Method:

let E be the expected error, that is, we would like to obtain the root with an error of at most of e. Then, accept x = ck as root of f(x) = 0. If any of the following criteria are satisfied.[5]

i. |f(ck)| ≤ E (i.e. the functional value is less than or equal to the expected error).
ii. $\frac{|c_{k-1} - c_k|}{|c_k|} \leq$ E (ie the relative change is less than or equal to the expected error).
iii. $\frac{b-a}{2^k} \leq$ E (i.e, the interval length after k iterations is less than or equal to the expected error).
iv. The number of iterations k is greater than or equal to a predetermined number, say N.

The function has been implemented using the bisector method with MATLAB 2017 and ANN software.

3. Newton- Raphson Method

The Newton-Raphson method finds the slope of the function at the current point and uses the zero of the tangent line as the next reference point. The process is repeated until the root is found [3]. The Newton method is the most common technique for solving nonlinear equations Using fixed repetition:

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

x_o –the initial approximation near solution, $f'(x)$: Is a differential value for the function $f(x)$ at x_o,

Algorithm
Input:
i. *f(x)* is the given function
ii. *a; b* the two numbers such that *f*(a) *f*(b) <0
Output:
An approximatin of the root of *f(x)* = 0 in [*a; b*], for k = 0; 1; 2; 3……. do until satisfied.
i. $c_k = 1/2(a_k+b_k)$
ii. Test if c_k is target root. if so, stop.
iii. If c_k is not the target root, test if $f(c_k)f(a_k) < 0$. if so, set $b_k + 1 = c_k$ and $a_k + 1 = c_k$.
Otherwise $c_k = b_k + 1 = b_k$
End

If none of the above criteria are met within a predetermined, for example, N, the method fails when a certain number of repetitions. Then, you can try the technique again using a different xo. Finally, you can find an appropriate value for xo with a graph of f(*x*) using graph Software. Unfortunately, no specific method to choose a correct starting point (x_o) ensures a root convergence. The function has been implemented using the Newton-Raphson method with Matlab 2017 [10] and ANN(Artificial Neural Network).

III. ARTIFICIAL NEURAL NETWORK

The training procedure is classified into two types called: Supervised Training and Unsupervised Training. In the first one, a teacher or an instructor is an essential part of the training system to tell the structure what is wrong and what is right. This study used an algorithm working with this approach. This other type is Unsupervised Learning or learning without a "teacher". The network does not require an output for every input pattern in the training process.

Retrieval Number: 100.1/ijcgm.H96830712823
DOI: 10.54105/ijcgm.H9683.083223
Journal Website: www.ijcgm.latticescipub.com

Published By:
Lattice Science Publication (LSP)

Hybrid Learning Can be classified as a third type of training, which is a combination of the two mentioned [9][7][8]. A feed-forward neural network is trained with pairs of input-output examples in supervised learning. The most frequently used for supervised learning in forwarding feed networks is the backpropagation learning algorithm extracted from (Rumelhart, Aleksander) [6], used in this paper. Classification in a neural network: "Is a part of pattern recognition which is an assignment of the input data to one of a finite number of categories" [7][8], neural networks can be used to solve classification problems in many applications, and the most frequently used model is Multi-Layer Feed-Forward NNs.

1. Backpropagation algorithm

In supervised learning for feedforward networks, the most frequently used method is Error Back Propagation Training Algorithm (EBPT). Which is used to train a neural network to perform some tasks, it must adjust the weights of each unit so that the error between the desired output and the actual output is reduced. This process requires that the neural network compute the weights' error derivative. In other words, it must calculate how the error changes as each weight is increased or decreased slightly; it is shown in Figure (1) [7][8].

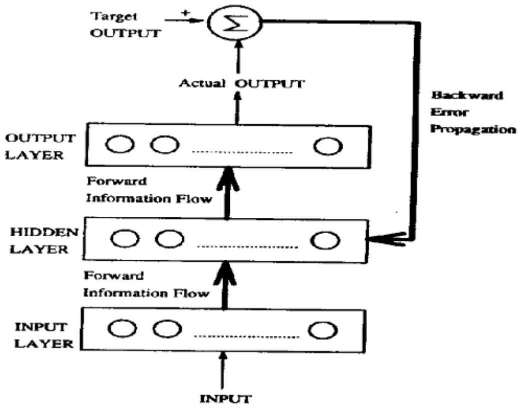

Fig. 1. Signal flow diagram for backpropagation algorithm

After developing the electronic exam platform, ten academicians were included in this study for their views on this platform. However, few studies were concerned to provide the foundations for rethinking and adapting traditional educational exams development models according to the Backpropagation algorithm procedure passed in successive steps:

1. Weight initialization:
   Set all weights to small random numbers.
   $$W_1, W_2 \ldots\ldots. W_n$$
   Since n represents the number of layers as i=1, 2, ……………n
2. Choose an input pattern (x)
3. Calculation of output of layers:

For last layer calculation output of activation function by a linear function, which determines by:

$$yc = f(w.x)$$

Since x is the output of previous layer.

But the output of hidden layers is determined by a sigmoid function:

$$y_i = \frac{1}{1+\exp(-wx)}$$

Since x is the output of the previous layer.

4. Computed errors:
   First, calculate cycle error:
   $$E = \frac{1}{2}\sum(yd - yc)^2$$
   Since yd is desired output, yc is computed output
5. Calculated δ for the output layer is:
   $$\delta_j^t = g'(h_j^t)[yd - yc]$$
   Where h represents the net input to the i^th unit in the i^th layer and g' is the derivative of the activation function.
6. Compute the deltas for the preceding layers by propagating the errors backward:
   $$\delta_j^t = g'(h_j^t)\sum_j w_j^{l-1}\delta_j^{l-1}$$
   For l=1,…………(l-1)
7. Adjust weights of each layer by:
   $$w_{ji}(t+1) = w_{ji}(t) + \Delta w_{ji}$$

Where t is iteration and Δw_{ji} is the weight adjustment, which is computed by:

$$\Delta w_{ji} = \eta \delta_j^t y_j^{l-t}$$

Since η is a learning rate where 0<η<1, and δ is a calculated error, sometimes to enhance the training process by adding moment term:

$$w_{ji}(t+1) = w_{ji}(t) + \Delta w_{ji} + \alpha\left[w_{ji}(t) - w_{ji}(t-1)\right]$$

Where 0< α <1

8. Repeat steps 2 through 5 until the error is acceptably low or a maximum number of iterations is reached.

2. Classifier using ANN

The ability of ANN to classify data is well known. However, it needs a proper design of ANN structure with enough training examples, The most used structure type is a multi-layer feed-forward neural network with backpropagation training algorithm. This can be represented graphically as:

An expert system contains.

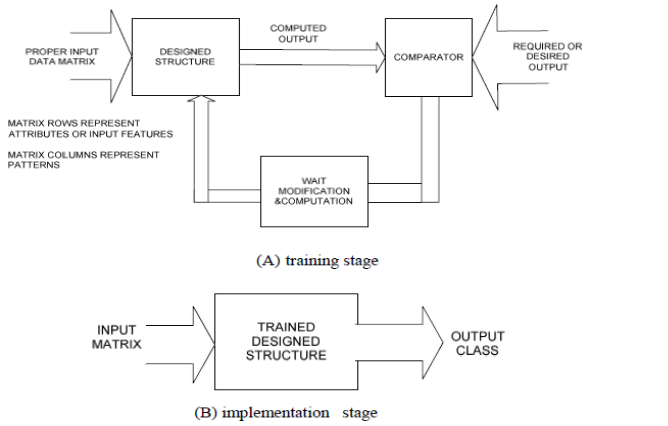

Fig. 2. Training &implementation phases of ANN

Retrieval Number: 100.1/ijcgm.H96830712823
DOI: 10.54105/ijcgm.H9683.083223
Journal Website: www.ijcgm.latticescipub.com

Published By:
Lattice Science Publication (LSP)

Figure (2) shows the general structure phases of ANN at the training stage and implementation stages.

IV. RESULT AND DISCUSSION

The technical specifications of the system were determined in work was done on the two equations presented and the application of the method of Bisection and Newton- Raphson and training of the neural network based on the results in the specified periods and record the following results:

1. $f(x) = x\cos x - 2x^2 + 3x - 1$

a. Newton- Raphson Method [0.2,0.3]

n	x	fx
0	0.2100	-0.2528
1.0000	0.2917	-0.0157
2.0000	0.2975	-0.0001

root =

0.2975

Fig. 3. Results of Newton- Raphson Method [0.2,0.3]

b. Newton- Raphson Method[1.2,1.3]

n	x	fx
0	1.2100	0.1290
1.0000	1.2592	-0.0076
2.0000	1.2566	-0.0000

root =

1.2566

Fig. 4. Results of Newton- Raphson Method[1.2,1.3]

c. Bisection Method [0.2,0.3]

a	b	c	f(c)	error_bound
0.2000	0.3000	0.2500	-0.1328	0.0500
0.2500	0.3000	0.2750	-0.0616	0.0250
0.2750	0.3000	0.2875	-0.0271	0.0125
0.2875	0.3000	0.2937	-0.0102	0.0062
0.2937	0.3000	0.2969	-0.0018	0.0031
0.2969	0.3000	0.2984	0.0024	0.0016
0.2969	0.2984	0.2977	0.0003	0.0008
0.2969	0.2977	0.2973	-0.0007	0.0004
0.2973	0.2977	0.2975	-0.0002	0.0002
0.2975	0.2977	0.2976	0.0001	0.0001

root =

0.2976

Fig. 5. Results of Bisection Method [0.2,0.3]

d. Bisection Method [1.2,1.3]

a	b	c	f(c)	error_bound
1.2000	1.3000	1.2500	0.0192	0.0500
1.2500	1.3000	1.2750	-0.0546	0.0250
1.2500	1.2750	1.2625	-0.0172	0.0125
1.2500	1.2625	1.2563	0.0011	0.0063
1.2563	1.2625	1.2594	-0.0080	0.0031
1.2563	1.2594	1.2578	-0.0035	0.0016
1.2563	1.2578	1.2570	-0.0012	0.0008
1.2563	1.2570	1.2566	-0.0001	0.0004
1.2563	1.2566	1.2564	0.0005	0.0002
1.2564	1.2566	1.2565	0.0002	0.0001

root =

1.2565

Fig.6. Results of Bisection Method [1.2,1.3]

Artificial Neural Network:

a. Based on Bisection Method

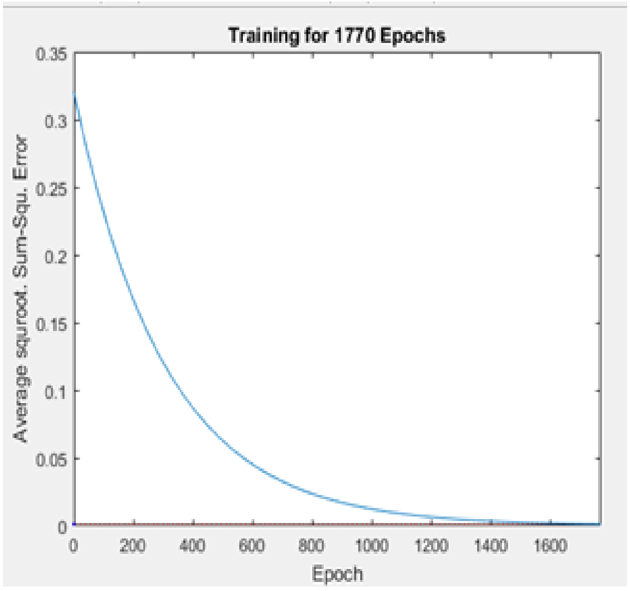

Fig.7. Results of ANN Based on Bisection Method

Table 1. Results of ANN Based on Bisection Method

Root	Iteration	Error	X_0
0.2979877	1844	0.0010008	0.21
1.2564296	1844	0.0010008	1.21

Retrieval Number: 100.1/ijcgm.H96830712823
DOI: 10.54105/ijcgm.H9683.083223
Journal Website: www.ijcgm.latticescipub.com

Published By:
Lattice Science Publication (LSP)

b. Based on Newton- Raphson Method

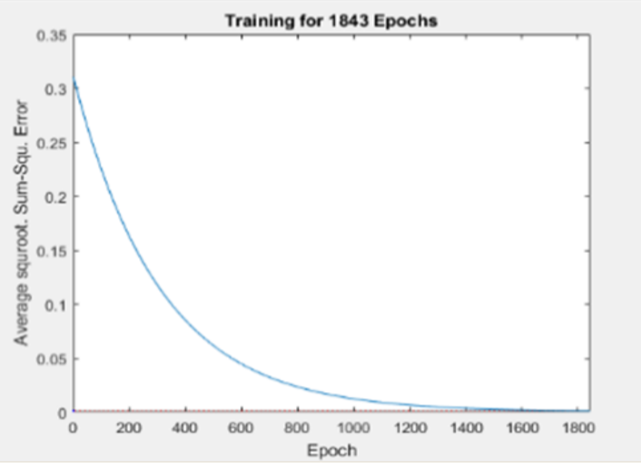

Fig.8. Results of ANN Based on Newton- Raphson Method

Table 2. Results of ANN Based on Newton- Raphson Method

Root	Iteration	Error	X_0
0.2979877	1844	0.0010008	0.21
1.2564296	1844	0.0010008	1.21

2. $f(x) = 3(x+1)(x-\frac{1}{2})(x-1)$

a. Newton- Raphson Method [-2,1.5]

```
        n          x         fx
-------------------------------------
        0     1.2000     0.9240

   1.0000     1.0547     0.1871

   2.0000     1.0061     0.0186

   3.0000     1.0001     0.0003

   4.0000     1.0000     0.0000


root =

   1.0000
```

Fig.9. Results of Newton- Raphson Method[-2,1.5]

b. Newton- Raphson Method [-1.25,2.5]

```
        n          x         fx
-------------------------------------
        0     0.2000     0.8640

   1.0000     0.4667     0.0782

   2.0000     0.4987     0.0029

   3.0000     0.5000     0.0000


root =

   0.5000

>>
```

Fig.10. Results of Newton- Raphson Method$[-1.25,2.5]$

c. Bisection Method [-2,1.5]

```
      a          b          c        f(c)    error_bound

  -2.0000     1.5000    -0.2500     2.1094     1.7500

  -2.0000    -0.2500    -1.1250    -1.2949     0.8750

  -1.1250    -0.2500    -0.6875     1.8787     0.4375

  -1.1250    -0.6875    -0.9063     0.7539     0.2188

  -1.1250    -0.9063    -1.0156    -0.1432     0.1094

  -1.0156    -0.9063    -0.9609     0.3357     0.0547

  -1.0156    -0.9609    -0.9883     0.1040     0.0273

  -1.0156    -0.9883    -1.0020    -0.0176     0.0137

  -1.0020    -0.9883    -0.9951     0.0437     0.0068

  -1.0020    -0.9951    -0.9985     0.0132     0.0034
```

Fig.11. Results of Bisection Method [-2,1.5]

d. Bisection Method [-1.25,2.5]

```
    0.9766    1.0938    1.0352    0.1149    0.0586
    0.9766    1.0352    1.0059    0.0178    0.0293
    0.9766    1.0059    0.9912   -0.0258    0.0146
    0.9912    1.0059    0.9985   -0.0044    0.0073
    0.9985    1.0059    1.0022    0.0066    0.0037
    0.9985    1.0022    1.0004    0.0011    0.0018
    0.9985    1.0004    0.9995   -0.0016    0.0009
    0.9995    1.0004    0.9999   -0.0003    0.0005
    0.9999    1.0004    1.0001    0.0004    0.0002
    0.9999    1.0001    1.0000    0.0001    0.0001
    0.9999    1.0000    1.0000   -0.0001    0.0001

root =

    1.0000

>>
```

Fig.12. Results of bisection method [-1.25,2.5]

Artificial Neural Network

a. Based on Newton- Raphson Method

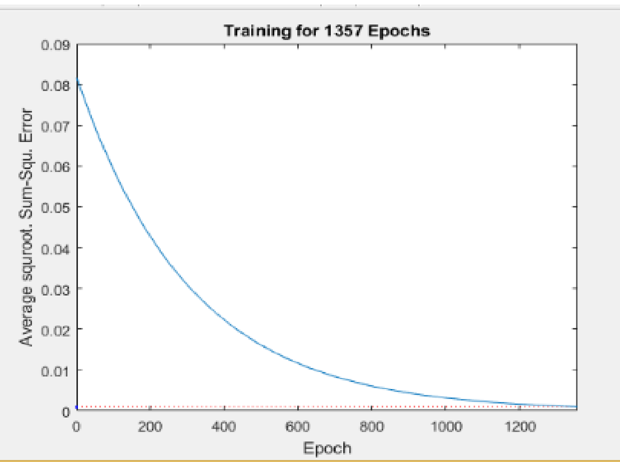

Fig.13. Results of ANN Based on Newton- Raphson Method

Table 3. Results of ANN Based on Newton- Raphson Method

Root	Iteration	Error(sser)	X_0
1	1357	0.0010022	1.2
0.5031653	1357	0.0010022	0.2

b. Based on Bisection Method

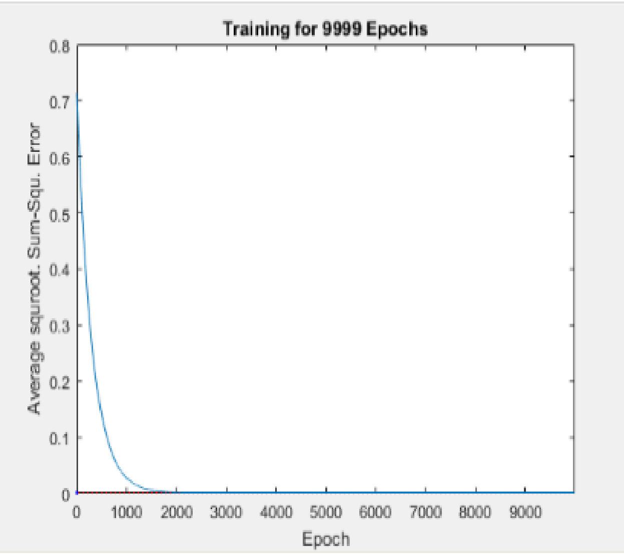

Fig14. Results of ann based on bisection method

Table 4. Results of ANN Based on Bisection Method

Root	Iteration	Error(sser)	X_0,X_1
-1.0028672	100000	0.00103	[-2,1.5]
1.0004909	100000	0.00103	[-1.25,2.5]

Looking at table 5. and after studying the traditional methods of solving problems in numerical analysis and comparing them with MATLAB and neural networks in solving problems of the same type, the following results were reached:

1. Traditional methods need more effort.
2. MATLAB helps solve numerical analysis problems in an easier, faster, more accurate and more effective way.
3. In the case of the use of Matlab application Newton method gave results faster and less in the number of steps.
4. In the case of training, the neural network based on the Newton method gave results faster depending on the Bisection method.This study laid the basis for applying MATLAB and neural networks to solve numerical analysis problems. Therefore, in the future, it is possible to develop this study using Matlab and artificial neural networks on different mathematical equations. In addition to generalizing it to other methods of analysis and comparing results, integrated expert systems can be built-in numerical analysis to solve this complex type of problem.

Table 5. comparing solving problems in numerical analysis methods with MATLAB and neural networks

Function (1) xcos(x)-2x2+3x-1		Function (2) 3 (x+1)(x-1\2)(x-1)	
Method	**Root**	**Method**	**Root**
Newton [0.2,0.3]	0.2975	Newton [-2,1.5]	1
Newton [1.2,1.3]	1.2566	Newton [-1.25,2.5]	0.5
Bisection[0.2,0.3]	0.2976	Bisection[-2,1.5]	-1
Bisection[1.2,1.3]	1.2565	Bisection[-1.25,2.5]	1
ANN-Newton[0.2,0.3]	0.2691	ANN-Newton [-2,1.5]	1
Ann=Newton[1.2,1.3]	1.2625	ANN-Newton[-1.25,2.5]	0.5
Ann-Bisection[0.2,0.3]	0.2979	ANN-Bisection [-2,1.5]	-1
Ann-Bisection[1.2,1.3]	1.2564	ANN-Bisection[-1.25,2.5]	1

V. CONCLUSIONS

This study sheds light on the capabilities of Matlab and its strength in the field of methods for solving mathematical series. Thus the effectiveness of Matlab and neural networks for solving nonlinear equations by utilizing the repetitive techniques of numerical analysis has been proposed. Additionally, the results obtained from applying the method of training artificial neural networks using the backpropagation algorithm and from the application of Matlab have been compared. Hence, the proposed method helps students in mathematics solve complex methods faster and more accurately, studying the utilization of Artificial Neural Network algorithms in solving these methods, and clarifying the difference between them and programming Ordinary Matlab and comparing them with ordinary mathematical methods. This study laid the basis for applying MATLAB and neural networks to solve numerical analysis problems. Therefore, it is possible to develop this study using MATLAB and artificial neural networks on different mathematical equations in the future. In addition to generalizing it to other methods of analysis and comparing results, integrated expert systems can be built-in numerical analysis to solve this complex type of problem.

ACKNOWLEDGMENT

The authors are grateful to the Research Management Center College of Electrical & Electronics Technology – Benghazi. Libya to support this research.

DECLARATION

Funding/ Grants/ Financial Support	No, I did not receive
Conflicts of Interest/ Competing Interests	No conflicts of interest to the best of our knowledge.
Ethical Approval and Consent to Participate	No, this article does not require ethical approval and consent to participate with evidence.
Availability of Data and Material/ Data Access Statement	Not relevant.
Authors Contributions	All authors having equal contribution for this article.

Retrieval Number: 100.1/ijcgm.H96830712823
DOI: 10.54105/ijcgm.H9683.083223
Journal Website: www.ijcgm.latticescipub.com

Published By:
Lattice Science Publication (LSP)

REFERENCES

1. Amani Ahmed, Thuwaiba Abdullah, Rafidah Awad, Mawaddah Al-Shibli, (2014) “The Effectiveness of Using Matlab in Solving Numerical Analysis Problems by Applying to (Newton Raphson and Gauss Methods)”, Graduation Project, Sudan University of Science and Technology.
2. Ehiwario, J.C., Aghamie, S.O. (2014),” Comparative Study of Bisection, Newton-Raphson and Secant Methods of Root- Finding Problems”, 2250-3021, ISSN (p): 2278-8719
3. ABDULAZIZ G. AHMAD (2015), “Comparative Study of Bisection and Newton-Rhapson Methods of Root-Finding Problems”, International Journal of Mathematics Trends and Technology-Volume 19 Number 2 Mar 2015. https://doi.org/10.14445/22315373/IJMTT-V19P516
4. R.L.Burden, J.D. Faires ,(2011), Numerical Analysis 9th Edition, Brook/Cole, a part of Cengage Learning.
5. J.C. Ehiwario, (2013), Lecture Notes on MTH213: Introduction to Numerical Analysis, College of Education Agbor. An Affiliate of Delta State University, Abraka.
6. M.Sordo,F.,(2002), Openclinical, “Introduction to Neural Networks in Healthcare.OpenClinical.:Knowledge Management for Medical Care,” Open Clinical.
7. L.Fu,(1994), NeuralNetworks in Computer Intelligence, McGraw-Hill Companies, International Editions.
8. A.K. Jain, J. Mao, and K., (1996), Mohiuddin, “Artificial Neural Networks: A Tutorial,” IEEE COMPUTER, vol. 29, pp. 31--44. https://doi.org/10.1109/2.485891
9. M. Negnevitsky, (2002), Artificial Intelligence: A Guide to Intelligent Systems, Addison Wesley.
10. D.,T. Allen,(2009),"Learning MATLAB", SIAM, ISBN 978-0-898716-83-2.

AUTHORS PROFILE

Mona A. El.zuway Master degree in Computer Sciences, Specialization in Artificial intelligence. Obtained Bachelor's degree in Computer Science, University of Benghazi, 2003. Lecturer in The College OF Electrical & Electronic Technology, Benghazi, Libya. and a collaborator in benghazi university and several private universities. Area of research interests: Computer Sciences, Artificial intelligence, Internet of things, About 17 years` experience of teaching, Author of several publications in In local and international journals. participation in local and international conferences. Google scholar ID/ link: AGnIfIAAAAJ, a member of the IEEE organization Number: 98806779.

Hend M.Farkash Assistant Professor ,Lecturer in The College OF Electrical & Electronic Technology, Benghazi, Libya, and a collaborator in several private universities. Education and Qualification: have Master degree in Computer Sciences. Specialization: Artificial Intelligence, Graduation Year :2010, University: faculty of information technology - Benghazi University, and have B.Sc in computer science. Graduation Year :2003 University: faculty of computer science - Benghazi University, Conferences and Seminars: have several researches in the field of intelligence, computer science and artificial intelligence, and distinguished participation in local and international conferences, In addition, and a member of the IEEE organization.

Amani M.Shatshat Master degree in Mathematics. from Benghazi University in 2001. Lecturer in The College OF Electrical & Electronic Technology, Benghazi, Libya. and a collaborator in several private universities. Author of several publications, Writing a book entitled "General Mathematics" published by the Mediterranean International University.