Optimization of the Neural Networks Parameters*

Dr.Mohammad Dweib**

Dr. Yousef Abuzir***

*Received: 25/7/2017, Accepted: 5/11/2017.
** Assistant Professor/ Al-Quds Open University/Palestine

***Associate Professor/ Al-Quds Open University/Palestine

Abstract:

Neural Networks (NN) is an effective approach used in many fields such as medicine, industry, security, stock market prediction, character recognition, image processing and many other fields. The main problem in the use of Artificial Neural Networks (ANN) is the control of parameters, since there is no explicit and specific way to determine the optimal values of Neural Network parameters. The aim of this study is to choose the best parameters that can be used to model and optimize the number of hidden layers, number of neurons contained in there, momentum, delta rule, transition functions and multidimensional network.

Keywords: ANN, optimization, BP neural network, hidden layers, momentum, delta rule, transition functions, and multidimensional network.

ملخص بالعربية:

الشبكة العصبية (NN) هو نهج فعال يستخدم في العديد من المجالات مثل الطب والصناعة والأمن والتنبؤ لسوق الأسهم، والتعرف على الحروف، ومعالجة الصور والعديد من المجالات الأخرى. المشكلة الرئيسية في استخدام الشبكة العصبية الاصطناعية (ANN) هي السيطرة على المعلمات، حيث لا توجد طريقة واضحة ومحددة لتحديد المعلمات المثلى من معلمات الشبكة العصبية. والهدف من هذه الدراسة هو اختيار أفضل المعلمات التي يمكن استخدامها لنمذجة وتحسين عدد الطبقات المخفية، وعدد من الخلايا العصبية الموجودة هناك، والزخم، قاعدة دلتا، وظائف الانتقال وشبكة متعددة الأبعاد.

الكلمات المفتاحية: ANN، النقطة الأمثل، شبكة العصبية بب، طبقات مخفية، والزخم، قاعدة دلتا، وظائف الانتقال، وشبكة متعددة الأبعاد.

INTRODUCTION

Preparation of solutions that generate Artificial Neural Networks (ANN) from a specific problem depends on selecting the appropriate value weights for the different inputs of all neurons. However, they do not take it on the way to assign these values by supervised learning, but automatically in the process called, learning network. The Backpropagation Artificial Neural Networks (BP ANN) can approximate any functions at arbitrary precision, can learn a lot of input – output model and have the mathematical equation without describing the models relation before (Rumelhart, 1986). For these reasons, we used in this article BP ANN. The structure of BP ANN consists of input layer, hidden layer and output layer, which is shown in Figure 1 (Rossana et al., 2011). The first layer has input neurons which sends data via synapses to the second layer of neurons (hidden layer), and then via more synapses to the third layer of output neurons. More systems that are complex will have more layers of neurons, some having increased layers of input neurons and output neurons. The synapses store parameters called "weights" that manipulate the data in the calculations.

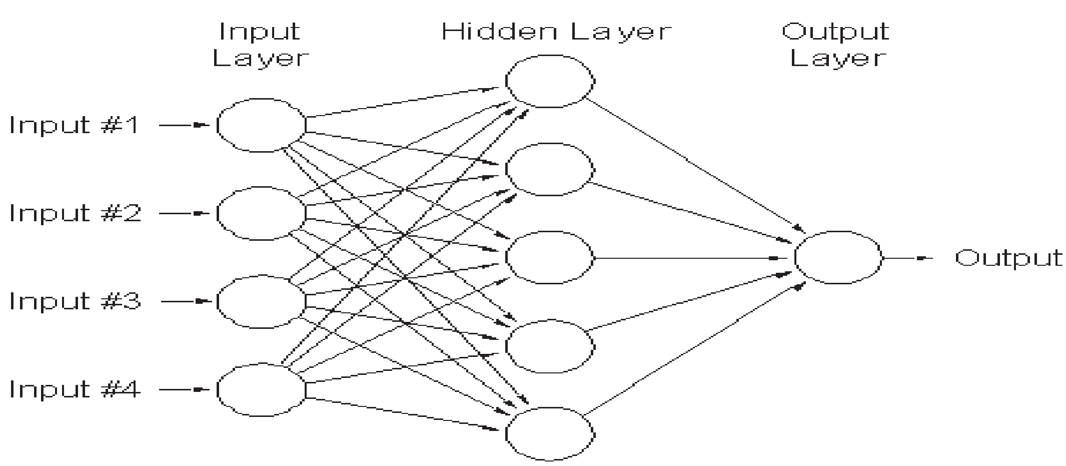

Fig.1

The structure of ANN structure.

Artificial Neural Networks ANN, as transformation systems of information founded in assignments need to link several types of information. This sort of assignments exists in many diagnostically processes such as the medical diagnosis. According to the kind and number of detailed measurements and researches (Amato, et. al., 2013), (Silva et al., 2010), the ANN structure as well as the way and duration of its learning must be selected to obtain the best diagnostically results with high specificity and sensitivity.

Figure 2 (Rossana et al., 2011) represents the block diagram of supervised learning. This diagram illustrates that the main objective is to minimize of the mean square error Q, D that represents the desired response (Rossana et al., 2011; Silva et al., 2010). Supervised learning involves the presentation of the data network together with the desired results. Therefore, we give a pair of complex networks z equation (1):

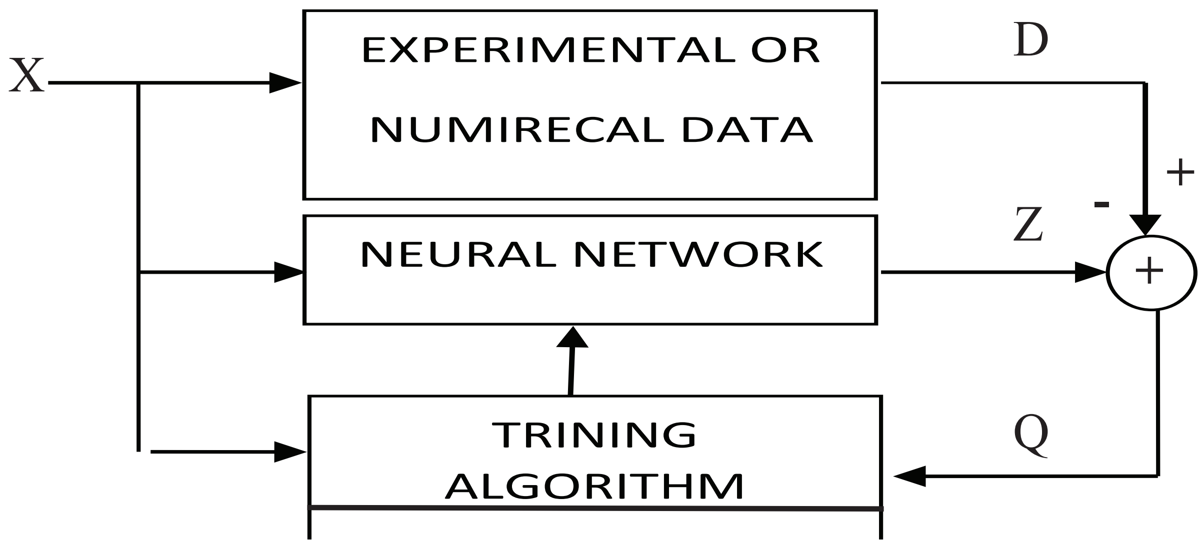

Fig.2

Block diagram of supervised learning.

$$\left(x_{(N\,x1)}\;,\; z_{(Mx1)}\right)=\left(\left\{x_1,\ldots..,x_N\right\},\left\{z_1,\ldots..,z_M\right\}\right) \quad (1)$$

Where

$$\left(x_{(N\,x1)}\;,\; z_{(Mx1)}\right)=\left(\left\{x_1,\ldots..,x_N\right\},\left\{z_1,\ldots..,z_M\right\}\right) \quad (1)$$

Where

input signal $x_{(N\,x1)}$

output signal $z_{(M\,x1)}$

Supervised learning is where we have input variables (X) and an output variable (Z) and we use an algorithm to learn the network function (f) from the input to the output equation (2).

$$Z = f\left(X\right) \quad (2)$$

The goal is to approximate the network function so that when we have new input data (X) then we can predict the output variables (Z) for that data.

It is clear that the network function (f) is not clear. However, we can learn network functions by using our input value, which is represented by an array of values. The use of relevant parameters describing the learning process are:

- Effectiveness: measured by the number of iterations K for established an acceptable error Q,

- Accuracy: error measured Q at established the number of iterations K.

In this study, we presented and reviewed an evolution experiment to determine the optimal values of Neural Network parameters. We did a performance analysis based on Backpropagation algorithm to train the Neural Networks.

The rest of the paper is outlined as follows; we gave a brief description about related researches. Section 3 presented the methodology used for experimenting the BP algorithm. Next, we analyzed and discussed about our approach experiment setup, results and discussions. Final section is the conclusion.

Supervised learning is where you have input variables (x) and an output variables (Z) and you use an algorithm to learn the mapping function from the input to the output.

$$Z = f(X)$$

The goal is to approximate the mapping function so well that when you have new input data (X) that you can predict the output variables (Z) for that data.

It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process. We know the correct answers, the algorithm iteratively makes predictions on the training data and is corrected by the teacher. Learning stops when the algorithm achieves an acceptable level of performance.

litreature REVIEW

Currently, there are many traditional algorithms and many alternative algorithms that specify a series of steps that perform a particular computation for the training of Artificial Neural Networks (ANN). These ANN algorithms use different optimization techniques (Peixoto et al., 2009). Backpropagation (BP) algorithm is very useful for training neural networks. Most commonly applied training algorithms are those derived from backpropagation algorithm (Widrow et al., 2013) (Li et al., 2012), (Dweib and Abuzir, 2012).

Li et al., presented the mathematical concepts and analysis of the characteristics of Backpropagation (BP) Artifcial Neural Network Algorithm and the different methods for improvement to achieve the minimum error sum of square (Li et al., 2012).

Hossam and El-Hawary (Hossam and El-Hawary, 2017) in their research to improve the performance of multilayer perceptron neural network (MLP) used Stochastic Fractal Search (SFS) to determine the optimal set of the MLP parameters. To solve the problems at the filtering stage, they applied a hybrid approach multilayer perceptron neural network - Frequency selective surface (MLP - FSS). They used the mean absolute percentage error index of the system state (phase and magnitude voltage) to determine the accuracy of the approach (MLP -FSS) (Silva et al., 2010).

Backpropagation is a term used for “backward propagation of errors”. The Backpropagation algorithm is used to train Artificial Neural Networks in conjunction with an optimization method (Li et al., 2009).

Rossana (Rossana et al., 2011) describe a hybrid EM-optimization for optimal design of FSS. The results show fact, accurate and improvement of MLP network.

To select more efficient features and emotion classifier based on Artificial Neural Networks, a hybrid method proposed based on gravitational search algorithm (GSA) and its binary version (BGSA). Mansour et. al., (Mansour et al., 2015) used this method to obtaining the optimal weight set and structure of a neural network.

Other researchers proposed a hybrid approach of combining a genetic algorithm (GA) with the Elman algorithm (GA–Elman neural network algorithm) to optimize the connection weights and thresholds, and improve the training speed and success rate (Zhao,, 2017).

Saishanmuga and Rajagopalan proposed a comparative study to optimize Neural Networks and to improve speed of recall and the efficiency of training. Their study shows that the Genetic Algorithm outperformed the other two algorithms Particle Swarm Optimization (PSO) and Ant-Colony Optimization (ACO) (Saishanmuga and Rajagopalan , 2014).

Other researchers use a Fast Fourier Transform (FFT) to determine the time-delay neural network (TDNN) input size (Taskaya et al., 2014). Different applications based on Neural network optimization including: approach for wind power prediction (Elattar , 2012), cutting parameters during cutting (Faping and Yuan , 2006), handwritten digit recognition (Parkins and Nandi, 2005), calculating the first-order derivative relationship between inputs and outputs in a trained neural network (Parkins and Nandi, 2005), Software Defect Prediction (Chug and Dhall, 2013) and Structural damage detection (Zhang, 2011; Arthur and Xiao-Hua, 2012), convert the nonlinear characteristics of turbine to torque and flow characteristics (Li et al., 2017)

MATERIALS AND METHODS

This section briefly reports the steps that we followed in carrying out our experiment in training the Artificial Neural Networks (ANN). It concisely summaries our own key steps which were taken in the experiment.

In this research, we want to study different experiments to improve the Artificial Neural Networks (ANN) and make it optimal. One of the most common ways to improve and optimize the Artificial Neural Networks (ANN) is Backpropagation (BP).

BP algorithm has multiple parameters, which may significantly affect the training of Artificial Neural Networks. Since there are no clear and established rules for selecting these parameters, our approach able to systematically obtain superior solutions to choose the best parameters for optimizing neural networks.

An ANN is typically defined by three types of parameters (Dweib,2015):

1. The activation function used to convert the weighted input of the neuron to its output activation.
2. The learning process for updating the weights of the interconnections.
3. Architecture (The interconnection pattern between the different hidden, input and output layers of neurons)

We propose a powerful technique that is based on the mathematical theory related to ANN to ensure fast optimization of the different parameters of ANN along with accuracy. We performed different simulation experiments to train Artificial Neural Networks (ANN) with BP and with delta rule, sigmoid transition function, learning coefficient for the first hidden layer and for the output layer.

This section contains the proposed methodology, which has been illustrated in Figure (3)

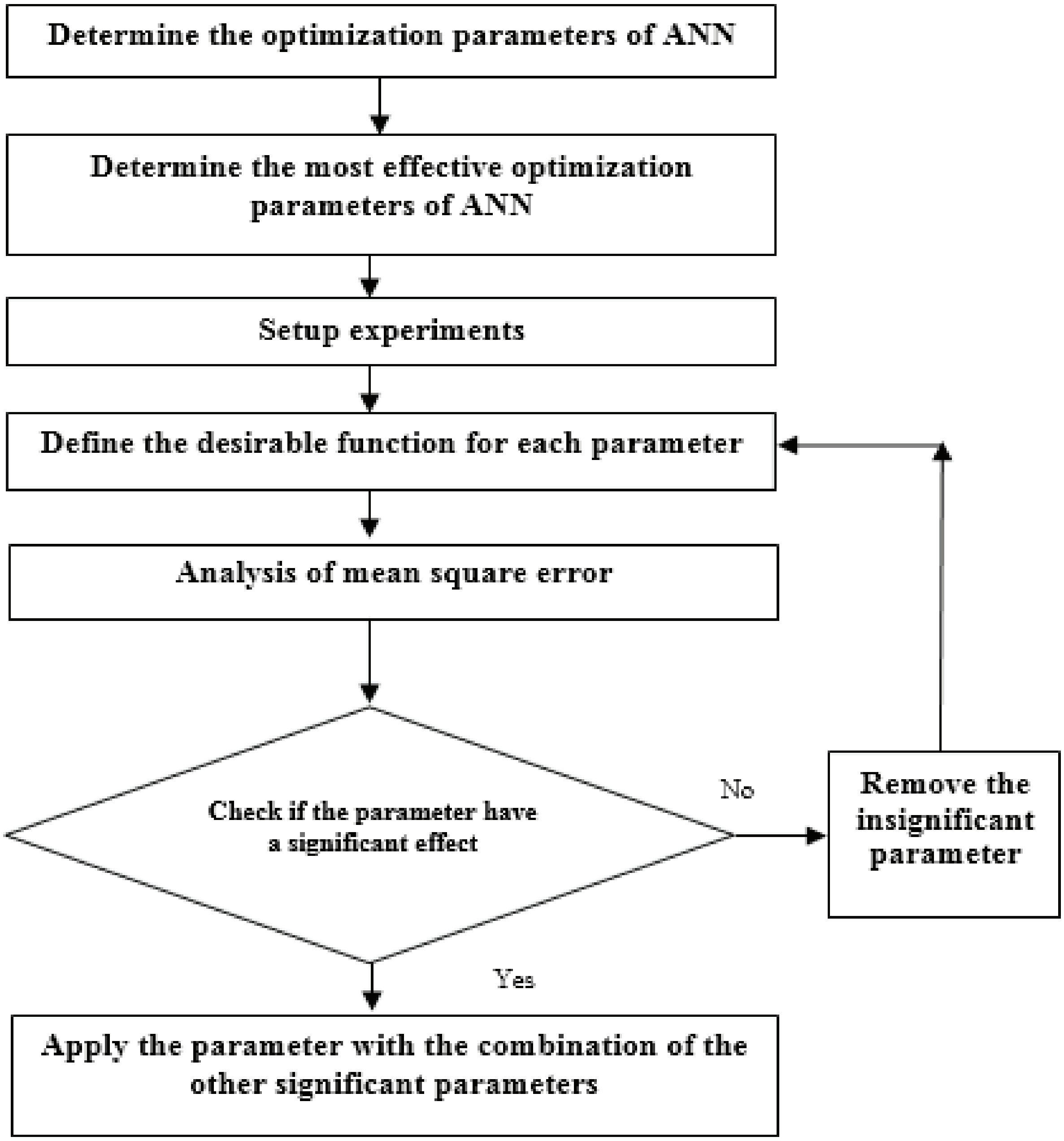

Figure 3
Proposed Approach

When we carried out our experiment, we usually follow a set of steps. We followed this procedure for our experiment to optimize ANN using BP and to determine the optimal parameters of Neural Network parameters. In our approach, we trained the Artificial Neural Networks for minimization of mean square error by following the main steps:

1. Determine the learning rule : we used two different functions

- Delta rule and
- Normal cumulative delta rule.

2. Determine the learning parameters for the hidden and output layers of the network to speed learning:

- Momentum (α),
- Transition functions - tangensoidal, sigmoid or simusoidal and
- Learning coefficient η

3. Determine the number of neurons in the hidden layer and the original input and Output.

The first considered performance criterion is Mean Square Error (MSE) between targets and outputs of the neural network. It is clear that this response is the Smaller. The Better (STB) type.

Our training dataset is collected from Wroclaw Hospital Heart Disease Dataset in Poland. There are a total of 142 patient records in the database. 87 patients records with heart attack who are between 45 and 68 years old and 55 students records (non-patients) aged between 20 and 23 years.

Results and Discussion

In this section, we present our results and discuss them by explaining the obtained results , commenting on these results and interpreting what the results mean. We present our result using figures, tables and discussions to clarify the experimental results in addition to validations results. Figure 4 shows system architecture. It is clear from figure 4 that the system imported the medical data and used BP ANN to train the neural network and to find our results (Table 1). These result imported to Statgraphics tool to find our result.

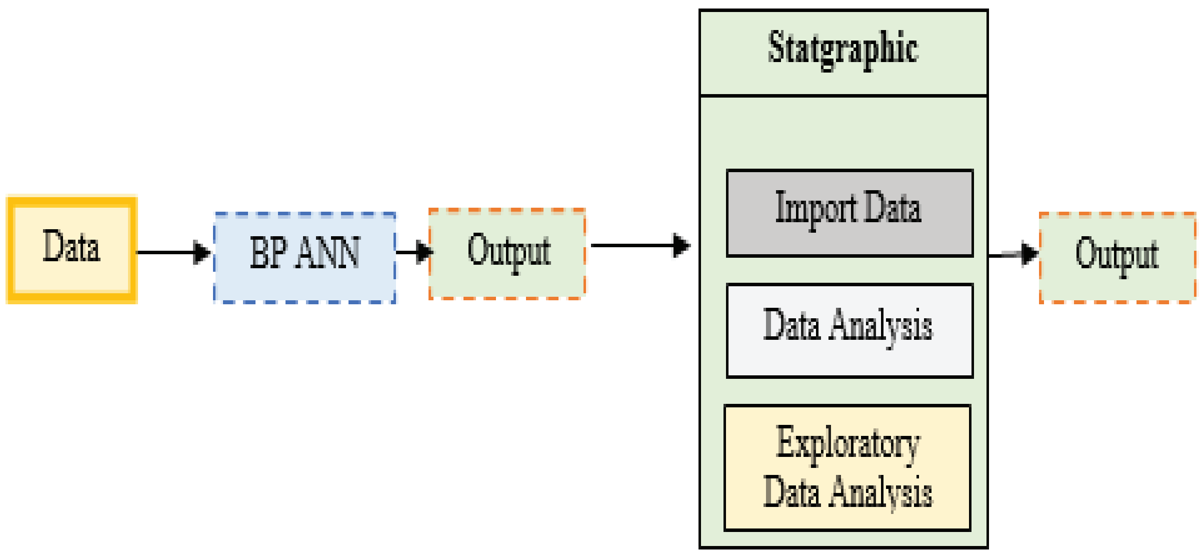

Fig 4

System Architecture

Determine the Learning Rule

Most ANNs contain some form of learning rulewhich modifies the weights of the connections according to the input patterns that it is presented with. The cost function is an important concept in learning, as it is a measure of how far away a particular solution is from an optimal solution to the problem to be solved. Learning algorithms search through the solution space to find a function that has the smallest possible cost which is the mean square error.

Before starting training network, we should determine the parameters of network learning. The first parameter we considered it in our study is the determination of the learning rule (Widrow et al., 2013) (Oksuz, 2007). Once we select the rule of the learning spaces for all layers of the network, the available options are:

- Delta Rule: if the weights changed, then it takes place immediately after their calculation. This means that, the neural network will calculate the new weights directly if values are changing.
- Normal Cumulative Delta Rule: if weight of connections between neurons changed, it takes place only at the end of an era.

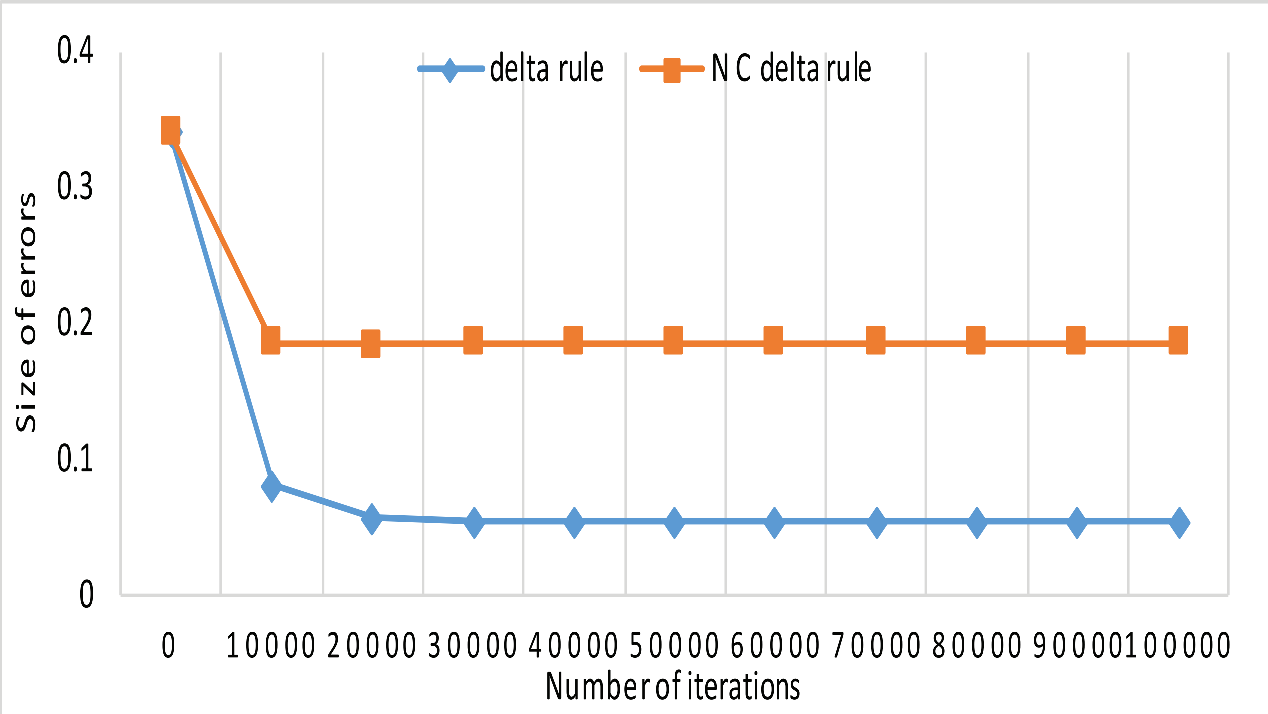

Fig. 5

Dependence of mean square error using Delta rule and NC delta rule of different iteration number

Figure. 5 clearly shows the performance comparison of using Delta rule and NC delta rule based on different number of iterations taken to train the neural network. The Mean square error is considered while evaluating the ANN training. Delta rule takes minimum mean square error to train the network. By using delta rule we got the smallest mean square error in the training process of ANN. According to this, we used it in this research. (Figure 4).

After training artificial neural network using BP ANN and our medical dataset as input and using delta rule and NC delta rule wereached the following results:

When we trained our ANN network function using our dataset for 10000 iterations,the error ratio was approximated to 0.1 for delta rule and 0.2 for NC rule. In the second phase we trained the ANN with 20000 iterations and we got an error 0.095 for delta rule and 0.2 for NC Rule. The figure shows diffrent iterations up to 1000000 iterations.

Determine the Parameters of the Learning

The learning process of the hidden and output layers of the ANN is based on adjusting the parameters of the network to reproduce an optimized value for mean square error. There are three types of parameters; the Momentum (α), Transition functions and the Learning coefficient η. In the following paragraphs we will discuss how to determine parameters of the learning for the hidden and output layers of the network:

- Momentum (α): the introduction of a momentum component gives the increasing speed of learning, improves its stability and revises the weights according to rule (Attoh-Okine, 1999; Saleem et al., 2017, Grossberg, 2013).

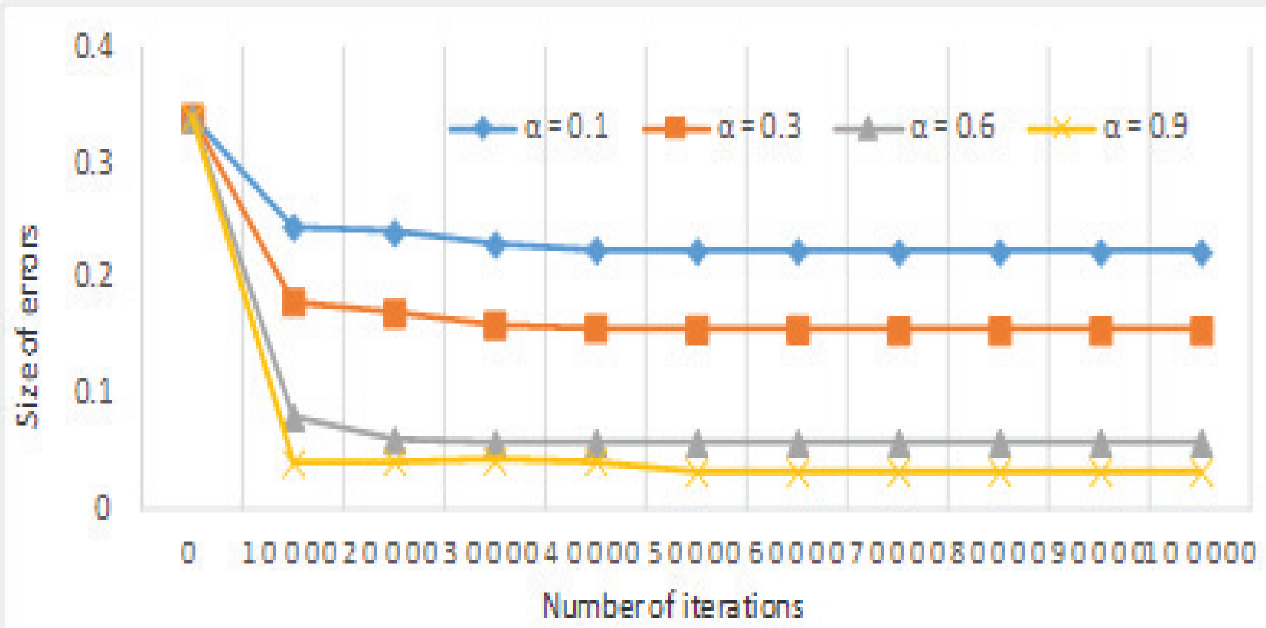

Fig. 6

Dependence of mean square error from coefficient momentum (α) of different iteration number

Figure 6 shows that we are training the ANN using the momentum. Refering to figure 6 it is clear that when learning the ANN using our dataset with only 10000 iterations the error ratio was approximately 0.05 for momentum = 0.9 and 0.089 for momentum = 0.5 and so on to 1,000,000 iterations. From the Figure (6), we can see, the smallest mean square error appears using coefficient momentum (α) when equals 0.9 (the larger the better).

- Transition functions - tangensoidal, sigmoid or simusoidal. The best results were obtained for sigmoid transition functions.

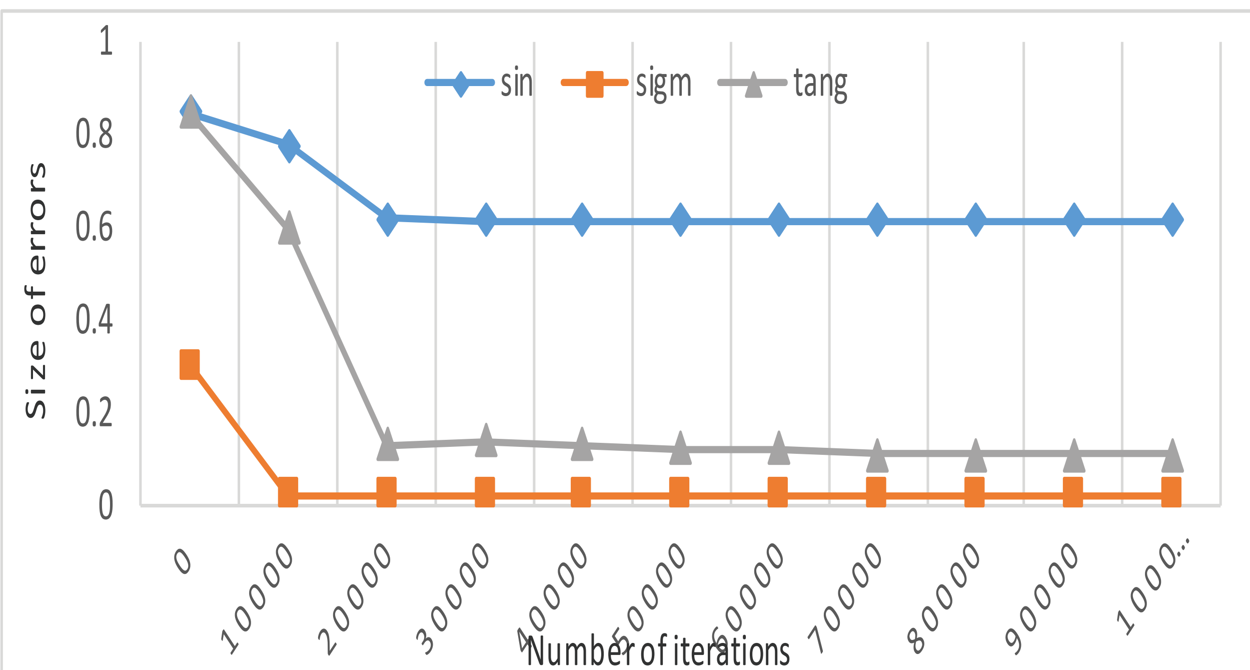

Fig.7

Dependence of mean square error from sinusoidal, sigmoid and tangensoidal transition functions of different iteration number

The third case is training our ANN using 10,000 iterations the error ratio was approximately 0.02 for sigm and 0.6 tangs. From the Figure (7), we can see, the smallest mean square error appears using sigmoid (the smallest the better).

In the second case, we consider two parameters (Learning Coefficient η and the Momentum α) in the learning process of the ANN. In the Backpropagation algorithm, the learning speed and the convergence of the algorithm are influenced by the selection of the two previous mentioned parameters (η and α) of the algorithm and the initial settings of the weight.

- Learning coefficient η : determines the speed of changing the weights' connections between neurons during learning. The big steps are large learning rate, and the convergence of so is faster. However, too large value of the parameter may cause too large oscillations, and the algorithm may prove to be unstable and not convergent. Unfortunately, there is no clearly defined method to choose the values. Podlak (Podolak et. al.., 2006) assumes that η =0.1, and Osowski proposes (Osowski et. al., 1996)

$$\eta \le \min(\eta_i) = \min\left(\frac{1}{n_i}\right).$$

Where : n_i - means the number of inputs of the ith neuron in the layer η - is determined separately for each layer.

However, it is technically difficult. Tadeusiewicz proposes η = 0.6. in the presence above state of affairs decided to give up the search for the optimal values of the coefficients of learning (η_1, η_2) method probe and blending proceedings and instead takes advantage of the design (planning) experiments method (Behera et al., 2005, Wang et al., 2013,).

This method allows for the selection of a mathematical model of the test input variables in order to facilitate experiments method by reducing the number of experiments. This can be traced to the series of researches (Dweib and Abuzir, 2012). In our case, for the different values of the coefficients η_1 and η_2 we obtained different values for the mean square error Q. We used Statgraphics (Statpoint, 2012) to find the results of the mean square error Q (Table 1) and analyz the results.

Table 1:

value η_1, η_2 and Q is printed to optimize (minimize) the error Q

Nr	η_1	η_2	Q	Nr	η_1	η_2	Q
1	0.2	0.15	0.7866	21	0.6	0.15	0.1879
2	0.2	0.20	0.0132	22	0.6	0.20	0.9230
3	0.2	0.25	0.2113	23	0.6	0.25	0.6320
4	0.2	0.30	0.4554	24	0.6	0.30	0.1420
5	0.2	0.35	0.6341	25	0.6	0.35	0.0511
6	0.2	0.40	0.1657	26	0.6	0.40	0.0634
7	0.2	0.45	0.3710	27	0.6	0.45	0.0967
8	0.2	0.50	0.0987	28	0.6	0.50	0.0189
9	0.2	0.55	0.9678	29	0.6	0.55	0.7210
10	0.2	0.60	0.8999	30	0.6	0.60	0.2555
11	0.4	0.15	0.1102	31	0.8	0.15	0.7550

Nr	η_1	η_2	Q	Nr	η_1	η_2	Q
12	0.4	0.20	0.0736	32	0.8	0.20	0.8660
13	0.4	0.25	0.0813	33	0.8	0.25	0.0789
14	0.4	0.30	0.0851	34	0.8	0.30	0.0623
15	0.4	0.35	0.0676	35	0.8	0.35	0.6321
16	0.4	0.40	0.0931	36	0.8	0.40	0.7492
17	0.4	0.45	0.0112	37	0.8	0.45	0.5776
18	0.4	0.50	0.0299	38	0.8	0.50	0.4661
19	0.4	0.55	0.2500	39	0.8	0.55	0.6632
20	0.4	0.60	0.0122	40	0.8	0.60	0.4666

In the next calculations, we obtained the graphs shown in the figure 6 obtained on the basis of the analysis of the results in table 1.

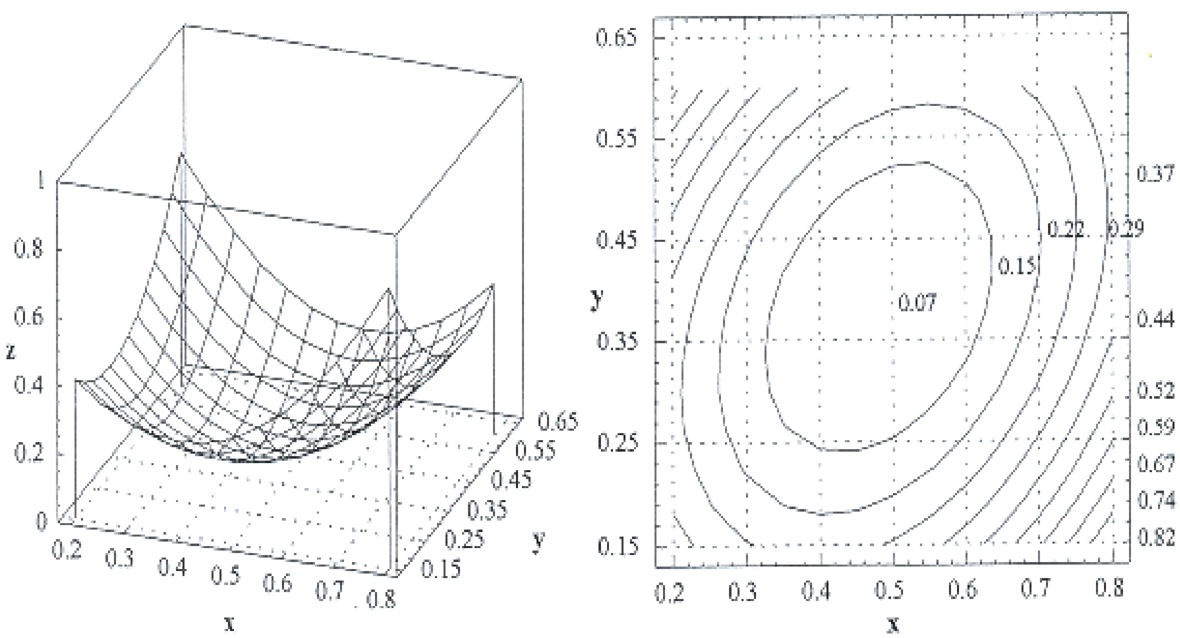

Figure.8

Relation of mean square error (axis z) from learning coefficient (η_1) the first hidden layer (axis x) and learning coefficient (η_2) output layer (axis y)

Statgraphics Centurion provides users with different tools to explore their result and present them in different graphical views. Figure 8 shows the results of the analysis procedure used by statgraphics tool based in the results botained from our BP ANN.

These results have different phases.They started by preparing the medical dataset and entering it into Excel sheet. In the second phase, we started training BP ANN using this data. The third phase is obtaining the reult of BP ANN and store it as text file or any other file supported by Statgraphics. The forth phase is importing this data to Statgraphics to anaylse it and present the result in grphical view. As shown in figure 8 the statgraphics shows our results in a surface.

Figure 8 shows the influence of the η on the weight changes in the learning of ANN. From the figure and the analyzed network, we found that the least mean square error is obtained when the coefficient of the learning set for the first hidden layer is 0.5 and the learning coefficient of the output layer is 0.37. while learning the value η_1 should be reduced by 0.1 in order to ensure better convergence of the learning.

This example will serve us to formulate a mathematical model, whose objective is to achieve the smallest mean square error $Q = f(\eta_1, \eta_2)$. It should be noted that if the current value is optimal error, it will be placed in the lowest area of the surface of the experiment. Mathematical model, which is characterized by error Q as a quadratic function of the parameters is given in polynomial forms equation 3:

$$Q = A + B\eta_1 + C\eta_2 + D\eta_1^2 + E\eta_2^2 + F\eta_1\eta_2 + \varepsilon \quad \ldots\ldots..(3)$$

Where:

A, B, C, D, E, F - polynomial coefficient

ε - measurement error (systematic)

The horizontal projection graph shows the value throws an error on the plane x and y (each line on the graph shows the error of the same value (figure 6). Error value does not change rapidly if η_1 and η_2 are only deviated from their best values, which takes place gradually. This allows us to choose the scope of the smallest mean square error (Q).

Determining the Number of Neurons

The determination of the number of the hidden layers and the of neurons (H) in the hidden layers is very important as it affects the training time, finding the optimum values of the coefficients of learning and generalization property of Artificial Neural Networks. To minimize the error and train an Artificial Neural Networks (ANN) that find and generalize the optimum values of nodes and the hidden layers, we need to pick an optimal number of hidden layers, as well as nodes in each hidden layer.

In general, the number of hidden nodes in hidden layer is based on a complex relationship between number of input and output nodes. Therefore, in our research we are interested in determining the number of neurons in the hidden layer and the original input and Output:

- Input layer neurons are 24 (features).
- Output is the layer 1 neuron, whose states respectively indicate two classes: 0 for healthy and 1 for patients (Dweib and Abuzir, 2012).
- Still remains unresolved problem of the number of neurons in the hidden layer. If the number of neurons in the hidden layer has too small neurons, this will cause large classification error, while large numbers of neurons cause reduction in the number of correct classifications for other data than in a set of learning, then the learning process will create the possibility of overtraining, that is too exact matching answers to the small set of details. It is often the learner or arising from measurement errors, or the phenomena blur the data, which in turn reduces the degree of generalization (one of the most important properties ANN). Because of this, the rule does not exist and determine the number of neurons in the hidden layers and make in method experiment and errors. It was decided to proceed as in the case of finding the optimum values of the coefficients of learning and apply any method for the design of the experiment in order to find the optimal range of values, which would minimize the mean square error

$$RMS = \sqrt{\frac{1}{n}\sum_{i=0}^{n-1}(t_i - \sigma_i)^2}, \quad (3)$$

Where t_i - desirable output,

σ_i - resulting output

The influence of the Momentum α on the weight changes is shown in Figure 7. The graph in Figure (9) shows the relation of mean square error (axis z) from the number of hidden layers the first hidden layer (axis x) and the number of neuron in hidden layer (axis y).

It is noted that for the analyzed networks smallest error is achieved at 12 neurons in two hidden layers for an exercise stress test. Experimental assumed that if we have two hidden layers, the ratio of the number of neurons in the second layer to the first layer is 0.5.

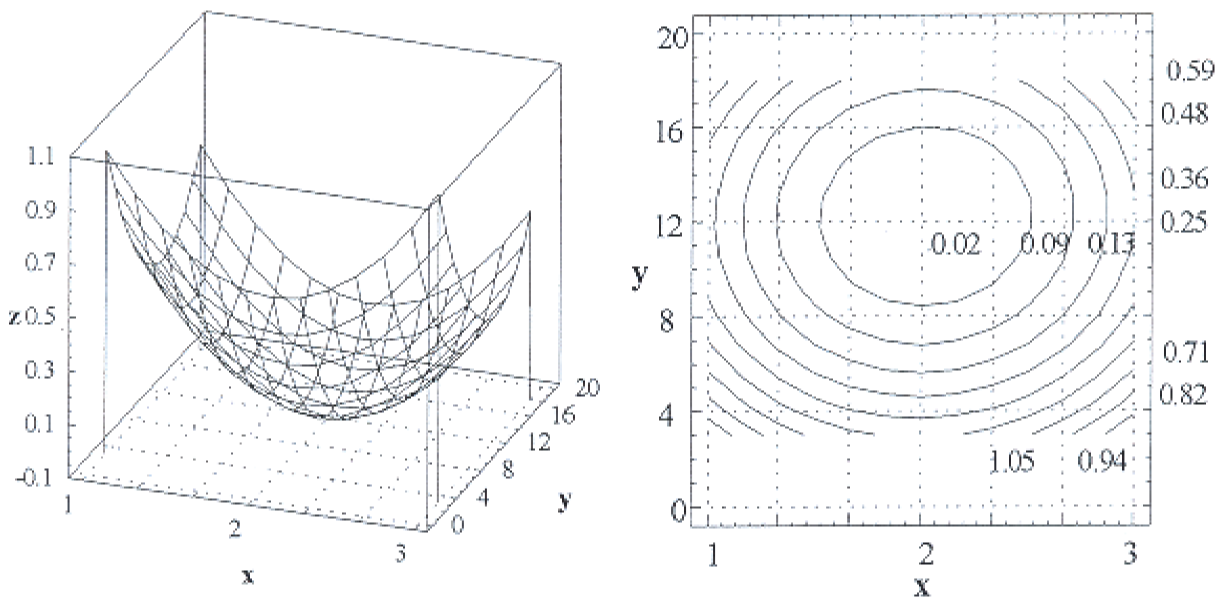

Figure.9

Relation of mean square error (axis z) from the number of hidden layers the first hidden layer (axis x) and the number of neuron in hidden layer (axis y)

CONCLUSION

In our approach, we included in the proposed method the optimization process for the parameters in BP ANN in order to avoid irrelevant features over the optimization process. In this research we used the size of mean square error to find the optimal parameters of the neural network such as the coefficient of learning, the optimal number of hidden layers and number of neurons in that layer. This kind of solution did not helped in finding the heuristic rules to determine the neural network structure.

In our study, the results showed that the most important parameters of the network learning method BP are delta rule, the value of momentum (0.9), sigmoid transition function, the learning coefficient (0.5 for the first hidden layer, 0.37 for the output layer for a network of scholarly method of BP.

In the current work, we have reviewed the optimization of BP algorithms for neural networks based on number of neurons and the number of layers. We found that the optimization algorithm used 12 neurons in the two hidden layer to get smallest error of the mean.

For demonstrating the goodness of the results of the proposed method, a comparison between the Counter back-propagation (Dweib, 2015) and the proposed method is performed. Results of the comparison show that all the considered optimization criteria have a similar situation as in the proposed approach.

As a future study, other experimental designs can be used for better ANN parameter tuning. Moreover, other parameters can be studied for the tuning of artificial neural networks.

References

1. Amato Filippo, López Alberto, MaríaPeña-Méndez Eladia, Vaňhara Petr, Havel Josef, Artificial neural networks in medical diagnosis, Journal of Applied Biomedicine, Volume 11, Issue 2, 2013, Pages 47-58.

2. Arthur S. and Xiao-Hua Y., “Structural damage detection using artificial neural networks and wavelet transform”, 2012 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, July, 2012.

3. Attoh-Okine,N. O., Analysis of learning rate and momentum term in backpropagation neural network algorithm trained to predict pavement performance, in Advances in Engineering Software 30(4):291-302 · April 1999 DOI: 10.1016/S0965-9978(98)00071-4

4. Behera Laxmidhar, Swagat Kumar and Awhan Patnaik, On Adaptive Learning Rate That Guarantees Convergence in Feedforward Networks, IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 17, NO. 5, SEPTEMBER 2006.

5. Chug A., Dhall S.: Software Defect Prediction Using Supervised Learning Algorithm and Unsupervised Learning Algorithm : Confluence 2013: The Next Generation Information Technology Summit (4th International Conference), 2013, p. 5.01 - 5.01.

6. Dweib M. and Abuzir Y., A Reduction Algorithm for Analysis of Linear Neuron,Journal of Computer Science, Volume 8, Issue 4, Pages 533-537, 2012, ,DOI: 10.3844/jcssp.2012.533.537 .

7. Dweib M., Architecture Setting of Learning Parameters Based on Counter Back-Propagation Neural Networks, VOL.5, NO.12, December 2015 ISSN 2225-7217 ARPN Journal of Science and Technology, http://www.ejournalofscience.org/archive/vol5no12/vol5no12_3.pdf

8. Elattar Ehab Elsayed: Prediction of wind power based on evolutionary optimised local general regression neural network, IET Generation, Transmission & Distribution, Volume 8, issue 5, 2014 , p. 916 – 923.

9. Ghosh R., Ghosh M., Yearwood J., and Bagirov A., “Comparative analysis of genetic algorithm, simulated annealing and cutting angle method for artificial neural networks,” 2005, pp. 62–70 .

10. Grossberg, S. (2013). Adaptive Resonance Theory: How a brain learns to consciously attend, learn, and recognize a changing world. Neural Networks, 37, 1-47

11. He, Y., Qiu, Y., Liu, G., et al. (2005). “Optimizing weights of neural network using an adaptive tabu search approach”, In The second international symposium on neural networks 2005 (ISNN-2005) (pp. 672–676), Chongqing, China.

12. Hossam Mosbah and M.E. El-Hawary , Optimization of neural network parameters by Stochastic Fractal Search for dynamic state estimation under communication failure , Electric Power Systems Research, Volume 147, June 2017, Pages 288-301.

13. LeCun, Y., “Learning Processes in an Asymmetric Threshold Network,” Disordered Systems and Biological Organization, 233-40. Berlin: Springer Verlag, 1986.

14. Li J., Canfeng Han, and Fei Yu, “A New Processing Method Combined with BP Neural Network for Francis Turbine Synthetic Characteristic Curve Research,” International Journal of Rotating Machinery, vol. 2017, Article ID 1870541, 11 pages, 2017. doi:10.1155/2017/1870541

15. Li J., Cheng J., Shi J., Huang F., Brief Introduction of Back Propagation (BP) Neural Network Algorithm and Its Improvement. In: Jin D., Lin S. (eds) Advances in Computer Science and Information Engineering. Advances in Intelligent and Soft Computing, vol 169. Springer, Berlin, Heidelberg, 2012.

16. Li, Y.; Fu, Y.; Li, H.; Zhang, S. W. (2009-06-01). “The Improved Training Algorithm of Back Propagation Neural Network with Self-adaptive Learning Rate”. 2009, International Conference on Computational Intelligence and Natural Computing. 1: 73–76. doi:10.1109/CINC.2009.111

17. Mansour Sheikhan, Mahdi Abbasnezhad Arabi & Davood Gharavian, Structure and weights optimisation of a modified Elman

network emotion classifier using hybrid computational intelligence algorithms: a comparative study, Journal Connection Science, Volume 27, Issue 4, Pages 340-357, 2015.

18. Oksuz A., 2007, Unsupervised Intrusion Detection System, Publisher : Informatics and Mathematical Modelling, Technical University of Denmark, DTU Series:IMM-Thesis-2007-20,p.p15 viewed 15 Nov 2013. Available at: http://etd.dtu.dk/thesis/200720/imm5193.pdf

19. Parkins A.D., Nandi A.K.: Method for calculating first-order derivative based feature saliency information in a trained neural network and its application to handwritten digit recognition: IEE Proceedings - Vision, Image and Signal Processing, Volume 152, issue 2, 2005 , p. 137 – 147

20. Peixoto, H. M.; Cruz, R. M. S.; Guerreiro, A. M. G.; Dória Neto, A. D. & D'Assunção, A. G. (2009). A comparison of multilayer perceptrons training algorithms for the optimization of frequency selective surfaces, Proceedings of 8th International Information and Telecommunication Technologies Symposium, Florianópolis, SC, Brazil, 2009.

21. Podolak, J., Shilane, P., Golovinskiy, A., Rusinkiewicz, S. and Funkhouser, T., A Planar-Reflective Symmetry Transform for 3D Shapes', ACM Transactions on Graphics (SIGGRAPH2006), Vol.25, No.3, (2006).

22. Rossana M. S. Cruz, Helton M. Peixoto and Rafael M. Magalhães (2011). Artificial Neural Networks and Efficient Optimization Techniques for Applications in Engineering, Artificial Neural Networks - Methodological Advances and Biomedical Applications, Prof. Kenji Suzuki (Ed.), InTech, DOI: 10.5772/15293. Available from: https://www.intechopen.com/books/artificial-neural-networks-methodological-advances-and-biomedical-applications/artificial-neural-networks-and-efficient-optimization-techniques-for-applications-in-engineering

23. Rumelhart, D., Hinton, G., & Williams, R., "Learning Internal Representations by Error Propagation," In D. Rumelhart & J. McCleeland (Eds.), Parallel distributed processing: Explorations in the microstructure of cognition (Vol. 1). Cambridge, MA: MIT Press, 1986.

24. Saishanmuga Raja and Rajagopalan, P., A Comparative Analysis Of Optimization Techniques For Artificial Neural Network In Bio Medical Applications, Journal of Computer Science 10 (1): 106-114, 2014, doi:10.3844/jcssp.2014.106.114 Published Online 10 (1) 2014 (http://www.thescipub.com/jcs.toc) .

25. Saleem W., Zain-ul-abdein M., Ijaz H., Mabrouki T., Computational Analysis and Artificial Neural Network Optimization of Dry Turning Parameters-AA2024-T351,in Applied Sciences, Jun 2017·

26. Silva, P. H. da F.; Cruz, R. M. S & D'Assunção, A. G. Blending PSO and ANN for optimal design of FSS filters with Koch island patch elements, IEEE Transactions on Magnetics, Vol. 46, No. 8. . (2010)

27. Statpoint Technologies, Statgraphics Centurion XVI English Paperback , ASIN: B00K5S5YHQ, 2012.

28. Tadeusiewicz Ryszard, Chaki Rituparna Chaki, Chaki Nabendu , Exploring Neural Networks with C#, CRC Press, Sep 2014.

29. Taskaya-Temizel T., Casey M.C., K. Ahmad: Pre-processing inputs for optimally-configured time-delay neural networks, Electronics Letters, Volume 41, issue 4, February 2005, p. 198 - 200

30. Wang, G., Hao, J., Ma, J., & Huang, L. . A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering. Expert Systems with Applications, 37(9), 6225-6232, 2010.

31. Widrow B. , Greenblatt A. , Kim Y. , Park D. , The No-Prop algorithm: A new learning algorithm for multilayer neural networks, Neural Networks 37 (2013) 182–188.

32. Zhang Faping, Sun Houfang, Lu Yuan :

Optimization of cutting parameters during cutting based on the FEA and neural networks, International Technology and Innovation Conference 2006 (ITIC 2006), 2006, p. 1704 - 1709

33. Zhang J., "Structural damage detection using parameters combined with changes in flexibility based on BP neural networks", 1st International Conference on Civil Engineering, Architecture and Building Materials, (2011) June 18.

34. Zhao, W., Zheng, D., Y. et al.,A novel optimized GA–Elman neural network algorithm, Jia, Neural Comput & Applications, pp 1–11, (2017). doi:10.1007/s00521-017-3076-7 .