ISSN 2278-3091
Volume 10, No.2, March - April 2021
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse671022021.pdf
https://doi.org/10.30534/ijatcse/2021/671022021

A Comparison between Seven Heuristic Methods to Estimate the Number of Hidden Layer Neurons in a Multi Layer Perceptron Neural Network

Mutasem Sh. Alkhasawneh
Software engineering Department, faculty of science and information technology,
Ajloun National University, 26810, Ajloun, Jordan

ABSTRACT

Multilayer Perceptron Neural Network (MLPNNs) constructs of input, at least one hidden and output layer. Number of the neurons in the hidden layer affects the NNs performance. It also consider difficult task to overcome. This research, aims to exanimate the performance of seven heuristic methods that have been used to estimate the neurons numbers in the hidden layer. The effectiveness of these methods was verified using a six of benchmark datasets. The number of hidden layer neurons that selected by each heuristic method for every data set was used to train the MLP. The results demonstrate that the number of hidden neurons selected by each method provides different accuracy and stability compared with other methods. The number of neurons selected by Hush method for ine data set was 26 neurons. It's achieved the best accuracy with 99.90%and lowest accuracy achieved by Sheela method with 67.51% using 4 neurons. Using 22 neurons with 97.97% accuracy Ke, J method received the best result for Ionosphere data set. While the lowest accuracy was 96.95% with 5 neurons achieved by Kayama method.For Iris data set with 8 neurons achieved 97.19 as best accuracy achieved by Hush method. For the same data set the lowest results were 92.33 % using 3 neurons obtained by using Kayama method. For WBC data set 96.40% the best accuracy achieved using Sheela and Kaastra methods using 4 and 7neurons, while Kanellopoulos method achieved the lowest accuracy 94.18% with 7neurons. For Glass dataset, 87.15% was the best obtained accuracy using 18 neurons Hush method and using Wang method 82.27 % with 6 neurons was the lowest accuracy. Finally for PID 75.31% accuracy achieved by Kayama method with 3 neurons, where Kanellopoulos method obtained 72.17% through using 24 neurons.

Key words: Pruning Neurons, Neurons Number, Hidden layer, MLP stability, MLP performance, Heuristic Methods.

1. INTRODUCTION

Artificial Neural Networks (ANNs) are widely used in many fields. It has different applications such as letter recognition, landslide hazard prediction medical classification, pattern recognitions and more [1-6]. The popular use of the ANN comes from its finite stability, parameterization, computational simplicity, smaller structure size for a particular problem, high accuracy for learning and good robustness [4].

Multilayer Perceptron (MLP) Neural Network is one of the different types of neural networks. It consists of a single input layer, one or more hidden layers and a single output layer. However, there are two major problems facing researchers when selecting a suitable structure for an ANN the number of hidden layers and the number of neurons in the hidden layer. A neural network with a single hidden layer is usually enough to solve most problems such as prediction and classification [7-10]. Furthermore, finding out the number of hidden neurons is very important, since it can Influence the complexity and accuracy of the neural network and thus, influence the error on the neurons to which their output is connected. Estimating the number of neurons in the hidden layer is an important stage in classification, prediction and in any application of the neural network; especially if it involves a large dimension of input factors. Furthermore, using the right number of the neurons in the hidden layer can improve the performance of the neural network [11].

The stability of the neural network is estimated by error. Less error reflects higher stability, and more error reflects less stability. Using excessive hidden neurons will cause over fitting,that is, the neural networks over-estimate the complexity of the target problem[12]. So far, different heuristic methods have been proposed to estimate the number of hidden neurons and most of these methods are based on trial and error.This research aims to do a comparison between seven different methods that are used to estimate the number of neurons in the MLP hidden layer. These heuristic methods are introduced by Sheela et.al [12], Ke, J et. al [11], Hush [13]Kanellopoulos et. al [14], Kaastra et. al [15], Kayama et. al [18]Wang [19].

2.PREVIOUSWORKS

Different methods have been proposed to estimate the number of hidden neurons. Some of these were evaluated by using MLP and Elman neural networks [13]while others wereevaluated using Fully Connected Cascade (FCC) neural network.

Sheela and Deepa [14]Proposed a new method to fix the number of hidden neurons in Elman networks, based on statistical errors and convergence theorem.The proposed method was tested using wind speed prediction in renewable energy systems. Moreover, 10000 sample were used to train and test Elam networks; divide into 7000 sample for training and 3000 for testing. The performance of NN were measured using MSE ,Mean Absolute Error and mean relative error (MRE), and Mean Absolute Error (MAE).
The proposed model improves accuracy and reduces error as in Eq. 1. where$n$ is the number of input Features.

$$N_h = \left(\frac{4n^2 + 3}{n^2 - 8}\right) \quad (1)$$

Ke and Liu [12] estimated the number of neurons in the hidden layer by proposing a model for stock market prediction. It represented a sensitivity analysis of the optimal number of hidden neurons and hidden layers in a neural network. The result with the minimal estimated generalization error was designated optimum for the application of Neural Network model. To run the experiment 1000 samples were used. In addition neural network were tested by changing the number of hidden layer from zero to 3 layers and the R^2 are used to evaluate the NN.

Equation 2explainshow the proposed method estimates the number of neurons in the hidden layer. Nhis the number of the neurons in the hidden layer, Np and Nin are the number of Featurein input layer and Number of samples whereas L is the number of hidden layers.

$$N_h = \frac{\left(N_n + \sqrt{N_p}\right)}{L} \quad (2)$$

Moreover, Hush [15]Proposed a new method to estimate the number of hidden neurons. The MLP neural network performance was analyzed based on different factors included the number of neurons inthe hidden layer. Four different training sample groups were used to train NN, the performance of NN estimated based on the error rate. A new formal to estimate the number of hidden neurons as in Eq.3, was proposed, where N_h is the number of neurons in the hidden layer, N_i is the number of input to the input layer.

$$N_h = 3N_i \quad (3)$$

Kanellopoulos and Wilkinson[16]proposed an improvement on the method proposed by Hush [15].The experiment was carried out on real datasets collected from satellites for five years. The research handled the number of hidden layers and the number of hidden neurons. As a result, the number of the hidden neurons (N_h) could be calculated as double the number of input parameters (N_i) as shown in Eq. 4. where

$$N_h = 2N_i \quad (4)$$

Kaastra and Boyd [17] used financial and economic time series forecasting as a dataset to estimate the number of

$$N_h = \left(\sqrt{1 + 8n} - 1\right)/2 \quad (6)$$

hidden neurons in the MLP neural network with back-propagation as a learning algorithm. Different steps were used to predict the number of hidden layers and neurons. These steps start from choosing a different number of layers with a different number of neurons in the hidden layer and later on choose the best MLP structure. Experiments were carried out multiple times with 50 samples and the results were evaluated based on accuracy and processing time. Equation5 is used to estimate the number of the required neurons in a hidden layer.

$$N_h = \sqrt{N_i * N_l} \quad (5)$$

Where N_h, N_i and N_l are the number of the neurons in the hidden layer, input Features and output neurons respectively.

Jin-Yan, and Ying-Lin [18] improved the theory proposed by previous study[19]. The developed method was applied and tested to the problems of the prediction of time series and system identification by higher-order neural network.

Kayama [20] mentioned for the first time the importance of the number of features samples as shown in Eq. 6.

Where N his the number of the hidden neurons and nis the number of feature samples. Regression analysis model was used to evaluate the NN.

Wang [21] proposed that the number of neurons in the hidden layer is double the input parameters divided by three as shown in Eq. 7.The NN performance was evaluated using regression model.

$$N_h = 2 * N_i / 3 \qquad (7)$$

Where N_h is the number of the hidden neurons and nis the number of training samples.

2.MULTI LAYER PERCEPTRON NEURAL NETWORK

Artificial Neural network build of input, hidden and output layer. It is based on units called neuron.

2.1 Neuron

The single neuron receives input and applies a mathematical function to make decision on that output. Figure 1 illustrates the single neuron.

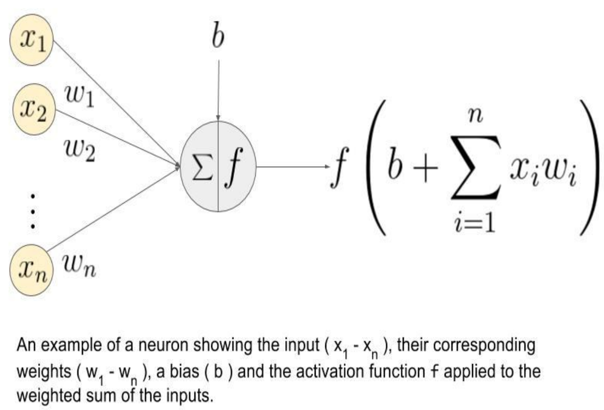


Figure 1: Example of single neuron.[22]

Every input to neuron has a weight that connects between inputs and neurons. In addition, Neuron receives another weight called bias. It used to avoid absent value of any neuron output. Each neuron has to make decision and this can be done using an activation function (the mathematical function). Different activation function can be used such as Rectified Linear Units and Sigmoid activation. As shown in Figure. 1 the activation function receives the input of any neuron multiplies by a random weight. The summation and add the biased value is used to trigger the neuron.

2.2 Multilayer Perceptron Neural Network Model

The MLP network is a feed-forward artificial neural network model composed respectively of a single input layer, one or more hidden layers, and a single output layer [23].
Figure 2 shows the MLP output network with m outputs and n_h hidden nodes as in following Eq. 8 [24]:

$$\tilde{y}_k(t) = \sum_{j=1}^{n_h} w_{jk}^2 F\left(\sum_{i=1}^{n_i} w_{ij}^1 x_i(t) b_j^1\right); \qquad (8)$$

For $1 \le j \le n_h$ and $1 \le k \le m$

w^1ij and w^2jk denote the weight between the input & hidden layers respectively.
By xi and b^1jrepresent bias values of the input nodes and hidden nodes respectively.

n_i is the number of input nodes and n_h is the number of hidden nodes. K is number of input
Linear and sigmoid were chosen as transfer functions *F(*)* for the output and input layers respectively.

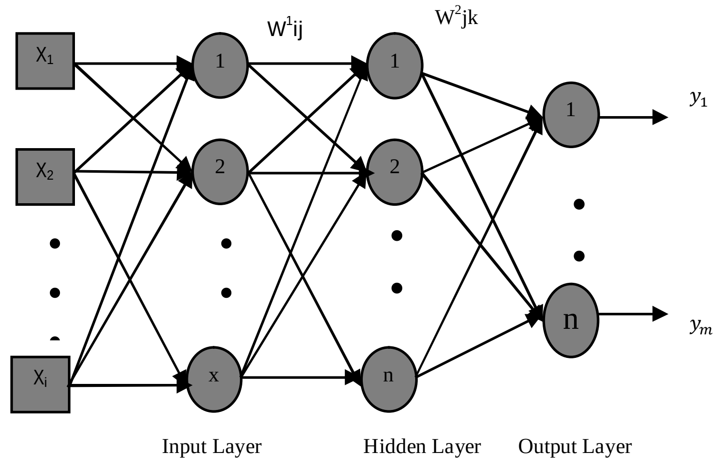
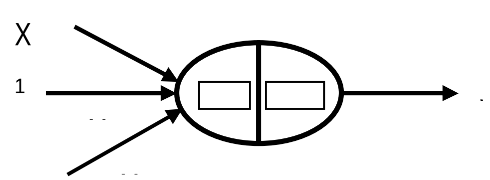


Figure 2: The conventional Multi Layer Perceptron network.

However, Multilayer Perceptron neural network is considered an example of nonlinear neural network. In the case of a linear system, it has to be approximated using the nonlinear MLP network model. Moreover, modeling a linear system using a nonlinear model can never be better than using a linear one[23, 25, 26].

5. EXPERIMENTAL SETUP

The data used in training the MLP will be normalized between 0 and 1, then the normalized data will be divided so that 70% of it will be used for training and 15% it will be used for testing and likewise for validations. Moreover, the number of neuron in the hidden layer will be calculated based on the seven heuristic methods. Figure. 3show the methodology of this work.



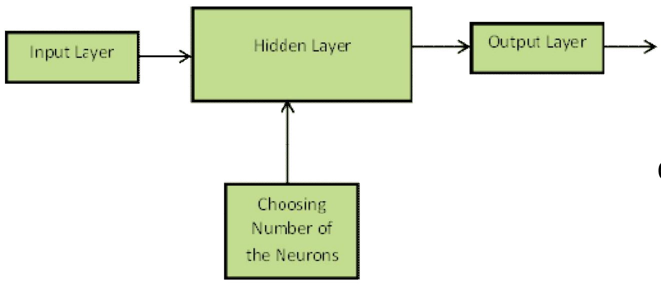


Figure 3:work methodology

5.1 Benchmark Dataset

Six benchmark data sets were used to evaluate the proposed model performance. The benchmark dataset is obtained from machine learning repository of the University of California, Irvine (UCI). Datasets used were Wine, Ionosphere, Iris, Wisconsin Breast Cancer (WBC), Glass and Pima Indians Diabetes (PID). Table 1 summarizes the number of training samples, the number of prediction features and the number of classes for the benchmark datasets. The description of each dataset is as follows.

Wine: 178 samples of wine represent three different classes. Itis divided into 59 samples representing the first class, 71 samples representing the second class, and 48 samples for the third class. Features included in each class are: hue, alcohol, OD280/OD315 of diluted wines and proline, color intensity, malic-acid, ash, non-flavonoid phenols, alkalinity of ash, flavonoids, magnesium, total phenols and proanthocyanidins. Samples were gathered from Italy and from the same region.

Ionosphere: It is a radar data with two classes; good and bad. If the returned radar data return with some evidence from the ionosphere, the class is considered good. It is considered bad if the data obtained from ionosphere do not include any structural evidence. The samples collected for this research were 352. Each sample has 34 features.

Iris: 150 samples of iris data were divided into three classes each of which contains four features. The three classes have 50 samples per class. The three classes are: iris virginica, iris versi colour and irissetosa. Petal width, petal length, sepal width and length are the four features comprising each sample.

WBC: 699 samples of patients with Breast cancer were obtained. Breast cancer patients can be divided into two classes: malignant with 241 samples and benign with 458 classes. Each class contains nine features: mitosis and normal nucleoli, clump thickness, bland chromatin, uniformity of cell size, bland chromatin, uniformity of cell shape, bare nuclei, marginal adhesion, and single epithelial cell size

Glass: this dataset is popular. It contains 214 samples with 6 classes and 9 features. The number of samples for each class is as follows: 29 samples for headlamps, 70 samples for float processed building windows, 76 samples for non-float processed building windows. 9 samples for tableware, 17samples for float processed window vehicles and 13 samples for containers.

6. PID: the PID dataset contains 768 cases. 268 cases represent diabetes patients while healthy cases are 500. Eight features were contained in the healthy and diabetes cases.

Table 1: Benchmark dataset features

Benchmark data	Cases	Number of Features	Class
PID	768	8	2
WBC	699	9	2
Iris	150	4	3
Ionosphere	352	34	2
Glass	214	9	6
wine	178	13	3

5.2 Optimal Number of Hidden Neuron

In this study seven heuristic methods stated in Table 2 are used to calculate the number of hidden neurons for each data set. Table 2 shows the heuristic method, data set and the number of feature for each data set. In addition, the calculated hidden neuron number based on the heuristic method, and features of each data set.

Table 2. Number of hidden neuron for the six datasets based on the seven Heuristic Methods.

Heuristic Formula	Data set	Number of Features: Where *n* and N_i and N_n is the number of input features, N_p is the number of input samples, *L* is the number of hidden layer=1;	Hidden node Calculated
$N_h = \left(\frac{4n^2+3}{n^2-8}\right)$ (1) Sheela and Deepa [12]	Wine	13	4
	Ionosphere	34	4
	Iris	4	8
	WBC	9	4
	Glass	9	4
	PID	8	5
$N_h = \frac{\left(N_n + \sqrt{N_p}\right)}{L}$ (2) Ke and Liu [12]	Wine	13	26
	Ionosphere	34	52
	Iris	4	16
	WBC	9	35
	Glass	9	23
	PID	8	26
$N_h = 3N_i$ (3) Hush [15]	Wine	13	39
	Ionosphere	34	102
	Iris	4	12
	WBC	9	27
	Glass	9	27
	PID	8	24
$N_h = 2N_i$ (4) Kanellopoulos and Wilkinson [16]	Wine	13	26
	Ionosphere	34	68
	Iris	4	8
	WBC	9	18
	Glass	9	18
	PID	8	16
$N_h = \sqrt{N_i * N_l}$ (5) Kaastra and Boyd [17]	Wine	13	6
	Ionosphere	34	8
	Iris	4	4
	WBC	9	4
	Glass	9	7

	PID	8	4
$N_h = \left(\sqrt{1+8n}-1\right)/2$ (6) Kayama [20]	Wine	13	5
	Ionosphere	34	8
	Iris	4	4
	WBC	9	4
	Glass	9	4
	PID	8	3
$N_h = 2 * N_i/3$ (7) Wang [21]	Wine	13	9
	Ionosphere	34	21
	Iris	4	3
	WBC	9	6
	Glass	9	6
	PID	8	8

(6)

(7)

6. PERFOFRMANCE EVALUATION

The MLP classification accuracy and Average Mean Square Error (MSE) are used in the performance analysis of the proposed method. as shown in equations 9,10, 11 and 12[9]:

$$Clasfication\ accuracy = \frac{Number\ of\ samples\ correctly\ classifed}{Total\ number\ of\ Samples} \quad (9)$$

$$Average\ Accuracy = \frac{1}{10}\sum_{i=1}^{10} Clasfication\ accuracy \quad (10)$$

$$MSE_i = \frac{1}{N}\sum_{i=1}^{N}\left(Actual\ output(i) - Predicted\ ouput(i)\right)^2 \quad (11)$$

$$Average\ MSE = \frac{1}{10}\sum_{i=1}^{10} MSE_i \quad (12)$$

The experiments were implemented to assess the effects of using the hidden neuron number estimated by the

proposed method on six datasets and compare the accuracy achieved [14],[12],[16],[15],[17],[20],[21].

7. Results and Discussions

The performance of MLP after being trained with Scaled Conjugate Gradient(SCG) learning algorithm[27] and 6 different datasets is presented. Table 2 shows the suggested number of hidden neuron for the six datasets based on seven previous heuristic methods. Table 3 shows the performance of the MLP for six datasets based on the number of hidden neuron that chosen by seven heuristic methods. The comparison between these methods is discussed from two aspects, i.e. classification accuracy and MSE. In this discussion, it is worth mentioning that the priority of choosing the best method is based on accuracy and MSE.

Table 3: MLP Performance based on seven heuristics methods.

Parameters		**Sheela et.al [14]**	**Ke, J et. al [12]**	**Kanellopoulos et. al [16]**	**Hush [15]**	**Kaastra et. al [17]**	**Kayama et. al [20]**	**Wang [21]**
Wine	**Number of neuron**	**4**	**22**	**39**	**26**	**6**	**5**	**9**
	Training Accuracy	67.65	99.79	99.71	99.78	99.74	99.60	99.96
	Tr.MSE	0.103	0.009	0.016	0.020	0.010	0.008	0.011
	Test accuracy	67.51	99.80	99.79	99.90	99.55	99.39	99.67
	Ts.MSE	0.103	0.025	0.037	0.046	0.044	0.012	0.002
Ionosphere	**Number of neuron**	**4**	**48**	**102**	**68**	**8**	**8**	**21**
	Training Accuracy	89.59	93.17	92.81	92.01	88.45	92.36	92.97
	Tr.MSE	0.019	0.089	0.081	0.007	0.071	.069	0.071
	Test accuracy	97.51	97.97	97.01	97.92	96.97	96.95	97.67
	Ts.MSE	0.026	0.029	0.041	0.033	0.040	0.034	0.028
Iris	**Numberofneuron**	**8**	**5**	**12**	**8**	**4**	**4**	**3**
	Training Accuracy	98.7	97.92	98.75	96.51	98.71	96.17	98.59
	Tr.MSE	0.021	0.030	0.019	0.030	0.022	0.019	0.019
	Test accuracy	96.75	96.65	95.29	97.19	95.29	93.28	95.42
	Ts.MSE	0.051	0.042	0.031	0.020	0.036	0.033	0.042
Wisconsin	**Number of neuron**	4	**12**	**27**	**18**	**7**	4	**6**
	Training Accuracy	96.40	95.97	95.77	95.95	96.23	96.14	96.17
	Tr.MSE	0.030	0.031	0.040	0.031	0.032	0.029	0.034
	Test accuracy	96.14	95.64	94.18	95.87	96.14	95.97	95.70
	Ts.MSE	0.010	0.010	0.019	0.011	0.010	0.010	0.010
Glass	**Numberofneuron**	**4**	**12**	**27**	**18**	**7**	**4**	**6**
	Training Accuracy	86.10	87.01	89.34	91.99	87.28	87.11	85.91
	Tr.MSE	0.038	0.045	0.035	0.030	0.042	0.029	0.039
	Test accuracy	85.18	83.46	85.67	87.15	83.19	86.02	82.27
	Ts.MSE	0.032	0.051	0.039	0.044	0.039	0.037	0.038
Pima	**Numberofneuron**	**5**	**11**	**24**	**16**	**4**	**3**	**8**
	Training Accuracy	79.04	78.99	78.13	78.02	78.99	79.65	79.49
	Tr.MSE	0.148	0.140	0.147	0.142	0.155	0.153	0.159
	Test accuracy	75.21	73.99	72.17	73.99	74.26	75.31	73.67
	Ts.MSE	0.190	0.179	0.201	0.173	0.179	0.188	0.175

7.1Based on wine datasets:

For training phase method in Eq.7 [21] with 9 hidden neurons achieved the best results with 99.96% accuracy, 0.011 MSE. All methods except Eq. 1[14]achieved at least 99.0% accuracy with a tiny difference of less than 0.29. The highest number of hidden neurons was in method in Eq. 4[16] while the lowest was in method in Eq. 1.[14]. Moreover, for testing phase, method in Eq.3[15]achieved the highest accuracy (99.90) and MSE (0.0046), while method in Eq. 1 [14]achieved the lowest accuracy(67.51%) and highest MSE (0.103) and lowest number of neurons ;4 neurons.

7.2Based on Ionosphere dataset:

For the training phase, method in Eq. 1[14] had the highest accuracy (93.17%) and the MSE (0.087). The number of neuron in the hidden layer was 48. Method in Eq. 5[17]had the lowest accuracy (88.45%) and 0.071 for MSE with 8 neurons. The rest of heuristic methods achieved accuracy more than 89.59% with number of hidden neurons varied from 8 to 102.
For testing phase, method in Eq. 2[12]had the highest accuracy (97.97%) and the lowest MSE (0.027) using 48 hidden neurons. Method in Eq. 6[20]had the lowest accuracy (96.95%), MSE (0.034) using 5 neurons.

7.3 Based on the Iris datasets:

Method in Eq. 4[16]achieved the highest accuracy (98.75%) and lowest MSE (0.019) by using 12 hidden neurons for training phase. Method in Eq. 6 [20]had the lowest accuracy(96.17%) and 0.019 for MSE in the training phase with 4 neurons in the hidden layer.
For the testing phase, methods in Eq.3achieved the best accuracy (97.19%) with an MSE of 0.020using 8 neurons in the hidden layer. The lowest performance in the testing phase was obtained using method in Eq. 6[20], i.e. 93.28%, 0.033 and 4 for accuracy, MSE and hidden neurons respectively.

7.4 Based on the Wisconsin Breast Cancer data sets:

In the training phase, methods Eq. 1[14] with 4 neurons had the best performance, i.e. 96.40% for accuracy and 0.0030 for MSE. With 95.77% classification accuracy and 0.040 MSE, heuristic method in Eq. 4 showed the lowest performance achieved foe WBC data set using 27 neurons.
In the testing phase MLP achieved the lowest accuracy and MSE i.e. 94.18% and 0.019 respectively with method in Eq. 4.[16] On the other hand, methods in Eq. 1 and Eq.5 achieved the best results with 96.14% and 0.010 for accuracy and MSE respectively using 4 neurons for Eq.1 and 7 for Eq.2.

7.5 Based on the Glass datasets:

In the training phase, in Eq.3[15]achieved the highest classification accuracy (91.99%) and the lowest MSE (0.030) through using 18 neurons in the hidden layer. On the contrary, method in Eq. 7 [21]achieved the lowest accuracy (85.91%) and 0.039 for MSE using 6 neurons.
In the testing phase, with 18 neurons in the hidden layer achieved through applying Eq.3 [15]the MLP showed the highest results when it was used to classify the Glass dataset:MSE was 0.044 and accuracy was 87.15%.The lowest accuracy in the testing phase came from applying Eq. 7 [21] 82.27% and 0.038 MSE with 6 neurons in the hidden layer.

7.6 Based on Pima Indian Diabetes Data Set:

For the training phase, MLP had the highest classification accuracy (79.65%) and 0.153 MSE after being trained using 3 neurons in. In contrast, the worst accuracy (78.02%) and MSE 0.142.MLP had the highest and worst accuracy after being trained using the number of neurons achieved by applying Eq.6 [20] and Eq.3 [15] respectively.
For the testing phase methods in Eq. 6 [20], achieved the highest performance, i.e. 75.31% accuracy and 0.188 MSE using 3 neurons. Furthermore, Method in Eq. 4[16]had the lowest accuracy (72.17%) and the highest MSE (0.201).
It canbe clearly seen in Table 3 that the number of neurons chosen by method in Eq. 1 [14]was the lowest among all other methods and for the six datasets. Followed by methods in Eq.5[17], 6[20], 7[21], 2[12], 4[16] and Eq.3[15].

8. CONCLUSION

Different methods were used to find the optimal number of hidden neurons in the Multi layer neural network. Using six different data sets, the performance of seven heuristic methods in terms of accuracy and MSE were examined. This study discovered the number of the neurons in the hidden layer can be beneficial for MLP neural network performance. It can be concluded that number of neurons in the hidden layer is varied, and depend on different factors such as the data size, attributes and type. In addition, the number of neurons in MLP neural networks hidden layer can't be determined using as a specific method. This study will help the researchers to uncover the critical areas of finding the number of the neurons in hidden layer in MLP neural network that many researchers were not able to explore.

REFERENCES

1. Alkhasawneh, M.S., **Hybrid Cascade Forward Neural Network with Elman Neural Network for Disease Prediction.** *Arabian Journal for Science and Engineering, 2019.*
2. Chunekar, V.N. and H.P. **Ambulgekar. Approach of Neural Network to Diagnose Breast Cancer on three different Data Set.** *in Advances in Recent Technologies in Communication and Computing, 2009. ARTCom '09. International Conference on. 2009.*
3. Bishop, C.M., **Neural Networks for Pattern Recognition.** *Oxford University Press. 1995.*
4. Barletta M. and A. Grisario, **An application of neural network solutions to laser assisted paint stripping process of hybrid epoxy-polyester coating on aluminum substrates.** *Surface and Coatings Technology 200 (24), 6678–6689. 2006.*

5. Abdul-Kadir, N.A., et al. **Applications of cascade-forward neural networks for nasal, lateral and trill arabic phonemes.***in Information Science and Digital Content Technology (ICIDT), 2012 8th International Conference on. 2012.*
6. Mutasem Shabeb Alkhasawneh, **Olive Oil Ripping Time Prediction Model based on Image Processing and Neural Network.***International Journal of Advanced Computer Science and Applications(IJACSA), 2021. **12**(1, 2021).*
7. Cybenko, G., **Approximation by superpositions of a sigmoidal function.** *Mathematics of Control, Signals and Systems, 1989. **2**(4): p. 303-314.*
8. Funahashi, K.-I., **On the approximate realization of continuous mappings by neural networks.***Neural Networks, 1989. **2**(3): p. 183-192.*
9. Alkhasawneh, M.S. and L.T. Tay, A **Hybrid Intelligent System Integrating the Cascade Forward Neural Network with Elman Neural *Network.****Arabian Journal for Science and Engineering, 2018. **43**(12): p. 6737-6749.*
10. Bajes Zeyad, A., A. Mutasem Shabib, and B. Mohammad Ali, *Isolated Arabic Hand Written Letters Recognition Based on Contour Matching and Neural Network.* Journal of Computer Science, 2018. **14**(11).
11. Wang, C., ***Theory of generalization in learning machines with neural application*** **PhD Thesis,***The University of Pennsylvania, USA., 1994.*
12. Ke, J. and X. Liu. **Empirical Analysis of Optimal Hidden Neurons in Neural Network Modeling for Stock Prediction.** *in 2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application. 2008.*
13. Elman, J.L., **Finding structure in time.***Cognitive science, 1990. **14**(2): p. 179-211.*
14. Sheela, K.G. and S.N. Deepa, **Review on Methods to Fix Number of Hidden Neurons in Neural Networks.** *Mathematical Problems in Engineering, 2013. **2013**: p. 11.*
15. Hush. **Classification with neural networks: a performance analysis.** *in IEEE 1989 International Conference on Systems Engineering. 1989.*
16. Kanellopoulos, I. and G.G. Wilkinson, **Strategies and best practice for neural network image classification.** *International Journal of Remote Sensing, 1997. **18**(4): p. 711-725.*
17. Kaastra, I. and M. Boyd, **Designing a neural network for forecasting financial and economic time series.** *Neurocomputing, 1996. 10(3): p. 215-236.*
18. Jin-Yan, L., T.W.S. Chow, and Y. Ying-Lin. **The estimation theory and optimization algorithm for the number of hidden units in the higher-order feedforward neural network.***in Neural Networks, 1995. Proceedings., IEEE International Conference on. 1995.*
19. Kayama, M., **Constructing optimal neural networks by linear regression analysis.** *Neura-Nimes'90, pp. 363-373, 1990, 1990. Neura-Nimes'9 363-373, 1990: p. 363-373.*
20. Kayama, M., **Constructing optimal neural networks by linear regression analysis.** Neura-Nimes'90, , 1990: p. 363-373,.
21. Wang, C., **Theory of generalization in learning machines with neural application***PhD Thesis, . The University of Pennsylvania, USA., 1994.*
22. https://medium.**com/coinmonks/***the-mathematics-of-neural-network-60a112dd3e05.*
23. Mitra, P., S. Mitra, and S.K. Pal, **Staging of cervical cancer with soft computing.***Biomedical Engineering, IEEE Transactions on, 2000. **47**(7): p. 934-940.*
24. Rosenblatt, F., **Principles of neurodynamics. perceptrons and the theory of brain mechanisms.** *1961, Cornell Aeronautical Lab Inc Buffalo NY.*
25. Al-Batah, M.S., et al., **Modified Recursive Least Squares algorithm to train the Hybrid Multilayered Perceptron (HMLP) network.***Applied Soft Computing, 2010. **10**(1): p. 236-244.*
26. Mat Isa, N.A. and W.M.F.W. Mamat, **Clustered-Hybrid Multilayer Perceptron network for pattern recognition application.***Applied Soft Computing, 2011. **11**(1): p. 1457-1466.*
27. Møller, M.F., **A scaled conjugate gradient algorithm for fast supervised learning.** *Neural networks, 1993. **6**(4): p. 525-533.*