Australian Journal of Science and Technology

ISSN Number (2208-6404)
Volume 5; Issue 4; December 2021

Original Article

Comparative study of back propagation and radial basis function neural network on efficiency prediction of smart lighting

Md. Abul Kalam Azad*

Department of Physics, Jahangirnagar University, Dhaka, Bangladesh

ABSTRACT

Artificial neural network (ANN) is functionally analogous to the human brain. Earlier, due to its innate capability of performing highly complex calculations, researchers began to develop an interest in designing a computer model which can work in a way similar to the human brain. Under this study, we have applied the two commonly known techniques of ANN back propagation and radial basis function neural network. The motive of this work is to outline the working of these two approaches for accurate prediction of efficiency and advantages of using these methods over the other ANN techniques. GUI software has been created in MATLAB for easy prediction.

Keywords: Artificial neural network, efficiency, error, function, lighting, smart

Submitted: 26-11-2021, **Accepted:** 06-12-2021, **Published:** 30-12-2021

INTRODUCTION

Artificial neural network (ANN) is a computing model which is functionally similar to the human brain. The first step into the innovation of ANN came into the fact when in 1943, Mc Culloch and Pitts introduced the concept of neurons that work in a way similar to the human brains.[1] In 1961, Frank Rosenblatt introduced the concept of back propagation but was unsuccessful in his attempt.[2] In 1986, Rummerlhart, Hinton and Williams popularize the algorithm.[3] Since then the Back Propagation is a widely known technique applied mainly in the field such as classification, function approximation, and forecasting. Broomhead and Lowe first introduced radial basis function neural network (RBFNN) method in 1988.[4] It consists of three types of layers input, hidden, and an output layer. This algorithm is conceptually similar to k-nearest neighbor algorithm.[5] ANN possesses very absolute power of computational intelligence. Few of the ANN method include back propagation neural network (BPNN),[6] RBFNN,[7] counter propagation neural networks,[8] and Kohonen networks[8] and these are widely used for data classification, pattern recognition, and function approximation. Traditional neural network has to undergo rigorous training before being applied. Training is done directly where the network inputs are applied, and the network weights are adjusted iteratively according to the values of the error. The most popular form of traditional neural network is multilayer perceptron network.[8] RBFNNs have a fixed three layered architecture consisting of input, hidden, and an output layer. Inputs are taken from an input layer, hidden layer maps the input data to make it linearly separable, and an output layer provides the linear separation.

Objective

The study's objective was to compare the back propagation and radial basis function network on efficiency prediction of smart lighting.

RESEARCH METHODOLOGY

BPNN is one of the most powerful algorithms of ANN, which is applied for minimizing the error through a multilayer network. It was first introduced by Rummelhart and MCelland in 1986.

Address for correspondence: Md. Abul Kalam Azad, Department of Physics, Jahangirnagar University, Dhaka, Bangladesh. E-mail: azad58@eub.edu.bd

BPNN algorithm employs a Delta rule[9] for calculating the error in the output layer. The error is propagated backward into the previous layer at each subsequent iteration (Figures 1 and 2).

RBFNNs are the simple class of functions. These can be applied to any linear or non-linear data. The learning technique of this method involves both supervised and unsupervised learning methods. Broomhead and Lowe first introduced this technique in 1988.[10] The RBFNN method is fixed three-layered structure. The RBFNN method is similar to working as K-nearest neighbor algorithm. The Figure 3 shows the structure of RBFNN network. Furthermore, the flowchart of the RBFNN algorithm is shown in Figure 4:

Following steps are used to achieve the desired objective:
- Input data are collected from an agent based modeling and simulation[11]
- Data preprocessing and normalization of data are done to prepare the data in a way to make it suitable for training using ANN methods. Data preprocessing is done to identify any missing data and remove any non-numeric value from the data. Data normalization is done to make the data uniform and in a smaller range of values
- Divide the data in the ratio of training and testing

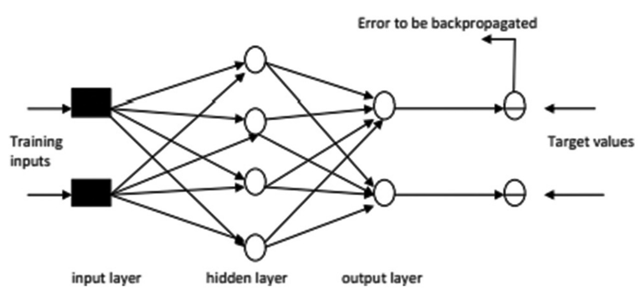

Figure 1: Error propagation through back propagation neural network[6]

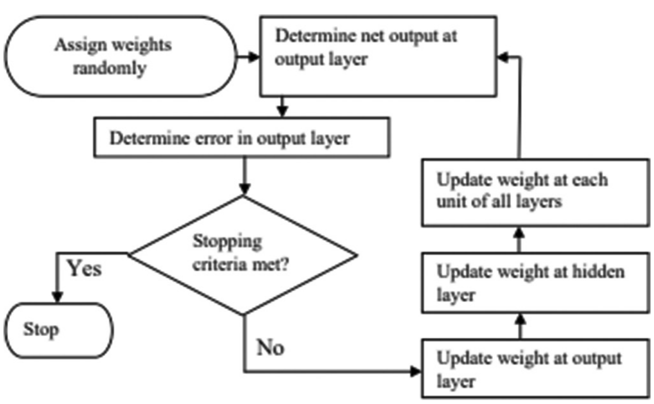

Figure 2: Flowchart of back propagation neural network algorithm[6]

- Create the BPNN and RBFNN networks using the inbuilt functions available in MATLAB
- Predict the future values using BPNN and RBFNN techniques based on the training and testing results
- Compute the accuracy and mean square error (MSE) values
- Compare and analyze the above results.

Implementation

We have used following steps in designing and programming BPNN and RBFNN model in MATLAB. These steps are briefly discussed below:

Collection of data

Collected data were plotted in Figure 5.

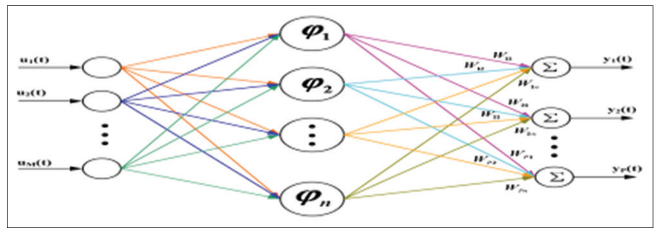

Figure 3: Structure of radial basis function neural network neural network[10]

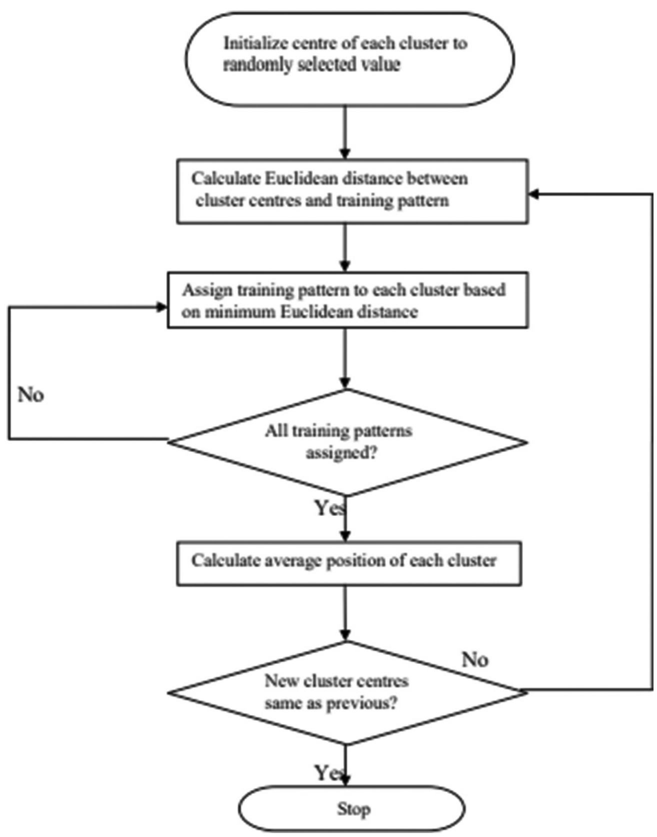

Figure 4: Flowchart of radial basis function neural network algorithm[7]

Data preprocessing and normalization

Data processing is done because of the three reasons: To find the missing data, to normalize the data, and to randomize the data. Normalization is done to make the data uniform.

Network creation

During this stage, the programmer specifies the type of network, number of hidden neurons, training method, learning method, number of epochs, and gradient descend of the network. We have used the BPNN and RBFNN functionalities of the neural network to do the same.

Network training

During the training process, weight and bias values are adjusted to make predicted results (output) close to the actual values. The figure of the training process using BPNN algorithm is shown in Figure 6:

Test the model

The data were divided into the ratio of 70% (training) and 30% (testing). So, the next step is to test the data using an unseen test data.

Experimental Outcomes

In this section, experimental outcomes of the research work are outlined. How the values of MSE and accuracy were estimated and also demonstrated here. Following are the criterion for measuring performance of back propagation and RBFN network in efficiency prediction:

MSE

MSE measures the square of the errors o deviations, that is, the difference between actual and the predicted results. Equation used to calculate MSE is shown below:[12]

$$MSE = \frac{1}{n}\sum_{j=1}^{n} (z' - z)\,\hat{}\,2$$

Where z and z' represent actual and predicted data.

Training means square error

This is an error which occurs when the model is applied to the training data. In our research work, we have used 70% of data as the training data.

Testing MSE

This is an error which occurs when the model is tested on the unseen dataset. This error determines how well a model can perform on the future unseen data.

Accuracy

Accuracy is usually the performance parameter used to measure the quality of a prediction model. It is the ratio of a total number

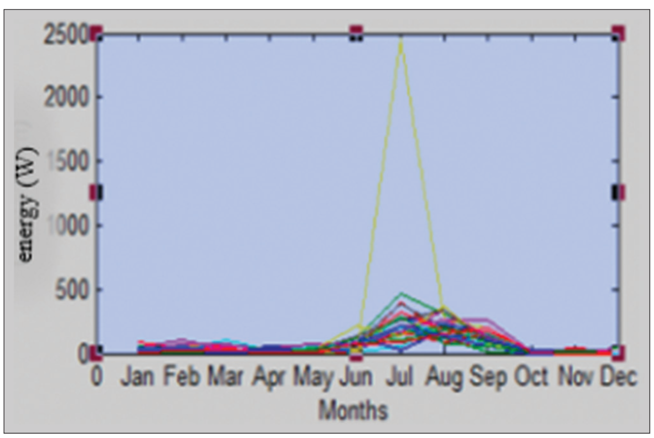

Figure 5: Plot of real data

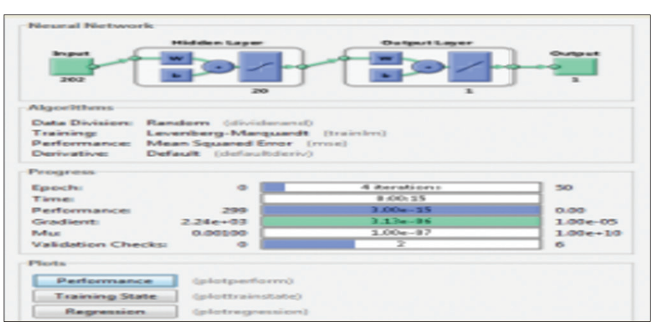

Figure 6: Training using back propagation neural network network

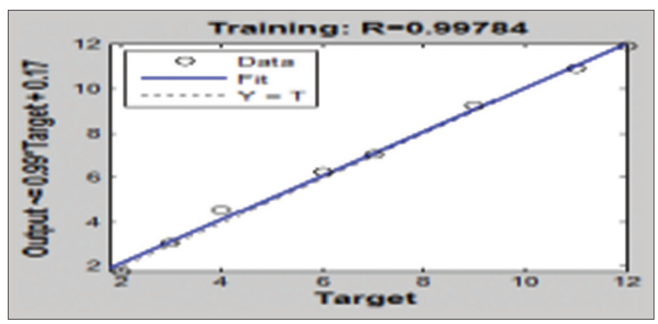

Figure 7: Plot of regression using back propagation neural network[11]

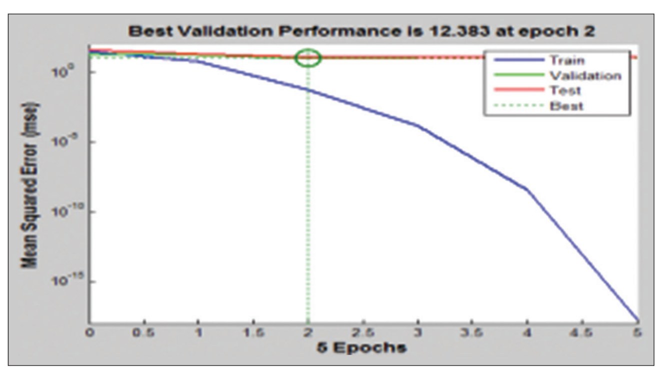

Figure 8: Plot of performance using back propagation neural network[11]

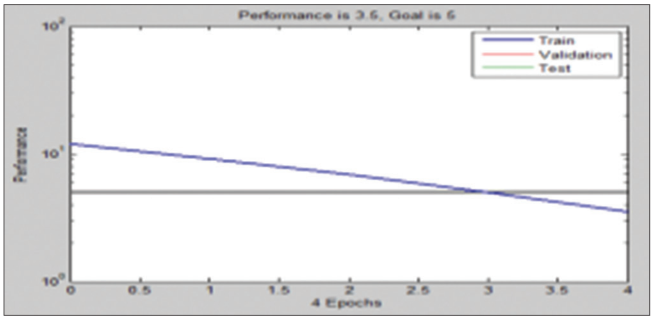

Figure 9: Plot of performance using radial basis function neural network[11]

Table 1: Performance results

	Training Error	Testing Error	Accuracy
RBFNN	0.003172	0.013761	67.30
BPNN	0.024478	0.015659	63.50

of correct predictions to the total cases being evaluated. The equation used for the same is shown below: [12]

$$Accuracy = \frac{(Actual - Forecast)}{Actual} \times 100$$

Based on the experiment, it was found that the RBFNN model outperformed the BPNN model regarding its higher accuracy value of 67.30% and lower MSE (testing) value of 0.013761. Table 1 shows the performance results:

The plot of regression (Figure 7) shown that how close are the output values to the actual target values with the coefficient of linear regression of value 0.99784. The performance plot shows how many minimum numbers of iterations or epochs are required to achieve the best performance results. The result of the performance graph for the BPNN (Figure 8) training indicates that the best validation performance is 12.383 and is achieved at epoch 2. The result of the performance graph for the RBFNN (Figure 9) training indicates that the best validation performance is 3.6 and is achieved at epoch 6. The training models are efficient in prediction as the curves of testing and validation overlap in the performance graph of both the algorithm.

CONCLUSION

The result of accuracy using RBFNN and BPNN is found to be 67.30 and 65.30 which shows that RBFNN is a better model regarding higher accuracy values. Therefore, by analyzing the overall results, it is concluded that the RBFNN is a better model than BPNN model for the prediction of efficiency of smart lighting. As the RBFNN proved to be a better model than the BPNN model for efficiency prediction, therefore, the future work is suggested to use BPNN algorithm with some other techniques such as Fuzzy logic, which is a rule-based mechanism and can help in improving the accuracy rate of BPNN.

REFERENCES

1. McCulloch WS, Pitts W. Logical calculus of the ideas immanent in nervous activity. Bull Math Viophys 1943;5:115-33.
2. Moor J. The dartmouth college artificial intelligence conference: The next fifty years. Artif Intell Mag 2006;27:87-91.
3. Reklaitis GV, Tsirukis AG, Tenorio MF. Generalized hopfield networks and nonlinear optimization. In: Advances in Neural Information Processing Systems. Cambridge: The MIT Press; 1989. p. 355-62.
4. Schwenker F, Kestler HA, Palm G. Three learning phases for radial-basis-function networks. Neural Netw 2001;14:439-58.
5. Lee Y. Handwritten digit recognition using K nearesr neighbor, radial-basis function and backpropagation neural networks. Neural Comput 1991;3:440-9.
6. Dase RK, Pawar DD. Application of artificial neural networks for stock market predictions: Areview of literature. Int J Mach Intell 2010;2:14-7.
7. Plahl C, Kozielski M, Scluter R, Ney H. Feature Combinatin and Stacking of Recurrent and Non-Recurrent Neural Networks for LVCSR. IEEE International Conference on Acoustics, Speech and Signal Processing; 2013. p. 6714-8.
8. Xie T, Yu H, Wilamowski B. Comparison between Traditional Neural Networks and Radial Basis Function Networks. Poland: IEEE International Symposium on Industrial Electronics; 2011. p. 1194-9.
9. Jain AK, Mao J, Mohiuddin KM. Artificial neural networks: A tutorial. Computer 1996;29:31-44.
10. Broomhead DS, Lowe D. Multivariate functional interpolation and adaptive networks. Complex Syst 1988;2:321-55.
11. Epstein JM. Generative Social Science: Studies on Agent-Based Computational Modeling. Princeton: Princeton University Press; 2006.
12. Sundaram RM, Dhara BC. Neural Network Based Iris Recognition System Using Haralick Features. Vol. 3. 3rd International Conference on Electronics Computer Technology; 2011. p. 19-23.