May – June 2020
ISSN: 0193-4120 Page No. 5461-5466

Variation Analysis of Hidden Neuron Incitements on Neural Network Performance

Sumaya Sanober[1], K. Usha Rani[2]
[1]Research Scholar, SPMVV, Tirupati, India, sumayacsdept@gmail.com
[2]Professor, Dept. of Computer Science, SPMVV, Tirupati, India

Article Info
Volume 83
Page Number: 5461 - 5466
Publication Issue:
May - June 2020

Abstract

The modern computational process is used to design and develop an integrated solution as per industrial and social revolution. The mathematical approaches are used to design computing models to construct an optimum solution according to the nature and complexity of the problem. In general, computational problems are solved using iterative and optimization techniques. The research work represents Neural Network as an approach to compute optimum solution in addition to traditional mathematical algorithms. It's a layer based approach to compute the desired output via calculating and predicting weight variations in the hidden neurons. The hidden neuron functions are dynamic in accordance to fatal errors of the Neural Network. These functions are identical one with another to produce desired results. The number of iterations to achieve the desired result differs based on sigmoid function and weight values at the hidden layers. As per the Neural Network architecture, the desired result depends on the hidden layer neurons cumulative functional values and its variation with actual output. These weight values highly influence the desired results and determine the iterations. Therefore, hidden neuronsas well as its variations are determining the number of iterations. This study is aimed to evaluate the co-relation between hidden neurons and iterations, variation weights of hidden neuron and overall performance of Neural Network to produce the desired output. The experimental results exhibit the relationship between incremental or decremented nodes of hidden neurons and its impact on number of iterations to achieve the desired results.

Article History
Article Received: *19 November 2019*
Revised: *27 January 2020*
Accepted: *24 February 2020*
Publication: *17 May 2020*

Keywords: *Neural Network architecture, Neural Network, Hidden neurons, Iteration process*

I. INTRODUCTION

The modern computational approached are applied to solve social and scientific problems. The algorithmic approaches are adopted for optimization solutions [8]. The neural network model is one among the model to solve optimization problem based on weight values. Neural networks have been broadly used to solve the optimization problem (K. T. Sun and H. C. Fu, 93) as per the nature of the problem to find optimum values as well as predict the values based on training. Neural network approaches are applied all optimization and data analytical issues.

Research work aimed to analysis Artificial Neural Network (ANN) structural design via identifying variations among NN process and performance while varying neurons in the various stages of the network [2].Architectural changes and the performance variations are embedded together. This study analysis the neural network performance and its variations while computing the fatal error and its termination [7].

II. ARTIFICIAL NEURAL NETWORK (ANN)

In general ANN is outlined equally as computational method inclusive with a collection of extremely interrelated dispensation component, referred as "Neurons" that develop data and reply to outer incitements. ANN represents the emulation of the indication assimilation and threshold notice performance of biological neurons by using mathematical calculations [3, 17]. Similarly as biological neuron, ANN neurons exist and organized with one another through networks which regulate the stream of info among neurons. Incitements are communicated starting with single dispensation

May – June 2020
ISSN: 0193-4120 Page No. 5461-5466

component with other "synapse" which can be excitatory that transmit an excitatory signal or inhibitory that propagate inhibitory [1].

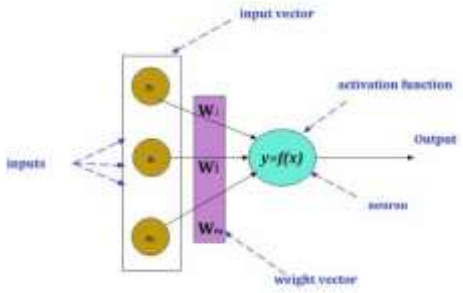

Fig 1. Computational process of neural network

An artificial network implements in learning and testing simulations. The training process emphasis on illustrations offered to the network and it is a procedure aimed to assign weights allied to all the interrelated neurons[5] .On the other hand the training process network 'estimates' the output for respective illustrations. While at the training phase system adjust within the network to obtain the desired output. The learning phase is obtained by 2 ways in Neural Network [6].

Supervised learning is accomplished by input and output sets. In-order to educate the network and recognize a specified input associated through preferred output. Aimed at every instance, network matches the actual and desired output by rectifying the error difference and adjusting the weights to the network in order to obtain the preferred output [4]. Whereas on the other side the Unsupervised learning is used for the network that trained by input signals. Consequently, the network composes inside with specific stimuli or collection related stimuli to create outputs that are reliable [9].

In the above two mentioned phases when the network achieved the preferred performance the learning phase terminates and the weights are termed to be stationary by stabilizing to organize the invisible inputs [10]. While in the testing phase, the grid obtains an input signal to produce an output. The network termed to be simplify, if the network has appropriately educate, and the actual output generated by the grid need to be similar as learning phase for similar inputs.

Neural networks are classically organized into layer based system. Each layer in feed-forward system can be described as a handling components or neurons [15]. Data moves through every component in an input and output way. Subsequently, every component obtains a flag, operates it and move ahead to an outcome indication to the next associated component accompanying nearby levels. The Multilayer Perceptron (MLP) shows three layers of handling components with individual hidden layer, without any limitation on the number of hidden layers [11]. Therefore, input layer obtain the outer incentives and transmit to the following level.

The hidden layer obtains the biased summation of inward signals focused by the input units (Equ. 1), formerly equates by using method of an activation function. These functions frequently are the inundation (Equ. 4), sigmoid (Equ. 5) and hyperbolic tangent (Equ. 6) functions. A hidden component directs an out coming signal to the neurons for the following successive level. Subsequently, components of output layer obtain the biased summation of inward signals and execute the situation with the help of an activation function. Data is broadcast and moves unless the network generates an output.

III. RESEARCH PROBLEM

As per the review of various ongoing research works related to neural network architecture and its computation show that neural network architecture neuron weights are computed and adjusted according to sigmoid function to produce optimum desired results[12]. The optimum values are obtained based on hidden neuron weighted values and its functional process towards output neurons. The number of hidden neurons and its weighted value adjustments are aids to train the networks as well as to determine the termination of the network training to declare the desired results. The number of neurons, error value and iterations represents the effectiveness of neural network to produce efficient results [14]. The research work is aimed to evaluate the performance and its variation established on total no of hidden neurons, error value and its iterations. It also enlightens the correlation among no of neurons and desired results, hidden neurons

May – June 2020
ISSN: 0193-4120 Page No. 5461-5466

with iterations and overall performance of NN training to obtain optimum result according to the nature of the problem.

The following objectives are derived to ensure the analysis of neural network performance and its variations to solve the above stated research problem.

1. Designing and developing of Artificial neural network(ANN) architecture to evaluate ,performance of hidden neuron process influences on neural network iterations
2. Variation analysis of hidden neuron optimum value to compute for Mean square error (MSE)
3. Analysis of computational variations on number of incoming, hidden and outgoing neurons to produce optimum results.

Methodology: An algorithmic model is designed with following stages. These steps are providing systematic construction of neural network for computing optimum solution.

Stage I: Initially regularize the input and output values from the highest one. Consider with every training duos a normalized form that there are ℓinputs given by {I} I and ℓx 1.N outputs given by {O} on x 1

Stage II: Take up the neurons at hidden layers from 1 < m < 9.

Stage III: Here [V] signifiessynapses load among input, hidden neuron and [W] signifies synapses load among hidden, output neuron. Subsequently, regulate loads of synapses with minimalarbitrary values typically ranging between -1 to +1;

Stage IV: In order to providetraining, contribute the pattern as inputs layer {I} Iby utilizinglinear stimulation, the outcome of the input layer is assessed.

$$\{O\}_I = \{I\}\ I \quad (t \times 1 \quad t \times 1)$$

Stage V: Calculate the inputs associated to hidden

$$\{I\}_H = [V]\ T\{O\}\ I \quad (m \times 1 \quad m \times \ell \quad \ell \times 1)$$

Stage VI: calculate output, by sigmoid function

$$\{O\}o = \frac{1}{(1 + e^{-(IHi)})}$$

m x 1

Stage VII: Calculate the inputs associated to outputs by multiplying parallelloads of synapses

$$\{I\}\ o = [W]\ T\{O\}_H \quad (n \times 1 \quad n \times mm \times 1)$$

Stage VIII: Estimate the output by sigmoid function:

$$\{O\}o = \frac{1}{(1 + e^{-(I)}O_J)}$$

(Network output)

Stage IX: Compute inaccuracy using the variance among the grid outcome with the predicted outcome of j[th] training

$$E^p = \frac{\sqrt{\sum(Tj - Ooj)^2}}{n}$$

layers by multiplying parallel loads of synapses **Stage X:** Calculate a term

$$d = (T_k - O_{ok})\ O_{ok}\ (1 - O_{ok}) \quad n \times 1$$

Stage XI: Fig 2.

$$[Y] = \{O\}_H \{d\} \quad (m \times n \quad m \times 1 \quad 1 \times n)$$

Stage XII: Compute

$$[\Delta W]^{t+1} = \alpha [\Delta W]^t + \eta [Y] \quad (m \times n \quad m \times n \quad m \times n)$$

Stage XIII: Compute

$$\{e\} = [W]\{d\} \quad (m \times 1\ m \times n\ n \times 1)$$

$$\{\square *\} = \frac{(\square\ \square\ \square)(1-}{\square\ \square\ \square\ \square} \quad m \times 1\ m \times 1$$

Published by: The Mattingley Publishing Co., Inc.

May – June 2020
ISSN: 0193-4120 Page No. 5461-5466

Compute

$$X = \{ O \}_I \{d^* \} = \{ I \}_I \{ d^* \}$$
$$1 \times m \quad \ell \times 1 \; 1 \times m \ell \times 1 \; 1 \times m$$

Stage XIV: Compute

$$[\Delta V]^{t+1} = \alpha [\Delta V]^t + \eta [X]$$
$$1 \times m \qquad 1 \times m \qquad 1 \times m$$

Stage XV: Compute

$$[V]^{t+1} = [V]^t + [\Delta V]^{t+1}$$
$$[W]^{t+1} = [W]^t + [\Delta W]^{t+1}$$

Stage XVI: Compute rate of "Error"

$$Error\ rate = \sum \frac{Ep}{n}$$

Stage XVII: Replicationof stages from IV to XVIcould conjugate error value and diminishes tolerance rate.

IV. ANN ARCHITECTURE

Back-propagation: Back-propagation (BP) broadly aimed at training algorithm to apply with in the networks in which preferred outputs are obtained; even the outputs of the intermediary layers are not present. This approach is known as supervised learning [16]. The use of Back- propagation algorithm is to reduce SSE by computing the steep descent (gradient) .Gradient descent rule also known as delta rule is used in SSE at the learning phase and extraordinary technique applied to accomplished both binary as well as continuous neurons [13]. It is consequently aimed to decrease the error in the output of the neural network through gradient descent.

$$E = \sum Ei \; i = 1\; 2 \sum (di - oi)2 \; i \quad (2)$$

$$\partial E \; \partial wj = -\sum (di - (\sum wjxj \;\;)).f'(\sum wjxj \; i \; j).xj \; i \; i \quad (3)$$

To obtain the weight update rule, the error descent rule $(\Delta wj = -\propto \;\;)$ is used and the change that is occurred in the weights can be calculated in Eq. (4) as follows:

$$\Delta wj = \alpha$$
$$\partial E \; \partial wj / \; xj \; i \quad (4)$$

First, E, with respect to the weights, is found and then weights should be updated as shown in Eq. (5).

$$\Delta wij = -\propto \quad (5)$$

V. RESULTS AND DISCUSSION

Here the dataset with four input neuron and 2 output neuron was considered to show the experiment result. The result shows the various iterations and hidden neurons in order to achieve the desired output neurons.

Table 1: With 4:2 networks

Input Neuron	Hidden Neuron	Output Neuron	Iterations
4	3	2	167
4	4	2	185
4	5	2	221
4	6	2	245
4	7	2	269

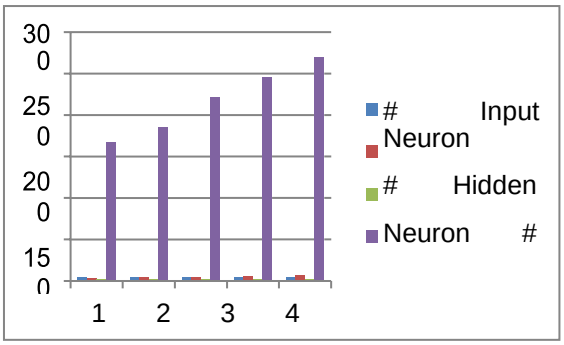

Fig 3. Represents iterations ofInput, Hidden and Output neurons

VI. CONCLUSION

This study analyses the performance of neural network feed-forwarded hidden layer process along with its iteration. The study proved that the number of iterations is increased while hidden neurons are more on the way to obtain optimum result with negotiable threshold value; although hidden neurons are influencing the iteration in-order to produce optimum outcomes. It's because of combination of computation and integrations of neuron process to obtain the optimum result. As per the study, if the numbers of hidden neurons are more than each neuron active function contributions are considered

May – June 2020
ISSN: 0193-4120 Page No. 5461-5466

to compute the optimum results. Therefore the occurrence of computation to eliminate the error is increased the iterations. Correspondingly, the number of hidden neuron is less the iterations are reduced it directly influences the computational time and the memory which is handles to process iterative computations. This study results shows that the hidden neurons variation is termed to be a significant factor to regulate artificial neural network performance. Present study outcomes could be further examined with different data set of different domains.

REFERENCES

[1] Carney J.G., Cunningham P., The NeuralBAG algorithm: Optimizing generalization performance in bagged neural networks, to be presented at 7th European Symposium on Artificial Neural Networks, Bruges (Belgium), 21-23

[2] K.Banu and N.Rama, "Study on Influence of Cognitive Load for Software Developer's Performance using BPNN Algorithm", International Journal of Computer Applications, Vol. 102, No.13, 2014, pp:1-5

[3] K. GnanaSheela and S. N. Deepa, "Review on Methods to Fix Number of Hidden Neurons in Neural Networks", Mathematical Problems in Engineering, 2013, pp.1-11,

[4] Sordo, Margarita, "Introduction to neural networks in healthcare", Open Clinical knowledge management for medical care, 2002, pp.1-17

[5] D. Hunter, H. Yu, M. S. Pukish, J. Kolbusz, and B. M. Wilamowski, "Selection of proper neural network sizes and architectures: a comparative study", IEEE Transactions on Industrial Informatics, vol. 8, no. 2, pp. 228–240, 2012.

[6] C. A. Doukim, J. A. Dargham, and A. Chekima, "Finding the number of hidden neurons for an MLP neural network using coarse to fine search technique", International Conference on Information Sciences, Signal Processing and their Applications, 2010, pp. 606–609.

[7] G. Panchal, A. Ganatra, Y. P. Kosta, and D. Panchal, "Behaviour analysis of multilayer perceptrons with multiple hidden neurons and hidden layers", International Journal of Computer Theory and Engineering, Vol. 3, no. 2, 2011, pp. 332–337.

[8] R. Devi, B. S. Rani, and V. Prakash, "Role of hidden neurons in an elman recurrent neural network in classification of cavitation signals," International Journal of Computer Applications, vol. 37, no. 7, 2012, pp. 9–13.

[9] Madhiarasan, M., and S. N. Deepa, "A novel criterion to select hidden neuron numbers in improved back propagation networks for wind speed forecasting", Applied intelligence, vol. 44, no. 4, 2016, pp. 878-893.

[10] Ramachandran, S., P. Aravind, and S. RathnaPrabha, "Model Identification and Validation for a Nonlinear Process using Recurrent Neural Networks", *International Journal of Research in Electronics and Computer Engineering, vol.* 2, no. 4, 2014, pp. 13-18.

[11] Keeni, Kanad, Kenji Nakayama, and Hiroshi Shimodaira, "Estimation of initial weights and hidden units for fast learning of multilayer neural networks for pattern classification", International Conference *Neural Networks*, vol. 3, pp. 1652-1656. 1999.

[12] Sun, Jianye, "Learning algorithm and hidden node selection scheme for local coupled feed forward neural network classifier", *Neuro computing*, Vol.79, 2012, pp. 158-163.

[13] Madhiarasan, M., and S. N. Deepa, "New Criteria for Estimating the Hidden Layer Neuron Numbers for Recursive Radial Basis Function Networks and Its Application in Wind Speed Forecasting.", *Asian Journal of Information Technology, Vol.*15, no. 21, 2016, pp. 4377-4391.

[14] Hunter, David, Hao Yu, Michael S. Pukish, JanuszKolbusz, and Bogdan M. Wilamowski, "Selection of proper neural network sizes and architectures—A comparative study", *IEEE Transactions on Industrial Informatics,* Vol.8, no. 2, 2012, pp. 228-240.

[15] Sheela, K. Gnana, and Subramaniam N. Deepa, "Review on methods to fix number of hidden neurons in neural networks", *Mathematical Problems in Engineering*, 2013.

[16] Qasim, Mohammed, and VinodKhadkikar, "Application of artificial neural networks for shunt active power filter control, *IEEE Transactions on*

May – June 2020
ISSN: 0193-4120 Page No. 5461-5466

industrial informatics, Vol. 10, no. 3, 2014, pp. 1765-1774.

[17] Arya, Sabha Raj, and Bhim Singh, "Neural network based conductance estimation control algorithm for shunt compensation", *IEEE Transactions on Industrial Informatics, Vol.* 10, no. 1, 2014, pp. 569-577.