Understanding Neural Networks for Machine Learning using Microsoft Neural Network Algorithm

Nagesh Ramprasad
UG student ,
ECE Department
RV College Of Engineering,
Bangalore, India

ABSTRACT
A Neural Network refers to a simple computing system that consists of some highly fixed and interconnected processing elements. All the neural networks appear as a set of layers. The layers of the neural networks have some interconnected nodes that make up the activation function. In this research, focus is on the Microsoft Neural System Algorithm. The Microsoft Neural System Algorithm is a simple implementation of the adaptable and popular neural networks that are used in the machine learning. The algorithm operates through the testing of every possible state of the inputs that attribute against all the possible states of some predictable calculating of probabilities for all the combinations on the training data. For instance, the neural networks can be used to write a computer program that will recognize some handwritten digits. Later in the study the artificial neural networks are discussed. In the artificial neural networks, no multiple central processors exist. Most of them consist of the learning rules that are used for the modification of weight that connect them to the input patterns that present them continuously.

From the study, it is clear that the Microsoft neural network viewers can be used to work and see with the data models that correlate with the outputs and the inputs. Through this utilization a custom viewer can then explore the model structure of a Microsoft neural network quickly. The neural networks can be used to recognize some handwritten digits. In this case consider a sequence that is written as 505762. Several individuals will basically recognize the digits as 505762. Such an ease might be deceptive. In all the hemispheres of the human brain, people have the primary visual cortex that is also known as the v1. The v1 contains 140 million neurons that have 10 billion connections between them. On the other hand the human vision has a series of visual cortices of v2, v3, v4 and v5 that help in recognizing more difficult image processing. Therefore our heads are used as the supercomputers. It is not easy to recognize the handwritten digits. Humans have therefore come up with the neural networks to help in recognizing handwritten digits so that they can make sense to what is being done unconsciously. In this case, the neural networks help in sampling data, distributing and invoking it into simple values.

General Terms
Neural Networks, Machine Learning, Artificial Intelligence

Keywords
Neural Networks, Microsoft Neural Networks, handwritten digits, data models, Artificial Neural Networks

1. INTRODUCTION
A Neural Network system is a modest computing system that consists of some highly fixed and inter-connected processing elements. The items have the ability to process information through an active state of reaction to the external inputs. A neural setup can also be classified as an Artificial Neural Network system (ANN). The Artificial Neural Systems are the main processing services that are either algorithms or the actual hardware. Most of them are modeled loosely after the structure of a neuronal mammalian cerebral cortex but has a rather smaller scale. A larger Artificial Neural Network consists of more than a hundred or a thousand processor units. On the other hand, the mammalian brain is made up of billions of neurons which have an increasingly equal magnitude that helps in the interaction of some emerging behavior. The Artificial Neural Network researchers are however not concerned with their networks accurately since most of them resemble the biological systems. Some of the researchers have simulated the functions of the human retina accurately and have been able to model the eye perfectly. Most of the mathematics associated with the neural interacting is not an inconsequential situation. The user can gain at least a unique operative considerate of the function and structure of the neural networking. In a mathematical application, the network has some 784 neurons in the input layers. The corresponding neural network has 28*28=784 pixels that are in the input images. One can use around 30 hidden neurons that involve of about 10 outputs of neurons that respond to about 10 possible classifications of a MNIST digits (‘0’, ‘1’, ‘2’, ‘3’… ‘9’). Trying to train on the neuron networks has some complete 30 epochs that are used as the mini-batches of about 10 training examples. Neural networks have a learning rate that can be approximated as n=0.1 and a regulation parameter of about x=0.5. One can train the accuracy of the validation that can classify the accuracy of about 96.48 percent.

2. BASIC CONCEPTS OF NEURAL NETWORKS
All the neural networks are classically set up in layers. The layers consist of several interconnected nodes that are made up of the instigation function. Some shapes are therefore accessible to the neural linkage through an input cover that is used for communication to a single or more unseen layers where the physical processing is undertaken through the major process of some weighted networks. The output layer is connected through linking it up to the hidden layer. The answer found is output.

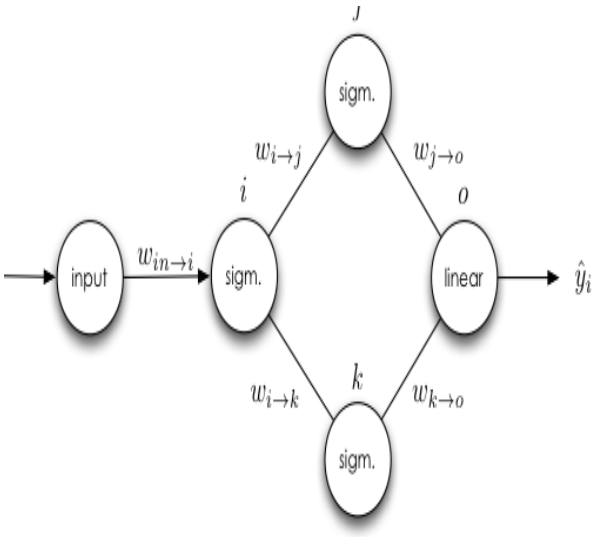


Fig 1: Output Mathematical model

From figure 1, the mathematical case shows the hidden unit of some more output units. This shows that there is a type of weight that derives a W unit derivative. The differences in the output cases of I units are a much more immediate succession that sums up the errors that are accumulated along the unit rates of I units. This derives some weight updates that define the sigma tracks.

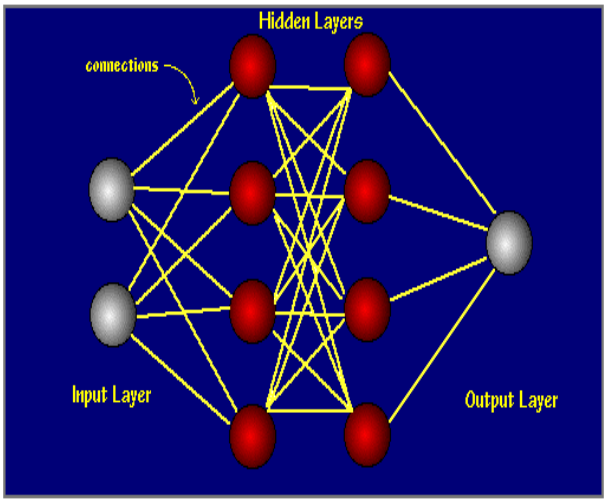


Fig 2: Artificial neural network figure

From Figure 2, it is clear that most of the Artificial Neural Networks consist of some forms of learning rules that are used to modify the weight of connecting them to the input patterns that are used to present them continuously. In a more critical sense, the Artificial Neural Networks then learn through examples more so in doing their biological matching part. For instance, a kid studies how to identify a dog from other samples of dogs. Some of the neural networks use several kinds of learning rules that are different. In this situation, the delta rule can be used to find out how this demonstration works [2]. First, the delta rule utilizes a more conventional class of the Artificial Neural Networks that is referred to as the Back propagation Neural Networks. The term Back-propagation can be abbreviated as the backward propagation of errors. In using the delta rule, one can apply to other forms of the back- propagation where erudition is a supervised procedure that only follows only when the linkage is presented with some new input patterns [1]. The cycle, therefore, moves through a forward flow that involves the activation of several flows of other outputs. The process also includes the backward process of propagation of errors that can easily be adjusted through their weights. In a simpler manner, the neural networks have the ability to be presented with their initial patterns through making some random guesses regarding what might be happening. The neural network has the final ability to see the farther end of where the answer appears more so from the actual sense that makes some more appropriate adjustments through the connection weights. Below is the Presentation of the process of change in the weight relationships:

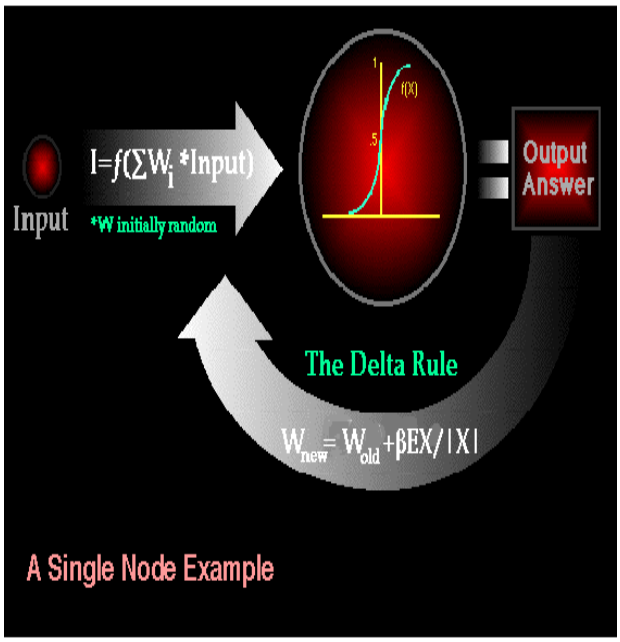


Fig 3: Node Example

The sigmoidal activation function within the hidden layer as shown in Figure 3, enables the polarization of the network activities and also helps in stabilizing the systems. The Back-propagations are also used to perform some gradient descents that are within the solution vectors in the space that is directly opposite the minimum global direction on the steeper vectors of the surface error. The theoretical solution used within the lower possible error is the minimum global space. The surface error is then used as a hyper -paraboloid since it has smooth surfaces that can be used as the solution areas [5]. The solution space appears as an irregular shape that contains several pits and hills that might result in the down settlement of the network more so in the local minimum. This, however, does not appear to be the best solution for the system irregularity.

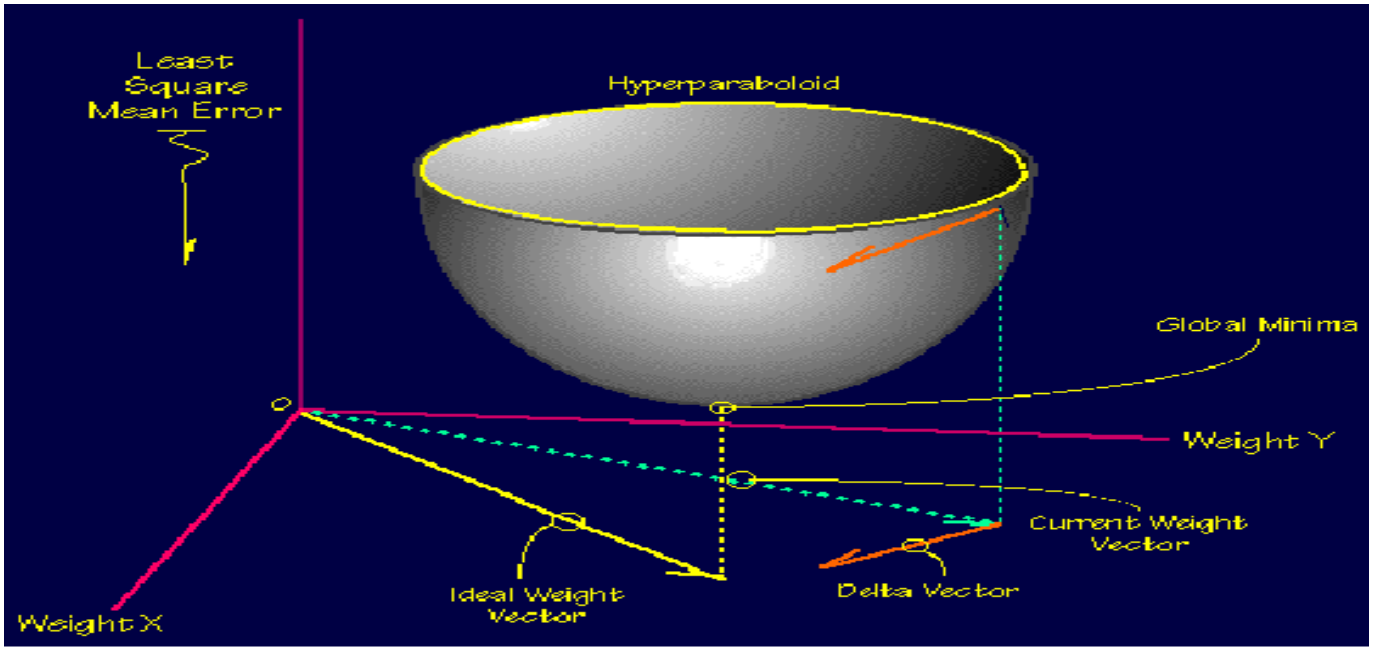


Fig 4: Natural Neural Network

The natural surroundings of the error space as shown in Figure 4, is not recognized as a preceding. Additionally, the neural setup analysis requires some larger numbers where individuals can learn so as to define the best explanation. Most of the learning guidelines have been used to build some complex mathematical terms that help in the process of controlling the coefficient swiftness and the learning thrust. The knowledge speed is more important in the convergence of the current solutions and in maintaining the global rate. A neural setup is proficient satisfactorily to a close that is used as an investigative device on the learning figures [8]. The user has to specify the training that can allow the networks to work on the forward propagation of the mode only. The new input is then presented to the input patterns that are later filtered into the middle layer where it is processed in the area where the training takes place. The output information is however reserved and at that situation, none of the back-propagation will occur. The production of the forward dissemination is then run in the projected model where the records are then cast-off for further investigation of the interpretation.

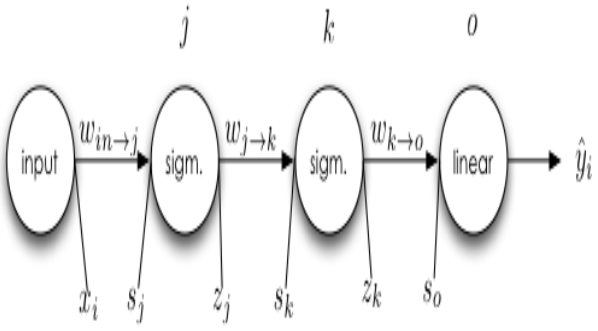


Fig 5: Mathematical Representation

The unit in Figure 5, can be used as a unit indices to define the values of a signal error in the general form of a back-propagation. In a simple case considering only 1 neuron with a single neuron before its appearance, the single neuron can form a value T that represents the entire threshold. The one charges the neuron for instance C it might get larger than T. If C>Y then C will fire up and charge one of the neurons.

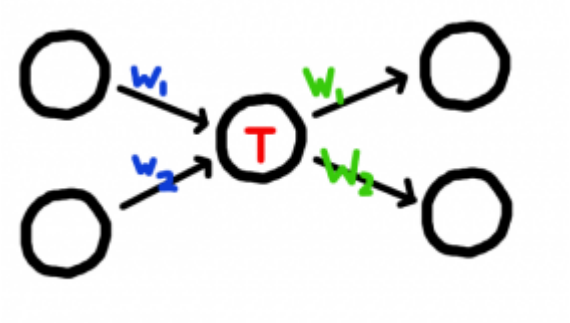


Fig 6: Mathematical Representation

In the above layer of Figure 6, 3 neurons have a value of 2, 3 and 2. In this case the middle neuron has an input that varies in the following sequence: C=Winput1+Winput2. C is >Y and it will fire up an amount of Woutput1 and Woutput2. Consider the equations below:

F1*W1, 1+ F2*W2, 1=C1

F1*W1, 2+ F2*W2, 2=C2

F1*W1, 3+ F2*W2, 3=C3

From the above case, F changes from 0 or 1 due to the firing up of the previous neuron. W and M represent the weight of the previous neurons while n represents the next neuron M.

Over-training of the neural network is possible since most of the system has purely been skilled to reply to a particular sort of input. The information used is like a role of memorization. If such a situation occurs then, the learning will not occur any further, and the linkage is then mentioned to as being the grand mothered network in the neural net jargon[6].Such a situation is useful in the real world since one has to separate the original grand mothered network from the other kinds of inputs.

The neural networks differ from conventional computing through the following computing process of information. The serial computers have some central processors that are used to address the arrays of memory where the data location is used in storing the given instructions. The computations are then made by the processors through the reading of instructions and through the use of data instructions that requires the input of more memory that is used in addressing the executed results that are saved in a more specified memory location just as it is necessary[7]. In the following systems, the computing process requires some logical and sequential determinists. The given variables can easily be tracked from a single location to another.

The Artificial Neural Networks are not necessarily or sequentially determinable. In an ANN, no multiple central processor exists. Simple central processors usually exist that do not necessarily need a weighted sum of input from any other processor. The ANN do not typically contain any program instructions. Most of them respond in a parallel state through a pattern of data that are presented together in a systematic format [9]. Separate memories are also arranged together on it that are used for the storage of data. The information found in the network states contains essential knowledge that presents itself through some channels that are present in the individual components. In this research, the Microsoft Neural System Algorithm was handled.

3. MICROSOFT NEURAL SYSTEM ALGORITHM

The Microsoft Neural System algorithm refers to an implementation of some adaptable and widespread neural networks that appear as an architecture for machine learning[3]. The algorithm works through a process of testing every possible state of different inputs that are attributed against any possible format of a predictable attribute. Some occur through the course of calculating probabilities for every combination that is based on the training of data [10]. The usage of this kind of expectation in regression of tasks and in classifying complex data is easy. One can also predict an outcome by basing on the input attributes. Some analysts use the neural network in the process of association analysis. Creating mining models help to use the Microsoft Neural Network algorithms. One can use multiple outputs to create some several networks. Some of the primary systems that exist in the single mining models contain some depended number of states in the input columns. The predictable columns are used in modeling of the state columns. For instance, the Microsoft neural network algorithm can be utilized in the analysis of some complex data that begins from the commercial to manufacturing process [11]. The business problems are also a substantial training magnitude of data that is available in the guidelines that can never be derived easily through the use of algorithms. The Microsoft Neural Network algorithms can occur through various suggestions.

First, the promotion and marketing analysis suggests that the success of measuring direct emails helps to promote the radio advertising campaigns. Secondly, the scenarios can also contribute to predicting the stock movement; it can also assist in the fluctuation of currency and in moving the highly fluid finances and information from the historical data. All the suggested circumstances can also be used in the process of industrial and manufacturing analyses. It is necessary that one can use Microsoft numeral networks in text mining[12]. Finally, the Microsoft Neural networks can be used in predicting any model that analyzes the complex relationships that are in between the inputs that are relatively fewer than the outputs.

3.1 Working of the Microsoft Neural System Algorithm

A scientist can use the Microsoft Neural Network algorithm in the creation of a network that consists of some three layers of neurons[4]. Most of these layers include some input layers, the output layers, and the hidden layers. The information layers can be identified as the attributes that have values that are used in model mining and in determining probability. The hidden layers can be determined as the hidden nodes that usually receive the inputs that originate from the input nodes and those that provide some output to the output nodes. All the hidden layers contain various probabilities that have some assigned weights. The masses provide some relevant, essential particulars that are at the input hidden nodes. The greatest weight is allocated to a data that has some more values that are the input level. The negative weights have some inputs that can inhibit the concrete results and favor. The output layers have some nodes that represent the predictable attributes of values that are used in the model mining of data.

3.2 Data Essential for the Neural Network Models

The neural network models contain some major columns which have one or extra contribution columns. Some contain single or more expectable set pillars. The withdrawal of the data simulations can use the Microsoft neural system algorithm to influence the parameter values heavily. The parameters can be used in defining the sampled data quickly[7]. The statistics are then disseminated or even anticipated to be circulated to every pillar where the selection of features is later invoked to some limited values which can be used in making the final models. The multilayer perception of the neural networks helps in receiving one or more inputs that are used to produce some identical outputs. Every sample of the production is used to test some non-linear functions which sum up all data to the visible neurons.

The data later pass forward from the nodes to the input layers where the nodes are reflected the hidden layers. The information is then transmitted to the hidden layers. No connections exist between the neuron layers. The logistic regression model then moves forward directly to the nodes that are at the input layers and those that are at the output layers. A neural link comprises of three major types of neurons which are created within the Microsoft neural network algorithm. First, there is the input neuron which provides the input attributes with the facts removal models. For the distinct input features, the feedback neuron can classically represent a particular state of neuron coming after the data quality. Missing values might be experienced when the training data has some null information in the input neurons. Some of the discrete data and attributes that have several generating states contain one input for every state of an input neuron.

If any null exists in the training data, then some continuous input attributes lead to the generation of two different kinds of input neurons where each neuron has a missing state of an input attribute. Two inputs of neurons have some missing states that have a single neuron to the worth of a continuous quality by the aforementioned. The participation neurons can contribute some data in one or more hidden neurons. Secondly, there is the hidden neuron that is used in the process of receiving some input neurons that leads to the

provision of outputs to production neurons. The output neurons provide some predictable attributed values that are used in data mining models. The discrete inputs and attributes together with the output neurons provide some typically represented and single state of some predictable characteristics. The attributes have some missing values. Some of the missing values in the predictable binary attributes have some productions in the output nodes that can be easily described as the missing states that are in the existing state.

Sometimes the associated double features to indicate the values that are in the predictable characteristics[9]. The Boolean columns lead to the generation of several output neurons where each neuron is used as an actual value. The neurons are used for receiving the input information that originates from other kinds of neurons. The information depends on some layers that come from the neural networks. The input neurons can be used to receive some input information more so from the original information. The output and hidden neurons help to receive some input information coming from the output neurons. The information gotten goes further to the neural networks where the input information establishes some relationships between the neurons. The relationship established at the neurons goes further to serve as the primary path of analyzing the specific sets of data cases. Every input in the assigned values acts as a weight that can be used to represent some relevant information. The greatest weight allocated to the inputs serves as the relevant primary value coming from the data. The original weights used can either be harmful or used to imply the inhabitation of the inputs rather than activating it on the stated neurons.

3.3 Training of the Neural Networks

Some of the key steps used in the training of the neural networks[4] include the training of data mining model and data sourcing. The main percentage of data training is referred to as the holdout data since it is reserved for use in the accurate network algorithm parameter. In the training process, some channels are employed in the evaluation of some next iteration that is found in the training data[2]. The training process then stops when the accuracy does not exist further. Most values exist through sampled sizes and are later used in the holding out of the percentage parameters that are then used in determining the number of instances that ascertain the sample of working out data and also the quantity of circumstances that can be put separately of the holdout figures. The primary values belonging to the held out seeded parameters are used to determine most individual cases randomly in cases where data cannot be put aside.

The algorithm also determines some complex number of networks that are used to mine the model support. If some of the mining models have one or more attributes, then they can use them to predict the algorithm that is created using single networks that are represented through the primary attributes. If the key models can contain one or more attributes, then one can use both models as predictions and inputs in the construction of the algorithm provider and as a linkage for every aspect. We can use the following concept in Figure 7 to handle the linkage of calculating the weighted linkage aspects:

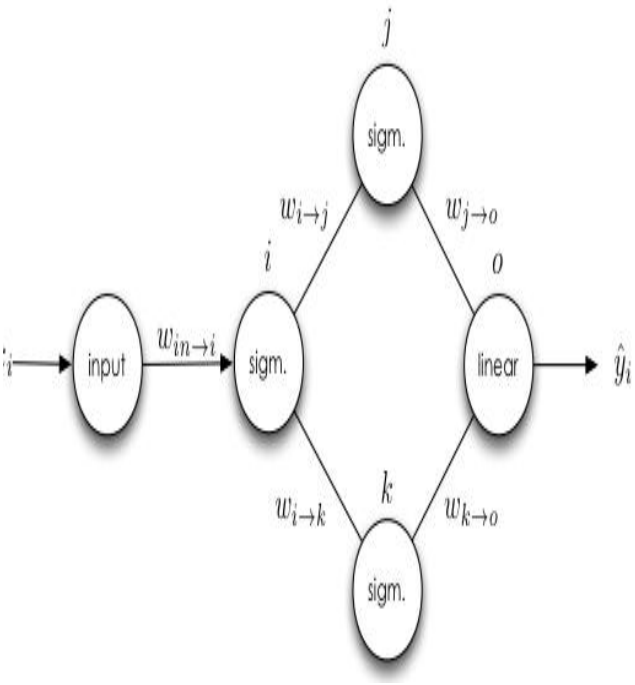
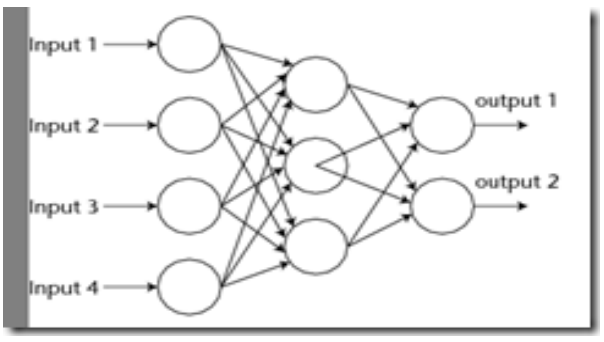


Fig 7: Mathematical Representation of Calculating the Weighted Linkage Aspect

Some of the predictable and input features that contain the unconnected values, every output and contribution neuron can represent respective neuron in a single state. For the data that appear through the predictable attribute, some have some continuous values where each output and input has some individual distribution and range values for the attribution of anticipated. The primary states of data that support the depended cases exist through maximum States of algorithm parameters[10]. The leading traditional relevant state for the chosen attributes exists to a concentrated number of situations that are allowed in the outstanding states of some congregated values that are absent in the purpose of the quality analysis. All algorithms use some hidden node ratio values and parameters in the process of shaping the preliminary amount of neurons that form the concealed primary layers. One can easily set a concealed layer and values so as to determine the original number of the neurons that create some hidden layers of neurons. It is easy to set an unknown node ratio so as to prevent any creation of an opaque layer of the network of an algorithm that generates the primary mining model so as to keep the main neural networks as a logistic regression.



Most researchers use the iterative algorithm providers to evaluate the weight of input that is put across the network of the sets of training data. The actual values can easily be compared to every case of the holdout data. The prediction of systems can be used in the processing of known consignment ignorance. Immediately after the evaluation of a whole set of the working out data, the algorithm evaluations can be used in predicting the actual values for every single neuron used[11]. The calculation of algorithms can be used in evaluating the degree of errors and in adjusting the weights of the neuron inputs that come from the working backward neurons and the input neurons through the process of the back propagation of the output neurons. The algorithms appear as repeated processes that are on a complete set of the working out data[8]. In Figure 8, the algorithms appearance easily supports several loads that are at the output neurons. The main gradient algorithms can be used to guide the training processes more so in evaluating and in assigning the weights of the inputs.

3.4 Viewing of the Neural Networks

The Microsoft Neural Network Viewer can be used to see and work with the data model that correlates with the outputs and the inputs[3]. Through the utilization of the custom viewer, one can, therefore, explore the model structures quickly. It is also possible to filter the attributes of the data and their values. The graphs shown have some significant effects on the yields. The tooltips found in the onlooker account for the likelihood that lifts some association with every pair of indicated input and also the output value. All outputs of a network neutral can be used as an encoded target or as multiple single targets which can be employed in the process of predicting the input information. The models should be used to create a single network, and this can necessarily lead to the normalization of values. Some several attributes cab however be employed in the prediction and input process where the models create some networks that help in normalizing the output. All outputs should always be encoded so that they can exit in a normal direction to the system. The process of encoding can be built on the addition of each distinct value that is used in case of preparation and the process of, multiplying the weight of the costs. Such a procedure is known as the weighted summing. The process is later passed on to the instigation function where the concealed coating is at the z-score, and this can be sent in the encoding process.

The most common type of the artificial neural network is the perceptron. The perceptron is a threshold value that works through a binary input for instance of x1, x2…and leads to the production of a single binary output.

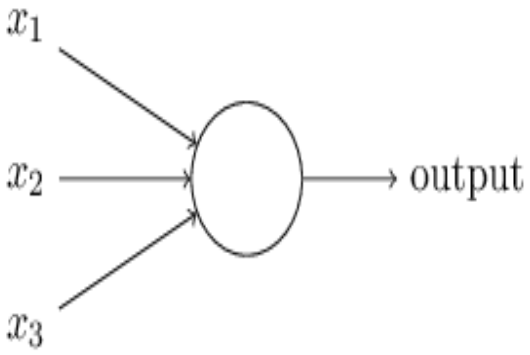


Fig 9: A Representation of a Perceptron

From Figure 9 of the example of the perceptron, there are three main inputs, x1, x2 and x3. Generally, fewer or more inputs could have been viewed. Rosenblatt projected a simpler rule for use in computing the output. He brought up the weights, w1, w2, and extra. He expressed them as the real numbers that can be used in the expression of important respective inputs to outputs. The Neuron output, 0 or 1 can be determined through a weighted sum w1, x1 that is less than or greater that the threshold values. The threshold therefore appears to be a real number that is a parameter of the neurons. This can be viewed as an algebraic term:

$$\text{output} = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{threshold} \\ 1 & \text{if } \sum_j w_j x_j > \text{threshold} \end{cases}$$

Fig 10: Algebraic Term of the Threshold Values

4. CONCLUSION

From the above research, it is obvious that the neural network was designed to calculate the models that are founded on the human brain to solve one's trainers tasks. The artificial neural network for instance solves mathematical problems that are simple while solving using a computer but difficult for an individual. For instance, computing the square root of 964,324 using the line of codes produces 982 as the number value for the square root. That works in less than one second from a computer. Other problems can easily be solved by an individual but appear difficult for the computer. For instance a toddler can identify a kitten easily without using a computer. The neural networks have therefore been designed by the scientist only to help in solving composite solutions.

5. REFERENCES

[1] Bishop, Christopher M. "Pattern recognition." *Machine Learning* 128 (2006).

[2] Funahashi, Ken-Ichi. "On the approximate realization of continuous mappings by neural networks." *Neural networks* 2.3 (1989): 183-192.

[3] Microsoft Neural Network Algorithm, https://msdn.microsoft.com/en-us/library/ms174941.aspx

[4] Microsoft Neural Network Algorithm Technical Reference, https://msdn.microsoft.com/en-us/library/cc645901.aspx

[5] Graves, Alex, Abdel-Rahman Mohamed, and Geoffrey Hinton. "Speech recognition with deep recurrent neural networks." *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013.

[6] Hinton, Geoffrey E., et al. "Improving neural networks by preventing co-adaptation of feature detectors." *arXiv preprint arXiv:1207.0580* (2012).

[7] Hornik, Kurt, Maxwell Stinchcombe, and Halbert White. "Multilayer feedforward networks are universal approximators." *Neural networks* 2.5 (1989): 359-366.

[8] Haykin, Simon, and Neural Network. "A comprehensive foundation." *Neural Networks* 2.2004 (2004).

[9] Maass, Wolfgang. "Networks of spiking neurons: the third generation of neural network models." *Neural networks* 10.9 (1997): 1659-1671.

[10] Oja, Erkki. "Principal components, minor components, and linear neural networks." *Neural networks* 5.6 (1992): 927-935.

[11] Specht, Donald F. "Probabilistic neural networks." *Neural networks* 3.1 (1990): 109-118.

[12] Wolpert, David H. "Stacked generalization." *Neural networks* 5.2 (1992): 241-259.