IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 17, Issue 4 Ser. III (Jul. – Aug. 2021), PP 28-32
www.iosrjournals.org

Artificial Neural Network- A Principle Behind IT

*Mrinalini Smita**
Assistant Teacher , St. Xavier's College,
Ranchi , Jharkhand

Abstract:
The use of neural network architecture in deep learning models is called as Artificial Neural Network (ANN). It is one of the most powerful machine learning algorithms applied to tasks across many domains. (finance, humanities, science. research and academics etc.). An ANN is a form of computation inspired by the structure and function of brain. ***[Padhy, 2005]*** *In this paper, we concentrate on the fundamentals of human neurons and how they are applied to artificial neurons to understand the principles of ANNs.*
Keywords*: Artificial Neural Network (ANN), Neural Network (NN) , artificial neuron.*

Date of Submission: 13-08-2021 Date of Acceptance: 28-08-2021

OBJECTIVE:

- To understand the principle behind ANN.
- To explain the network topology (architecture) of ANN.

I. Introduction:

ANN is inspired by Neural Networks (NN) found in human brain. Neural Network is highly interconnected network of neurons. The human brain has approximately 100 billion neurons. Each neuron may be connected to up to 10,000 other neurons**.[Padhy ,2001]**

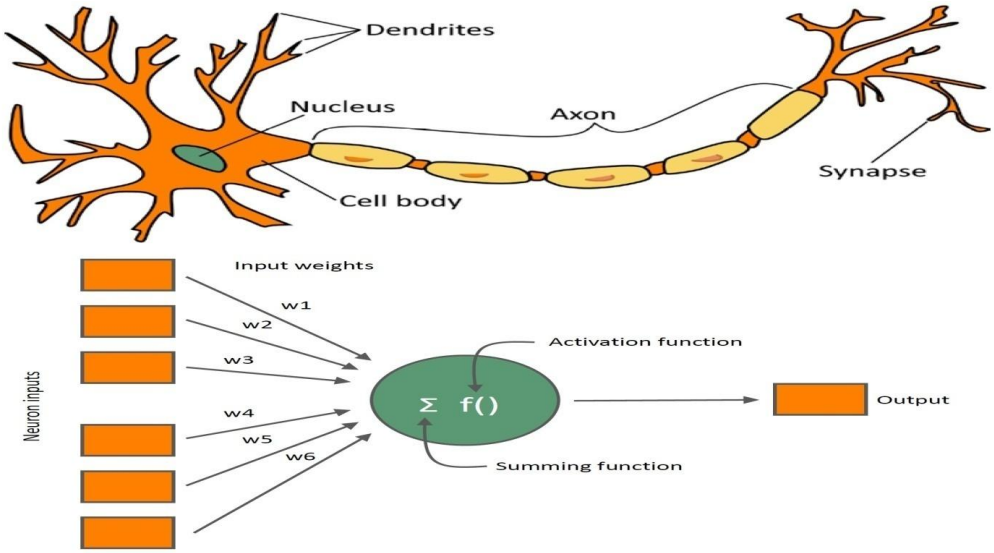

Fig 1. MODEL OF NEURON:

CELL BODY- receive input from other neurons
TENTACLES- each have different thickness as each one of them has different strength. Each of the tentacles – does whatever input it receives. It either amplifies or reduces that input and sends it to the cell body. Cell body receives all inputs (either amplified or reduced), combines them and processes them further. The combination part is done by a function called *Summation function* and further processing is done by a function called activation function and then output is passed onto other neurons.
In the same way , ***ANN is the basic building block***- cell body is called NODE and there are several links for receiving inputs and each of the links has a certain weight .Whatever input we get from other nodes multiplied

with this weight , associated with the link and pass it to the node. Now this summation function combines input from all the nodes, taking weighted sum of all inputs

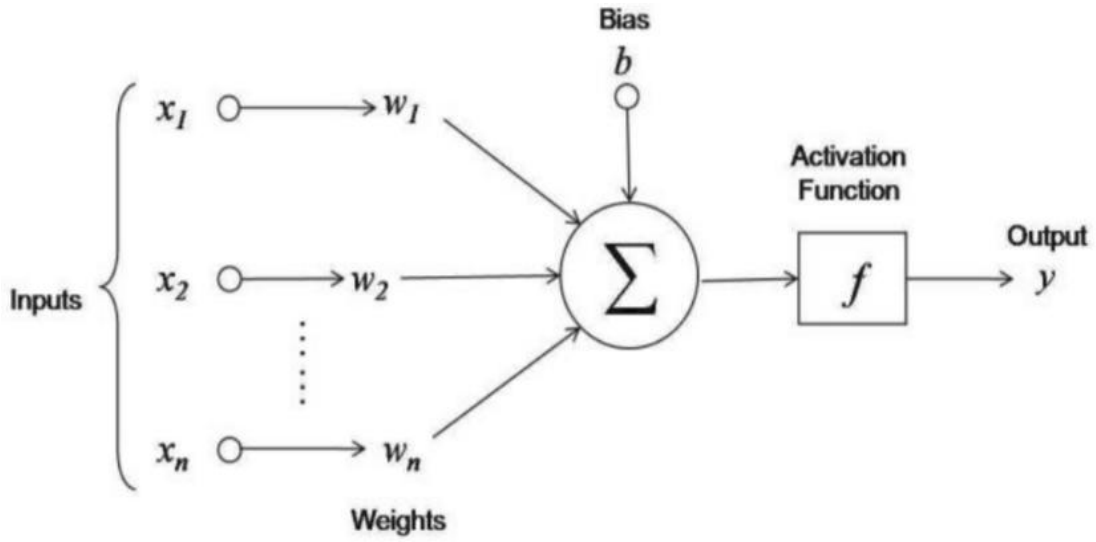

Fig.2 MATHEMATICAL MODEL FOR NEURALNETWORK

and weights are the weights of the node as given

$$X_j = x_1 W_{1,j} + x_2 W_{2,j} + \cdots \ldots + x_m W_{m,j} + b_j$$
$$= \sum_{k=1}^{m} x_k W_{k,j} + b_j$$

Bias term, b_j is added to weighted sum of inputs to the network to make sure even if the input to the network and weight and weight of the links are zero, the input to activation function is non-zero. So each node in hidden layer and output is, has a bias associated with it . Now, activation function takes X_j as input and gives Y_j as output so output of activation function can be written as

$$Y_j = f(X_j)$$

and this output Y_j is passed on to other nodes.

In human brain, these neurons are interconnected , in complex manner. However , in ANN simple arrangement of artificial neurons and simple pattern of connection between neurons is called the **ARCHITECTURE OF ANN.**

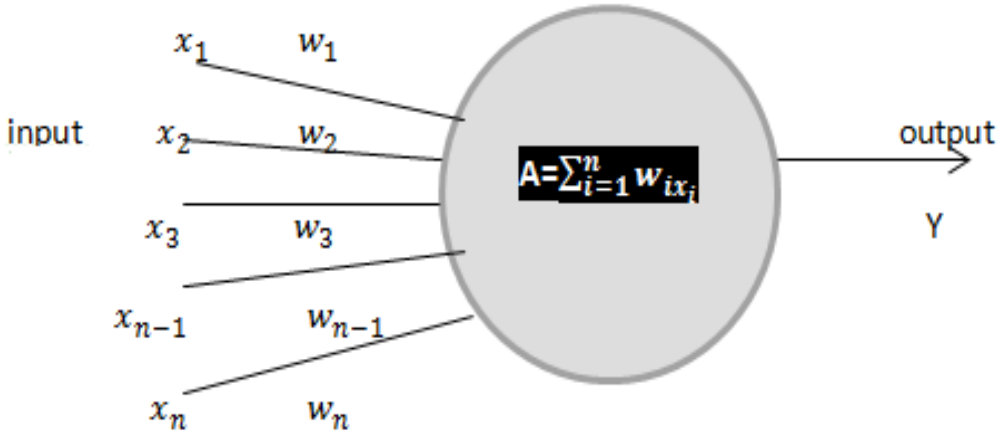

Fig 3: ARCHITECTURE OF NEURALNETWORK

The most common architecture of ANN is Feed Forward ANN. ***Feed Forward network*** is the simplest form of a layered network . The neurons are assigned in layers and these layers are straight one after another. Nodes in a particular are connected only with previous layer and nodes in the next layer. the Feed forward Network as we input the feature vectors X at the input layer(I/L) , they are being processed in every layer and

get output finally and nowhere propagated in the backward direction . ***This is the reason a NN is Feed Forward but the learning is Back Propagation Algorithm.***

Fig.4 FEED FORWARD AND BACKPROPAGATIONALGORITHM

❖ **FEATURES OF NEURAL NETWORK:**

- It is primarily a **BLACK BOX** as inner workings can be difficult to understand. Underlying models are based on complex mathematical systems.
- ANN models the relationship between a set of input signals (set of independent variables) and an output signal (a dependent variable).
- ANNs are versatile learners that can be applied to learning tasks such as classification, prediction and even unsupervised pattern recognition.

❖ **CONCEPT OF ANN:**

- A direct network diagram, shown in fig.3 above , defines relationship between the input signals received (x-variable) and the output signal (y-variable).
- Each signal is weighted- allow each of the n-inputs (x), to contribute a greater or lesser amount to the sum of input signals.
- Activation function f in $y = f(w^T X)$, receives the input signal.
- NNs use neurons building blocks to construct complex models.

❖ **CHARACTERISTICS OF ANN:**

- ***Activation function basically means "Input times weights, add bias and activate***" . *A NN without an activation function is essentially just a linear regression model.* [**Padhy,2005**]. It transforms a neural's net input signal into a single output to be broadcasted further in the network. The activation function does the non-linear transformation to the inputs making it capable to learn and perform more complex tasks.
- ***Network topology*** (***Architecture***) is the arrangement of nodes and pattern of connections. When we have a sequence of functions from input(I/P) to output (O/P), we cannot directly compute the partial derivative of output , $\frac{\partial O}{\partial X}$ where

$$O = f^{(k)}(f^{(k-1)} \dots\dots \left(f^{(i)} \dots \left(f^{(2)}\left(f^{(1)}(X)\right)\right)\right)$$

as output O is far away from the input X. rather we have to make use of chain rule of differentiation to find out the partial derivatives of output (O/P) with respect to input(I/P). ***The partial derivatives is the sensitivity of output on the input X.*** We have sequence of neural network functions from input to output which cannot be directly computed so in back propagation of ANN, we make use of these properties of chain rule of partial derivatives.

- ***Training algorithm- Partition*** is very important in machine learning. It is generally done in two parts - ***training dataset*** **and** ***testing dataset***. The reason behind partition is that most of the ANN techniques are ***black box techniques,*** we do not have minute details of what is exactly going inside the network and as the process is so complex that we cannot control the entire process. The *objective of training* is to adjust the weights so that application of a set of inputs produces the desired output.

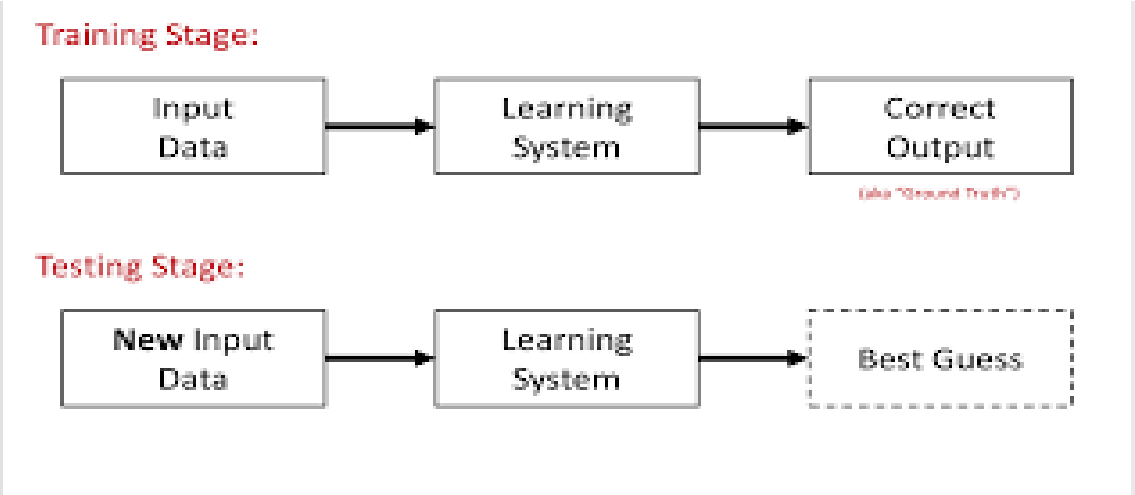

Fig 5. TRAINING AND TESTING PHASE OF ANN

❖ **LEARNING OF ANN:**

ANN learns under supervision from actual or expected output for a given input. So, the learning approach of ANN is called ***supervised learning***. In general , in ANN, we may not be able to predict the exact same output as our actual output. There will always be some error and our objective of learning is to come up with values of weight for links and biases for nodes that will minimize this cumulative error for all the input and output pairs in our training dataset. The function that minimizes the cumulative error for all the input and output pairs is called *cost function* or *loss function* given by

$$\mathcal{L} = \frac{1}{n}\sum_{k=1}^{n}(\hat{y}_k - y_k)^2$$

It can be observed that the value of loss function depends upon input and output pairs (x_k, y_k) , $k = 1,2,\dots\dots,n$ and the weights of the inputs $W_{i,j}$ and biases of the nodes , b_j of the network. Since input and output pairs (x_k, y_k) , $k = 1,2,\dots\dots,n$ are our given training dataset so they do not change . They should be treated as constant during our learning process. That is why , we need to minimize the loss function with respect to the weights of the inputs $W_{i,j}$ and biases of the nodes , b_j of the network and not $x_k or\ y_k$. As the functions involved are not always smooth, numerical approaches are commonly used to minimize the loss function. The most common numerical approach is ***Gradient Descent Approach.***

Gradient Descent means moving downward i.e. opposite to the direction of maximum slope. ***Gradient Descent Algorithm*** is an alternative algorithm to find the minimum of a higher dimensional loss function by moving downward in the direction of maximum change in the function.

WEIGHT UPDATION FORMULA:

$$W_{new} = W_{old} - \eta \frac{\partial \mathcal{L}}{\partial W}$$

Here $\boldsymbol{learning\ rate}, \eta$ is a hyper parameter in the optimization algorithm that determines the step size for iterative solution to find minimum of a loss function. Of course, if η is equal to 0, then no learning will take place. Therefore, η must always be positive.[**Tsoukalas & Uhrig ,1996]** A ***smaller*** *learning rate* results in a more optimal solution but it may take significantly longer time to search that optimal solution. A ***larger*** *learning rate* reduces the training time as compared to smaller learning rate but may miss the point of minimum.

In weight updation , we take ***partial derivatives (the sensitivity of output on the input X)*** of loss function with respect to weights, $\frac{\partial \mathcal{L}}{\partial W}$ which help us to find whether this derivative or slope we are finding is positive or negative. So in updating the weights using ***Gradient Descent Algorithm*** , we start with the last layer and we move one layer back at a time . This idea is also called the back propagation of error as we are updating on the basis of error or loss function and because this update happens in backward direction , this algorithm is known as ***back propagation algorithm***. Therefore, this ANN is called ***Feed Forward Back Propagation*** as input flows in forward direction and the error / updating of weights flows in backward direction.

❖ **MERITS OF ANN:**

- ANNs are non-linear in nature, capable to understand the data pattern and analyze, hence preferred over the traditional linear models.
- One of the most important properties of ANN is that NNs are capable of ***Learning by example*** means utilising examples from the data and organising them into a useful form constitutes a model representing the relationship between input and output variables.**[Padhy, 2005]**
- Stock markets are chaotic .The use of ANNs in a chaotic market does not require an understanding the market dynamics. This is why ***it is practically feasible and profitable to use Machine learning systems like NN to predict the behaviour of financial instrument such as stocks.***

❖ **LIMITATION OFANN**:

Although back propagation of ANN is viewed as powerful model ; it has some of its drawbacks in financial applications which include

- A major disadvantage of neural networks lies in their knowledge representation. Acquired knowledge in the form of network of units connected by weighted links is difficult for humans to interpret. **[Kamber,2001]**
- ***Overfitting*** is one of the limitations of back propagation , which occurs when a model is able to predict on data well when trained but unable to generalise accurately in testing phase .i.e. model has high variance but low bias.

- ***Under fitting*** is the limitation. A model is said to be under fitting when it is unable to classify/ predict means training accuracy of the model is very low i.e. model has low variance and high bias.
- Another drawback is that ***deactivating***, or dropping neurons will reduce the performance of the neural network.

II. Conclusion:

Neural networks have been considered "***black boxes***" and "***data mining tools***".

The artificial neural networks (ANNs) are an effective tool for stock market prediction and can be used on real world datasets. Advocating for the use of ANN will reduce significantly the level of bias and increase the *accuracy of forecasts.* Thus use of back propagation is just a stepping stone in future prediction technologies.

References:

[1]. Padhy, N.P. , "Artificial Intelligence and Intelligent Systems", ISBN-13: 978-0-19-567154-4,400-437,2005
[2]. Han, J., Kamber, M., "Data Mining: Concepts and Techniques" ISBN:1-55860-489-8, 310-311 , 2001
[3]. Tsoukalas, L.H. and Uhring , R.E., " Fuzzy and Neural Approaches in Engineering", John Wiley , New York1996.

*Mrinalini Smita. "Artificial Neural Network- A Principle Behind IT." *IOSR Journal of Mathematics (IOSR-JM)*, 17(4), (2021): pp. 28-32.