NEURAL NETWORKS

Dr. Krishnendu Guha

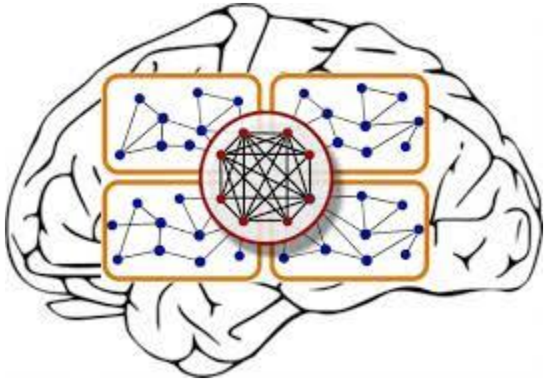
Assistant Professor (On Contract)

National Institute of Technology (NIT), Jamshedpur

Email: krishnendu.ca@nitjsr.ac.in

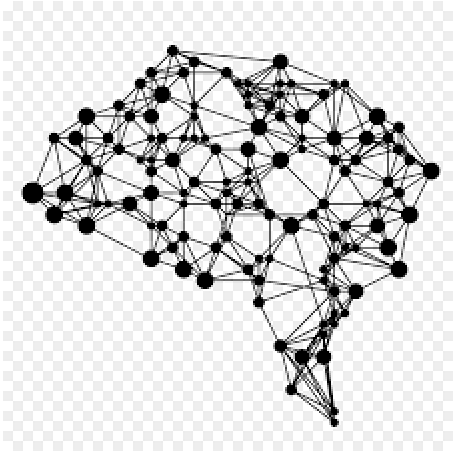
INTRODUCTION

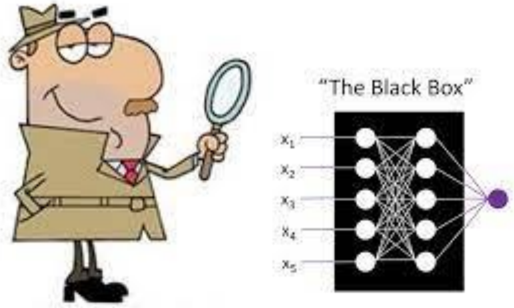
- Neural networks are parallel computing devices
  - which are basically an attempt to make a computer model of the brain.
- The main objective is to develop a system to perform various computational tasks faster than the traditional systems.
- These tasks include
  - pattern recognition and classification,
  - approximation,
  - optimization,
  - data clustering,
  - etc., etc.



ARTIFICIAL NEURAL NETWORK (ANN)

- ANN is an efficient computing system, also named as “artificial neural systems,” or “parallel distributed processing systems,” or “connectionist systems.”
- whose central theme is borrowed from the analogy of biological neural networks.
- ANN acquires a large collection of units (nodes or neurons, are simple processors which operate in parallel) that are interconnected in some pattern to allow communication between the units.
- Every neuron is connected with other neuron through a connection link.
- Each connection link is associated with a weight that has information about the input signal.
- This is the most useful information for neurons to solve a particular problem because the weight usually excites or inhibits the signal that is being communicated.
- Output signals, which are produced after combining the input signals and activation rule, may be sent to other units.





HISTORY
ANN DURING 1940S TO 1960S

- **1943** − It has been assumed that the concept of neural network started with the work of physiologist, Warren McCulloch, and mathematician, Walter Pitts, when in 1943 they modeled a simple neural network using electrical circuits in order to describe how neurons in the brain might work.
- **1949** − Donald Hebb's book, *The Organization of Behavior*, put forth the fact that repeated activation of one neuron by another increases its strength each time they are used.
- **1956** − An associative memory network was introduced by Taylor.
- **1958** − A learning method for McCulloch and Pitts neuron model named Perceptron was invented by Rosenblatt.
- **1960** − Bernard Widrow and Marcian Hoff developed models called "ADALINE" and "MADALINE."

HISTORY
ANN DURING 1960S TO 1980S

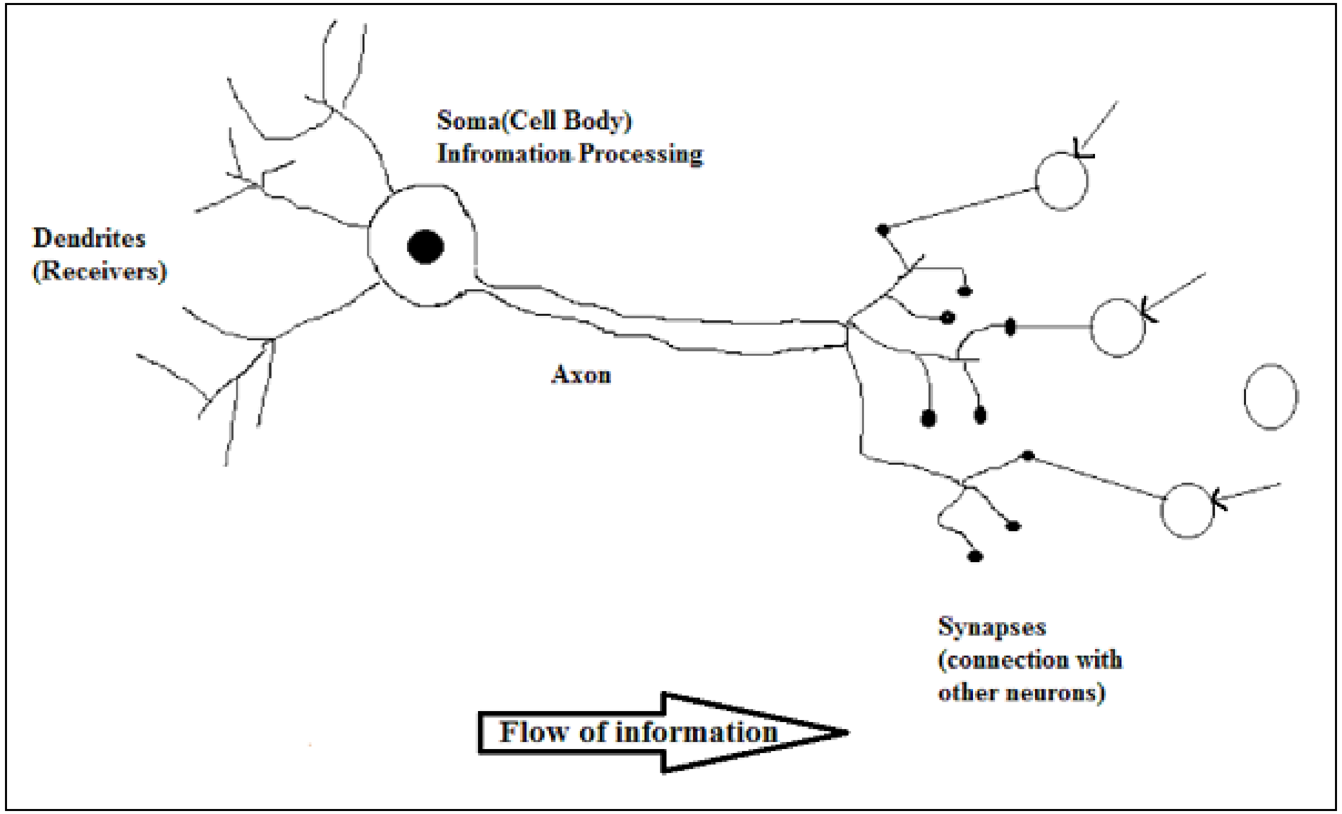
- **1961** − Rosenblatt made an unsuccessful attempt but proposed the “backpropagation” scheme for multilayer networks.
- **1964** − Taylor constructed a winner-take-all circuit with inhibitions among output units.
- **1969** − Multilayer perceptron was invented by Minsky and Papert.
- **1971** − Kohonen developed Associative memories.
- **1976** − Stephen Grossberg and Gail Carpenter developed Adaptive resonance theory.

HISTORY
ANN FROM 1980S TILL PRESENT

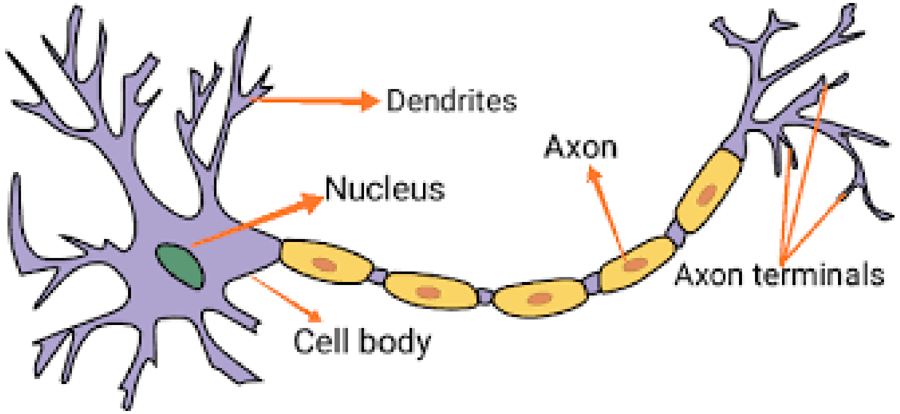
- Some key developments of this era are as follows −
- **1982** − The major development was Hopfield’s Energy approach.
- **1985** − Boltzmann machine was developed by Ackley, Hinton, and Sejnowski.
- **1986** − Rumelhart, Hinton, and Williams introduced Generalised Delta Rule.
- **1988** − Kosko developed Binary Associative Memory and also gave the concept of Fuzzy Logic in ANN.

BIOLOGICAL NEURON

- A nerve cell or neuron is a special biological cell that processes information.
- According to an estimation, there are huge number of neurons, approximately 10^{11} with numerous interconnections, approximately 10^{15}.

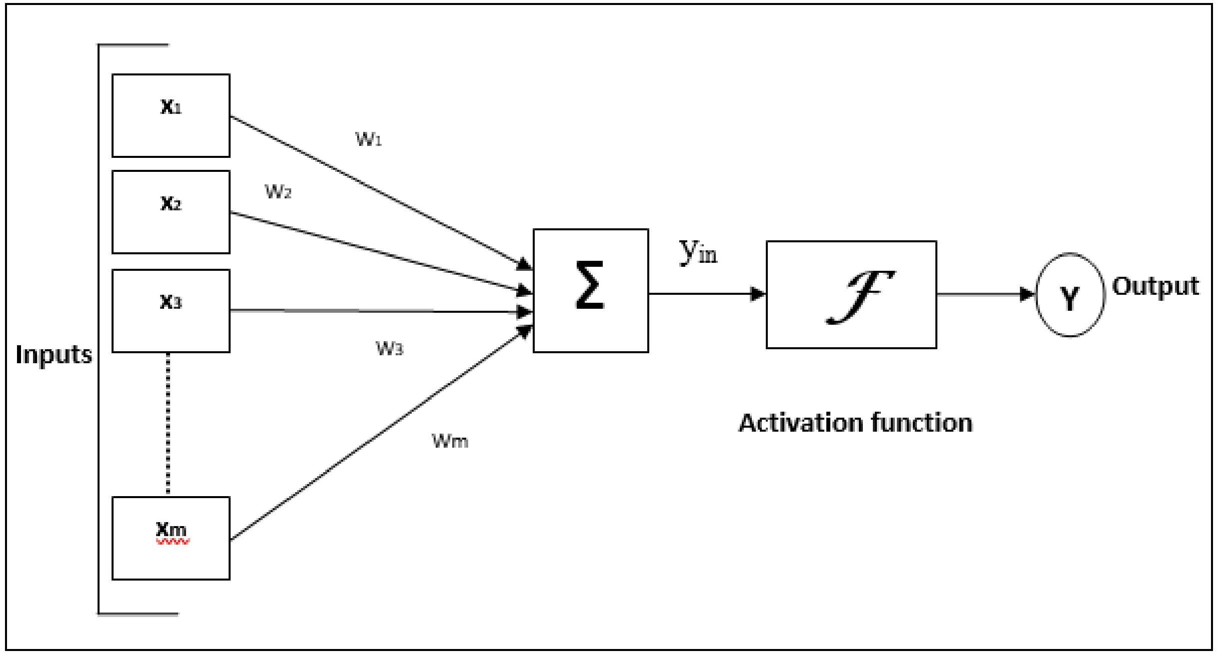


- Working of a Biological Neuron
- As shown in the above diagram, a typical neuron consists of the following four parts with the help of which we can explain its working −
- **Dendrites** − They are tree-like branches, responsible for receiving the information from other neurons it is connected to. In other sense, we can say that they are like the ears of neuron.
- **Soma** − It is the cell body of the neuron and is responsible for processing of information, they have received from dendrites.
- **Axon** − It is just like a cable through which neurons send the information.
- **Synapses** − It is the connection between the axon and other neuron dendrites



MODEL OF ARTIFICIAL NEURAL NETWORK

The following diagram represents the general model of ANN followed by its processing



$$y_{in} = x_1 . w_1 + x_2 . w_2 + x_3 . w_3 \dots x_m . w_m$$

i.e., Net input $y_{in} = \sum_i^m x_i . w_i$

The output can be calculated by applying the activation function over the net input.

$$Y = F(y_{in})$$

Output = function $netinputcalculated$

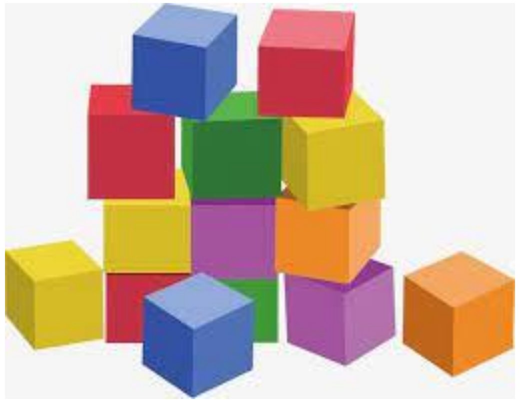
COMPARISON OF ANN AND BNN

Biological Neural Network BNN	Artificial Neural Network ANN
Soma	Node
Dendrites	Input
Synapse	Weights or Interconnections
Axon	Output

Criteria	BNN	ANN
Processing	Massively parallel, slow but superior than ANN	Massively parallel, fast but inferior than BNN
Size	10^{11} neurons and 10^{15} interconnections	10^2 to 10^4 nodes $mainlydependsonthetypeofapplicationandnetworkdesign$
Learning	They can tolerate ambiguity	Very precise, structured and formatted data is required to tolerate ambiguity
Fault tolerance	Performance degrades with even partial damage	It is capable of robust performance, hence has the potential to be fault tolerant
Storage capacity	Stores the information in the synapse	Stores the information in continuous memory locations

ANN BUILDING BLOCKS

- Processing of ANN depends upon the following three building blocks –
- Network Topology
- Adjustments of Weights or Learning
- Activation Functions



NETWORK TOPOLOGY

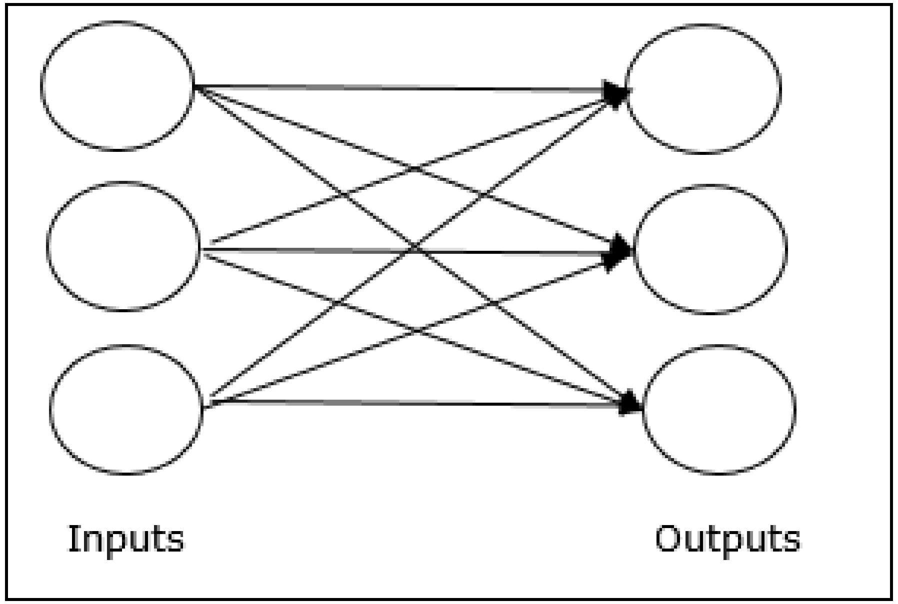
- A network topology is the arrangement of a network along with its nodes and connecting lines. According to the topology, ANN can be classified as the following kinds –

- Feedforward Network
- It is a non-recurrent network having processing units/nodes in layers and
- all the nodes in a layer are connected with the nodes of the previous layers.
- The connection has different weights upon them.
- There is no feedback loop means the signal can only flow in one direction, from input to output. It may be divided into the following two types –

•**Single layer feedforward network** –

•The concept is of feedforward ANN having only one weighted layer.

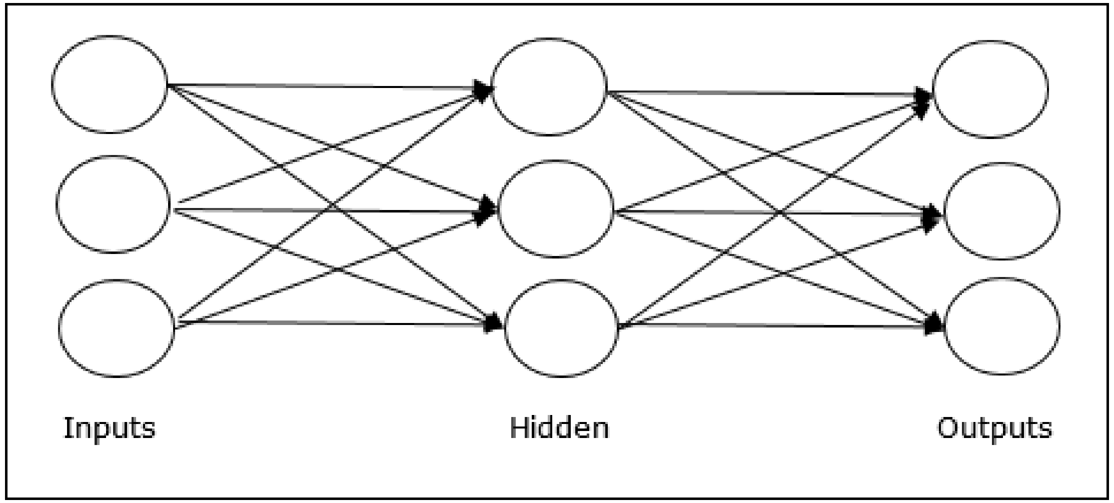
•In other words, we can say the input layer is fully connected to the output layer.



•**Multilayer feedforward network** –

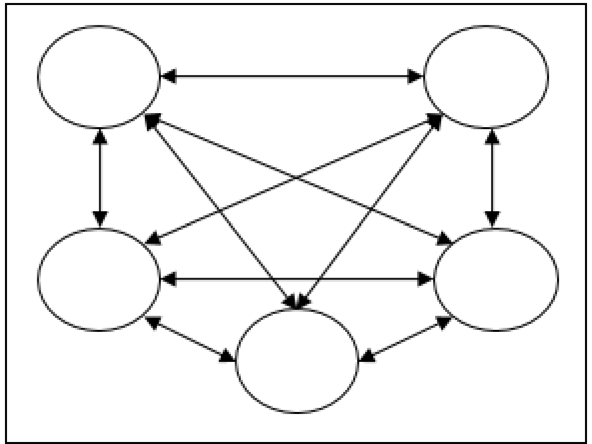
•The concept is of feedforward ANN having more than one weighted layer.

•As this network has one or more layers between the input and the output layer, it is called hidden layers.

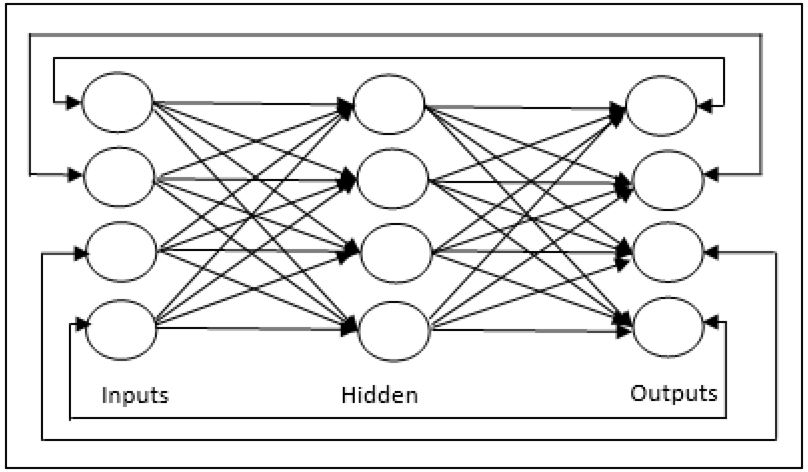


- FEEDBACK NETWORKS
- As the name suggests, a feedback network has feedback paths, which means the signal can flow in both directions using loops. This makes it a non-linear dynamic system, which changes continuously until it reaches a state of equilibrium. It may be divided into the following types −
- **Recurrent networks** − They are feedback networks with closed loops. Following are the two types of recurrent networks

•**Fully recurrent network** − It is the simplest neural network architecture because all nodes are connected to all other nodes and each node works as both input and output.

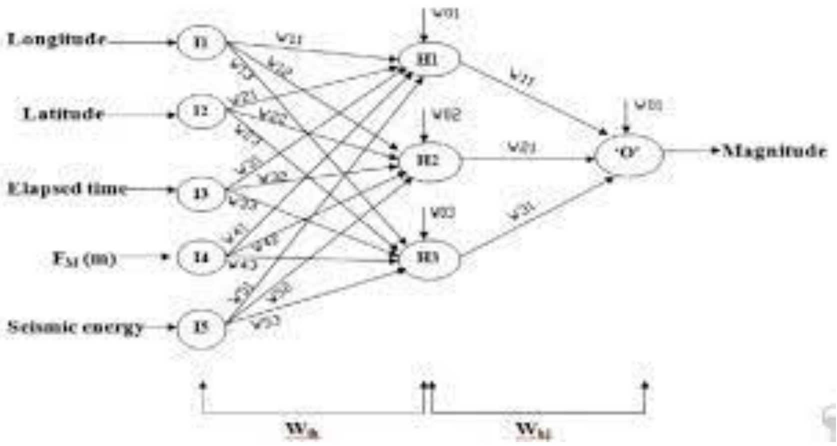


•**Jordan network** − It is a closed loop network in which the output will go to the input again as feedback as shown in the following diagram.

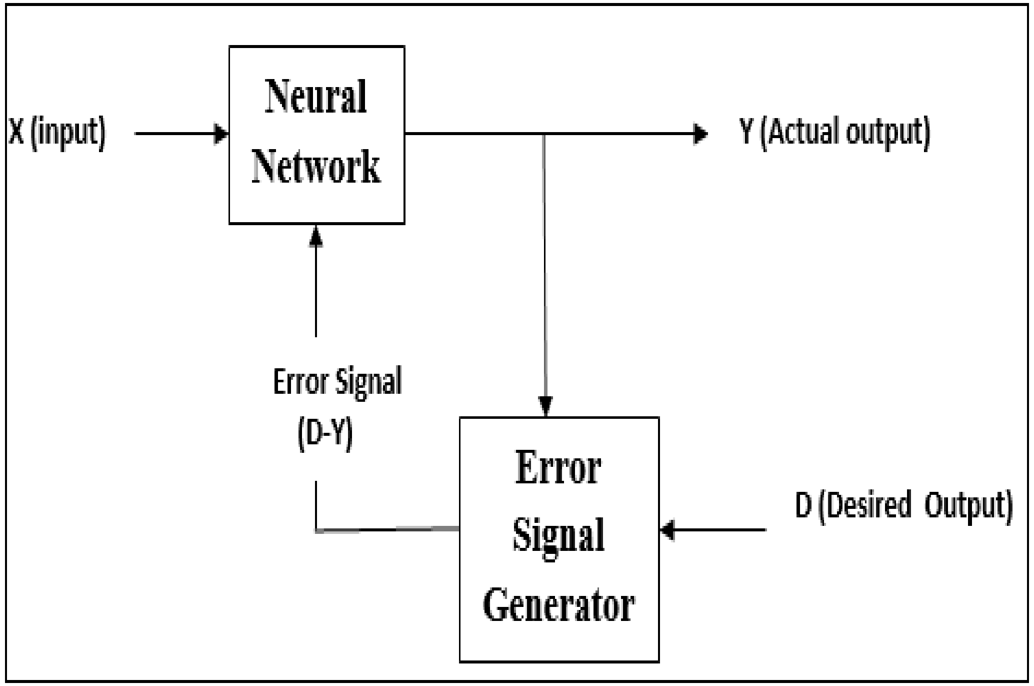


ADJUSTMENTS OF WEIGHTS OR LEARNING

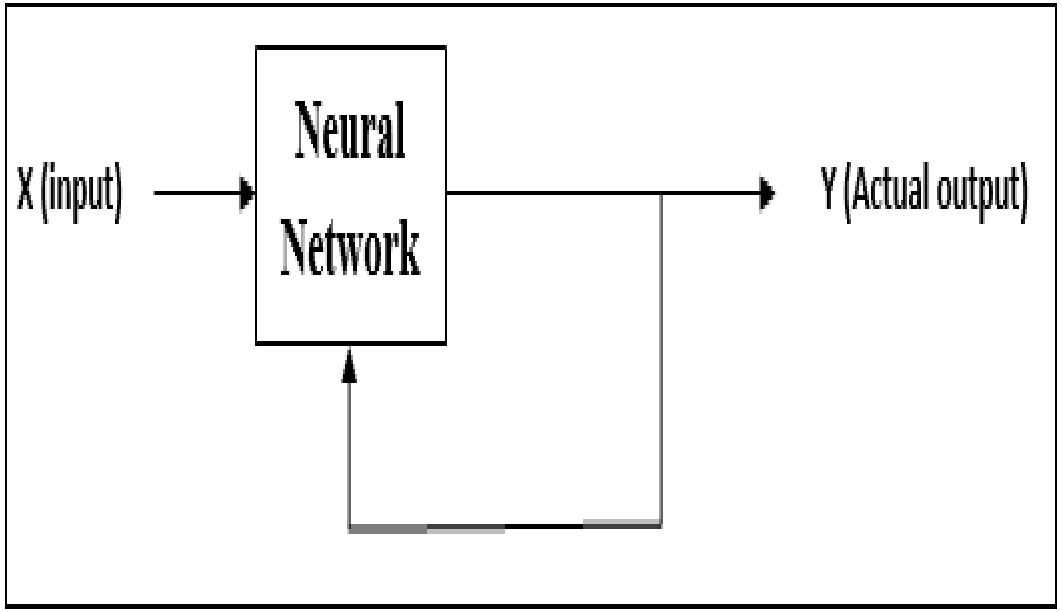
- Learning, in artificial neural network, is the method of modifying the weights of connections between the neurons of a specified network.
- Learning in ANN can be classified into three categories namely supervised learning, unsupervised learning, and reinforcement learning.



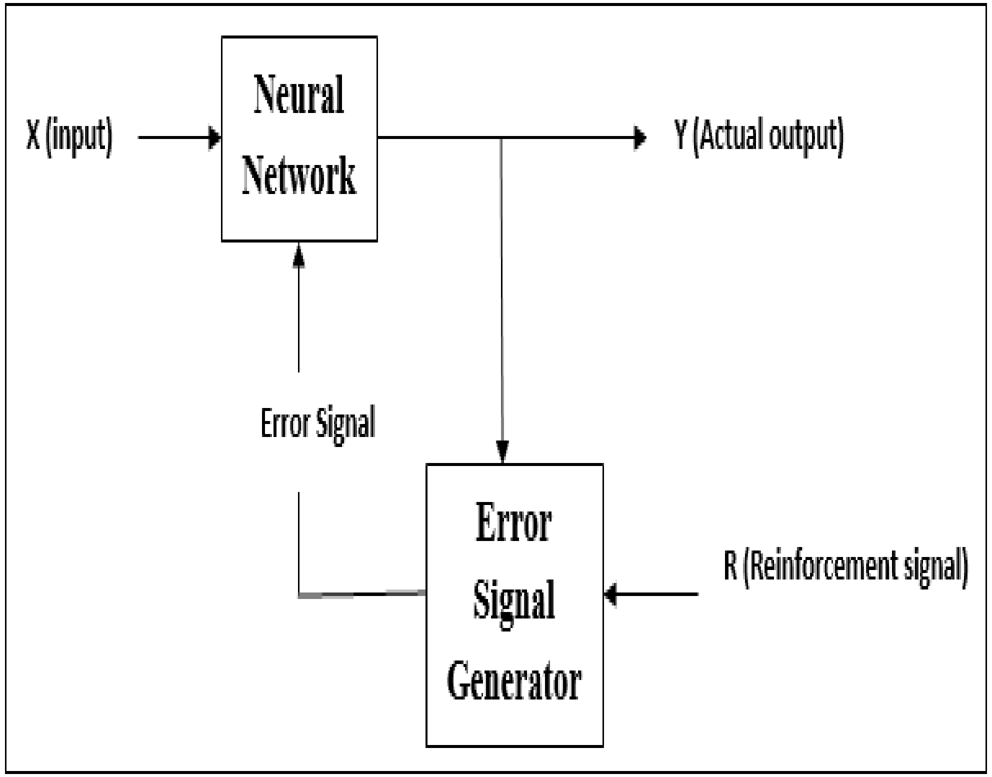
- Supervised Learning
- As the name suggests, this type of learning is done under the supervision of a teacher.
- This learning process is dependent.
- During the training of ANN under supervised learning, the input vector is presented to the network, which will give an output vector.
- This output vector is compared with the desired output vector.
- An error signal is generated, if there is a difference between the actual output and the desired output vector.
- On the basis of this error signal, the weights are adjusted until the actual output is matched with the desired output.



- Unsupervised Learning
- As the name suggests, this type of learning is done without the supervision of a teacher.
- This learning process is independent.
- During the training of ANN under unsupervised learning, the input vectors of similar type are combined to form clusters.
- When a new input pattern is applied, then the neural network gives an output response indicating the class to which the input pattern belongs.
- There is no feedback from the environment as to what should be the desired output and if it is correct or incorrect.
- Hence, in this type of learning, the network itself must discover the patterns and features from the input data, and the relation for the input data over the output.



- Reinforcement Learning
- As the name suggests, this type of learning is used to reinforce or strengthen the network over some critic information.
- This learning process is similar to supervised learning, however we might have very less information.
- During the training of network under reinforcement learning, the network receives some feedback from the environment.
- This makes it somewhat similar to supervised learning. However, the feedback obtained here is evaluative not instructive, which means there is no teacher as in supervised learning.
- After receiving the feedback, the network performs adjustments of the weights to get better critic information in future.



ACTIVATION FUNCTIONS

- It may be defined as the extra force or effort applied over the input to obtain an exact output. In ANN, we can also apply activation functions over the input to get the exact output. Followings are some activation functions of interest −
- Linear Activation Function
- It is also called the identity function as it performs no input editing. It can be defined as $F(x) = x$
- Sigmoid Activation Function
- It is of two type as follows −
- **Binary sigmoidal function** − This activation function performs input editing between 0 and 1. It is positive in nature. It is always bounded, which means its output cannot be less than 0 and more than 1. It is also strictly increasing in nature, which means more the input higher would be the output. It can be defined as

$$F(x) = sigm(x) = \frac{1}{1 + exp(-x)}$$

- **Bipolar sigmoidal function** − This activation function performs input editing between -1 and 1. It can be positive or negative in nature. It is always bounded, which means its output cannot be less than -1 and more than 1. It is also strictly increasing in nature like sigmoid function. It can be defined as

$$F(x) = sigm(x) = \frac{2}{1 + exp(-x)} - 1 = \frac{1 - exp(x)}{1 + exp(x)}$$

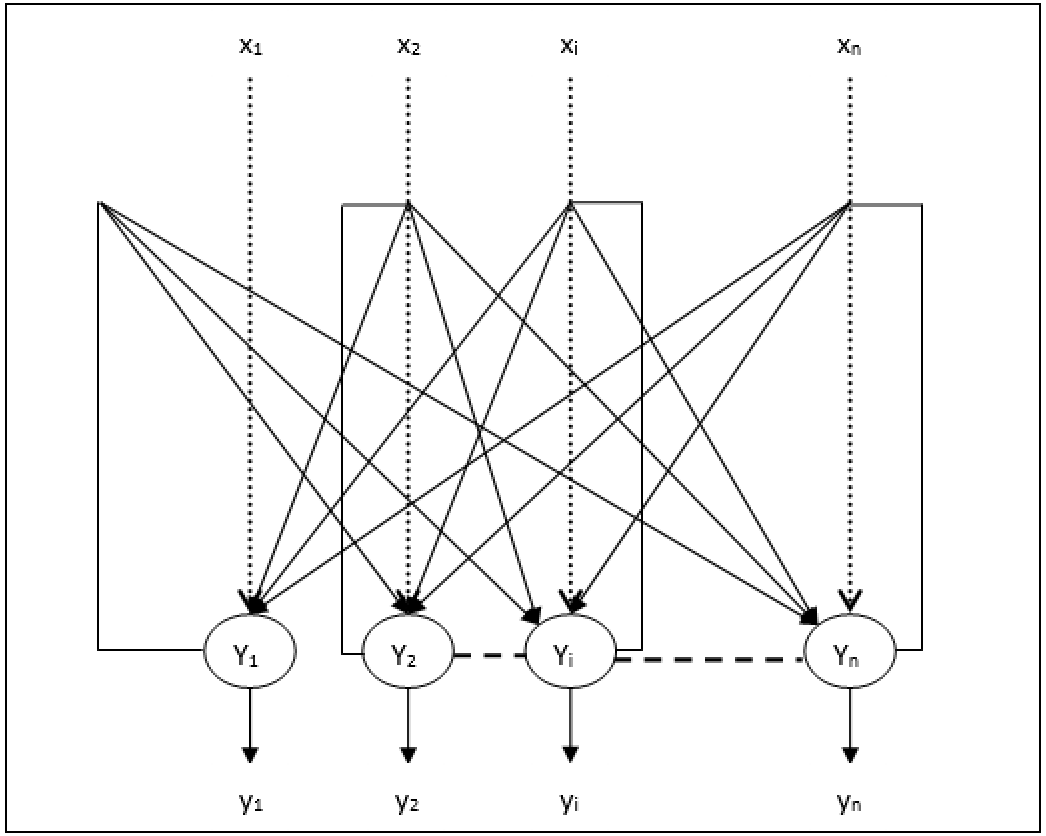
HOPFIELD NETWORKS

- Hopfield neural network was invented by Dr. John J. Hopfield in 1982.
- It consists of a single layer which contains one or more fully connected recurrent neurons.
- The Hopfield network is commonly used for auto-association and optimization tasks

DISCRETE HOPFIELD NETWORK

- A Hopfield network which operates in a discrete line fashion or in other words,
- it can be said the input and output patterns are discrete vector,
- which can be either binary 0,1
- or bipolar +1, -1
- The network has symmetrical weights with no self-connections i.e., $\mathbf{w_{ij} = w_{ji}}$ and $\mathbf{w_{ii} = 0}$

- Architecture
- Following are some important points to keep in mind about discrete Hopfield network −
- This model consists of neurons with one inverting and one non-inverting output.
- The output of each neuron should be the input of other neurons but not the input of self.
- Weight/connection strength is represented by $\mathbf{w_{ij}}$.
- Connections can be excitatory as well as inhibitory. It would be excitatory, if the output of the neuron is same as the input, otherwise inhibitory.
- Weights should be symmetrical, i.e. $\mathbf{w_{ij} = w_{ji}}$
- The output from $\mathbf{Y_1}$ going to $\mathbf{Y_2}$, $\mathbf{Y_i}$ and $\mathbf{Y_n}$ have the weights $\mathbf{w_{12}}$, $\mathbf{w_{1i}}$ and $\mathbf{w_{1n}}$ respectively. Similarly, other arcs have the weights on them



TRAINING ALGORITHM

- During training of discrete Hopfield network, weights will be updated.
- As we know that we can have the binary input vectors as well as bipolar input vectors.
- Hence, in both the cases, weight updates can be done with the following relation

Case 1 – Binary input patterns

For a set of binary patterns **s** p , **p = 1 to P**

Here, **s** p = **s**$_1$ p , **s**$_2$ p ,..., **s**$_i$ p ,..., **s**$_n$ p

Weight Matrix is given by

$$w_{ij} = \sum_{p=1}^{P}[2s_i(p) - 1][2s_j(p) - 1] \quad for\, i \neq j$$

Case 2 – Bipolar input patterns

For a set of binary patterns **s** p , **p = 1 to P**

Here, **s** p = **s**$_1$ p , **s**$_2$ p ,..., **s**$_i$ p ,..., **s**$_n$ p

Weight Matrix is given by

$$w_{ij} = \sum_{p=1}^{P}[s_i(p)][s_j(p)] \quad for\, i \neq j$$

TESTING ALGORITHM

- **Step 1** − Initialize the weights, which are obtained from training algorithm by using Hebbian principle.
- **Step 2** − Perform steps 3-9, if the activations of the network is not consolidated.
- **Step 3** − For each input vector **X**, perform steps 4-8.
- **Step 4** − Make initial activation of the network equal to the external input vector **X** as follows −

$$y_i = x_i \quad for\, i = 1\, to\, n$$

- **Step 5** − For each unit $\mathbf{Y_i}$, perform steps 6-9.
- **Step 6** − Calculate the net input of the network as follows −

$$y_{ini} = x_i + \sum_j y_j w_{ji}$$

- **Step 7** − Apply the activation as follows over the net input to calculate the output −

$$y_i = \begin{cases} 1 & if\, y_{ini} > \theta_i \\ y_i & if\, y_{ini} = \theta_i \\ 0 & if\, y_{ini} < \theta_i \end{cases}$$

Here θ_i is the threshold.

- **Step 8** − Broadcast this output $\mathbf{y_i}$ to all other units.
- **Step 9** − Test the network for conjunction.

ENERGY FUNCTION EVALUATION

- An energy function is defined as a function that is bonded and non-increasing function of the state of the system.
- Energy function $\mathbf{E_f}$, also called **Lyapunov function** determines the stability of discrete Hopfield network, and is characterized as follows −

$$E_f = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j w_{ij} - \sum_{i=1}^{n} x_i y_i + \sum_{i=1}^{n} \theta_i y_i$$

- **Condition** − In a stable network, whenever the state of node changes, the above energy function will decrease.

CONTINUOUS HOPFIELD FUNCTION

- Continuous Hopfield Network
- In comparison with Discrete Hopfield network, continuous network has time as a continuous variable. It is also used in auto association and optimization problems such as travelling salesman problem.
- **Model** − The model or architecture can be build up by adding electrical components such as amplifiers which can map the input voltage to the output voltage over a sigmoid activation function.
- Energy Function Evaluation

$$E_f = \frac{1}{2}\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n} y_i y_j w_{ij} - \sum_{i=1}^{n} x_i y_i + \frac{1}{\lambda}\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n} w_{ij} g_{ri} \int_0^{y_i} a^{-1}(y)\,dy$$

Here λ is gain parameter and g_{ri} input conductance.


Gracias
Thanks!
THANK
YOU!