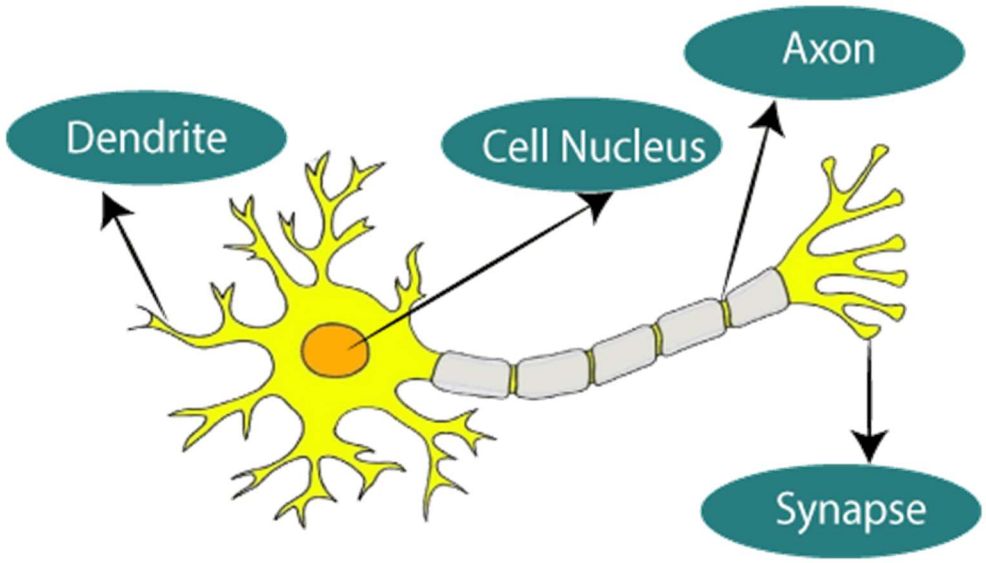
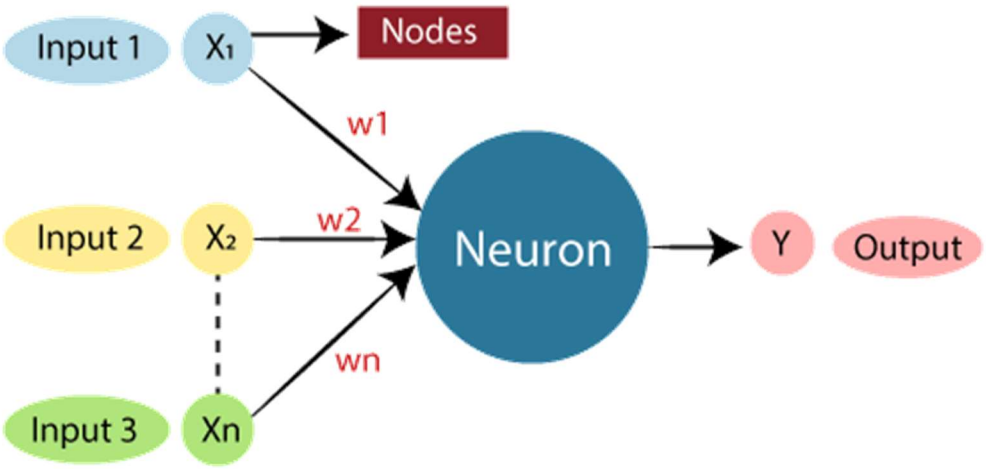
Artificial Neural Network

What is Artificial Neural Network?

The term "**Artificial Neural Network**" is derived from Biological neural networks that develop the structure of a human brain. Similar to the human brain that has neurons interconnected to one another, artificial neural networks also have neurons that are interconnected to one another in various layers of the networks. These neurons are known as nodes.



The typical Artificial Neural Network looks something like the given figure.



Dendrites from Biological Neural Network represent inputs in Artificial Neural Networks, cell nucleus represents Nodes, synapse represents Weights, and Axon represents Output.

Relationship between Biological neural network and artificial neural network:

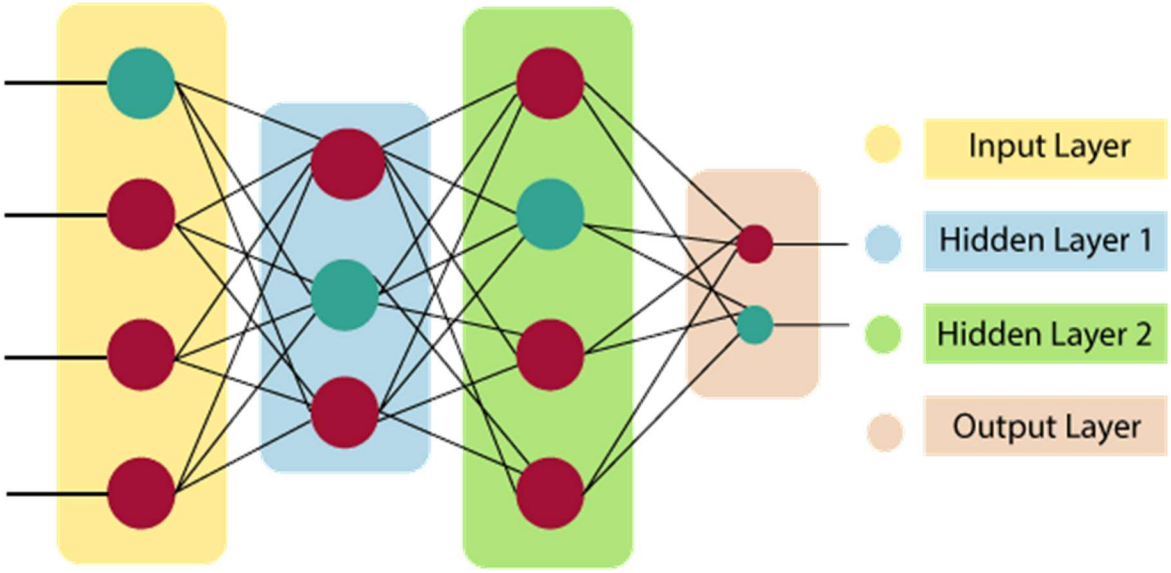
Biological Neural Network	Artificial Neural Network
Dendrites	Inputs
Cell nucleus	Nodes
Synapse	Weights
Axon	Output

An **Artificial Neural Network** in the field of **Artificial intelligence** where it attempts to mimic the network of neurons makes up a human brain so that computers will have an option to understand things and make decisions in a human-like manner. The artificial neural network is designed by programming computers to behave simply like interconnected brain cells.

The architecture of an artificial neural network:

To understand the concept of the architecture of an artificial neural network, we have to understand what a neural network consists of. In order to define a neural network that consists of a large number of artificial neurons, which are termed units arranged in a sequence of layers. Lets us look at various types of layers available in an artificial neural network.

Artificial Neural Network primarily consists of three layers:



Input Layer:

As the name suggests, it accepts inputs in several different formats provided by the programmer.

Hidden Layer:

The hidden layer presents in-between input and output layers. It performs all the calculations to find hidden features and patterns.

Output Layer:

The input goes through a series of transformations using the hidden layer, which finally results in output that is conveyed using this layer.

The artificial neural network takes input and computes the weighted sum of the inputs and includes a bias. This computation is represented in the form of a transfer function.

$$\sum_{i=1}^{n} Wi * Xi + b$$

It determines weighted total is passed as an input to an activation function to produce the output. Activation functions choose whether a node should fire or not. Only those who are fired make it to the output layer. There are distinctive activation functions available that can be applied upon the sort of task we are performing.

Advantages of Artificial Neural Network (ANN)

1. **Parallel processing capability:**

Artificial neural networks have a numerical value that can perform more than one task simultaneously.

2. **Storing data on the entire network:**

Data that is used in traditional programming is stored on the whole network, not on a database. The disappearance of a couple of pieces of data in one place doesn't prevent the network from working.

3. **Capability to work with incomplete knowledge:**

After ANN training, the information may produce output even with inadequate data. The loss of performance here relies upon the significance of missing data.

4. **Having a memory distribution:**

For ANN is to be able to adapt, it is important to determine the examples and to encourage the network according to the desired output by demonstrating these examples to the network. The succession of the network is directly proportional to the chosen instances, and if the event can't appear to the network in all its aspects, it can produce false output.

5. **Having fault tolerance:**

Extortion of one or more cells of ANN does not prohibit it from generating output, and this feature makes the network fault-tolerance.

Disadvantages of Artificial Neural Network:

1. **Assurance of proper network structure:**

There is no particular guideline for determining the structure of artificial neural networks. The appropriate network structure is accomplished through experience, trial, and error.

2. **Unrecognized behavior of the network:**

It is the most significant issue of ANN. When ANN produces a testing solution, it does not provide insight concerning why and how. It decreases trust in the network.

3. **Hardware dependence:**

Artificial neural networks need processors with parallel processing power, as per their structure. Therefore, the realization of the equipment is dependent.

4. **Difficulty of showing the issue to the network:**

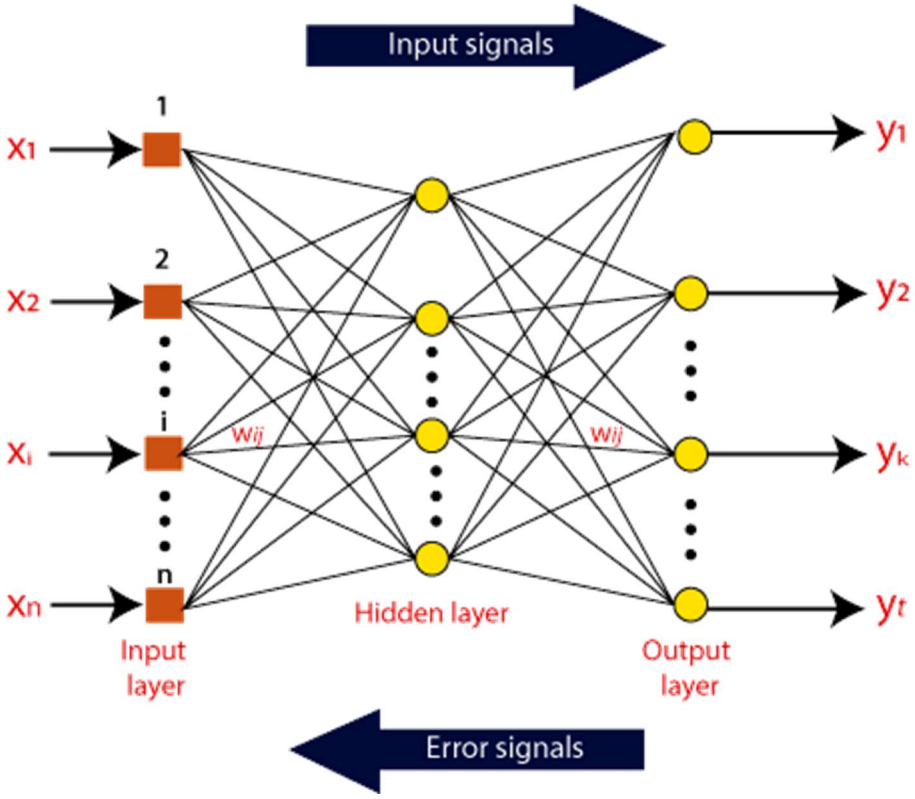
ANNs can work with numerical data. Problems must be converted into numerical values before being introduced to ANN. The presentation mechanism to be resolved here will directly impact the performance of the network. It relies on the user's abilities.

5. **The duration of the network is unknown:**

The network is reduced to a specific value of the error, and this value does not give us optimum results.

How do artificial neural networks work?

Artificial Neural Network can be best represented as a weighted directed graph, where the artificial neurons form the nodes. The association between the neurons outputs and neuron inputs can be viewed as the directed edges with weights. The Artificial Neural Network receives the input signal from the external source in the form of a pattern and image in the form of a vector. These inputs are then mathematically assigned by the notations x(n) for every n number of inputs.



Afterward, each of the input is multiplied by its corresponding weights (these weights are the details utilized by the artificial neural networks to solve a specific problem). In general terms, these weights normally represent the strength of the interconnection between neurons inside the artificial neural network. All the weighted inputs are summarized inside the computing unit.

If the weighted sum is equal to zero, then bias is added to make the output non-zero or something else to scale up to the system's response. Bias has the same input, and weight equals to 1. Here the total of weighted inputs can be in the range of 0 to positive infinity.

Here, to keep the response in the limits of the desired value, a certain maximum value is benchmarked, and the total of weighted inputs is passed through the activation function.

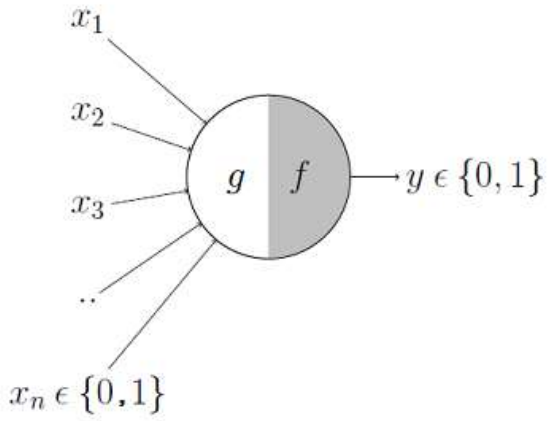
The activation function refers to the set of transfer functions used to achieve the desired output. There is a different kind of the activation function, but primarily either linear or non-linear sets of functions.

The neuron doesn't really know how to bound to value and thus is not able to decide the firing pattern. Thus, the activation function is an important part of an artificial neural network. They basically decide whether a neuron should be activated or not. Thus, it bounds the value of the net input.

It is very well known that the most fundamental unit of deep neural networks is called an artificial neuron/perceptron. But the very first step towards the perceptron we use today was taken in 1943 by McCulloch and Pitts, by mimicking the functionality of a biological neuron.

McCulloch-Pitts Neuron

The first computational model of a neuron was proposed by Warren MuCulloch (neuroscientist) and Walter Pitts (logician) in 1943.



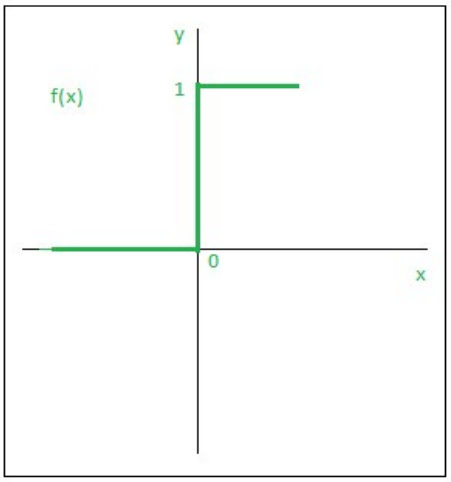
Types of Activation Functions

some of AF are discussed below:

1- **Step Function**:
Step Function is one of the simplest kinds of activation functions. In this, we consider a threshold value and if the value of net input say **y** is greater than the threshold then the neuron is activated.

$$f(x) = 1, \text{if x} >= 0$$
$$f(x) = 0, \text{if x} < 0$$

Given below is the graphical representation of step function.

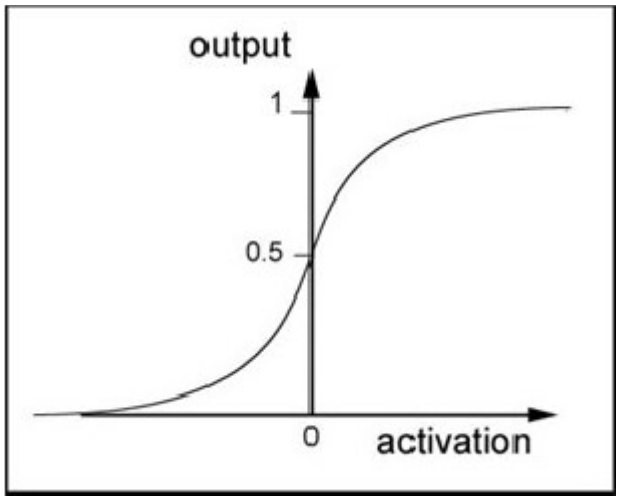


2- Sigmoid Function:

It is of two types as follows −

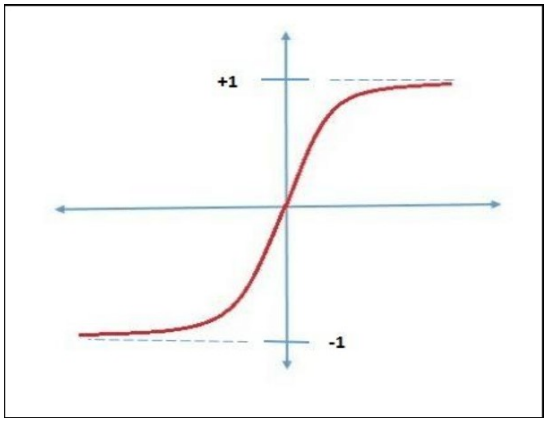
- **Binary sigmoidal function** − This activation function performs input editing between 0 and 1. It is positive in nature. It is always bounded, which means its output cannot be less than 0 and more than 1. It is also strictly increasing in nature, which means more the input higher would be the output. It can be defined as

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



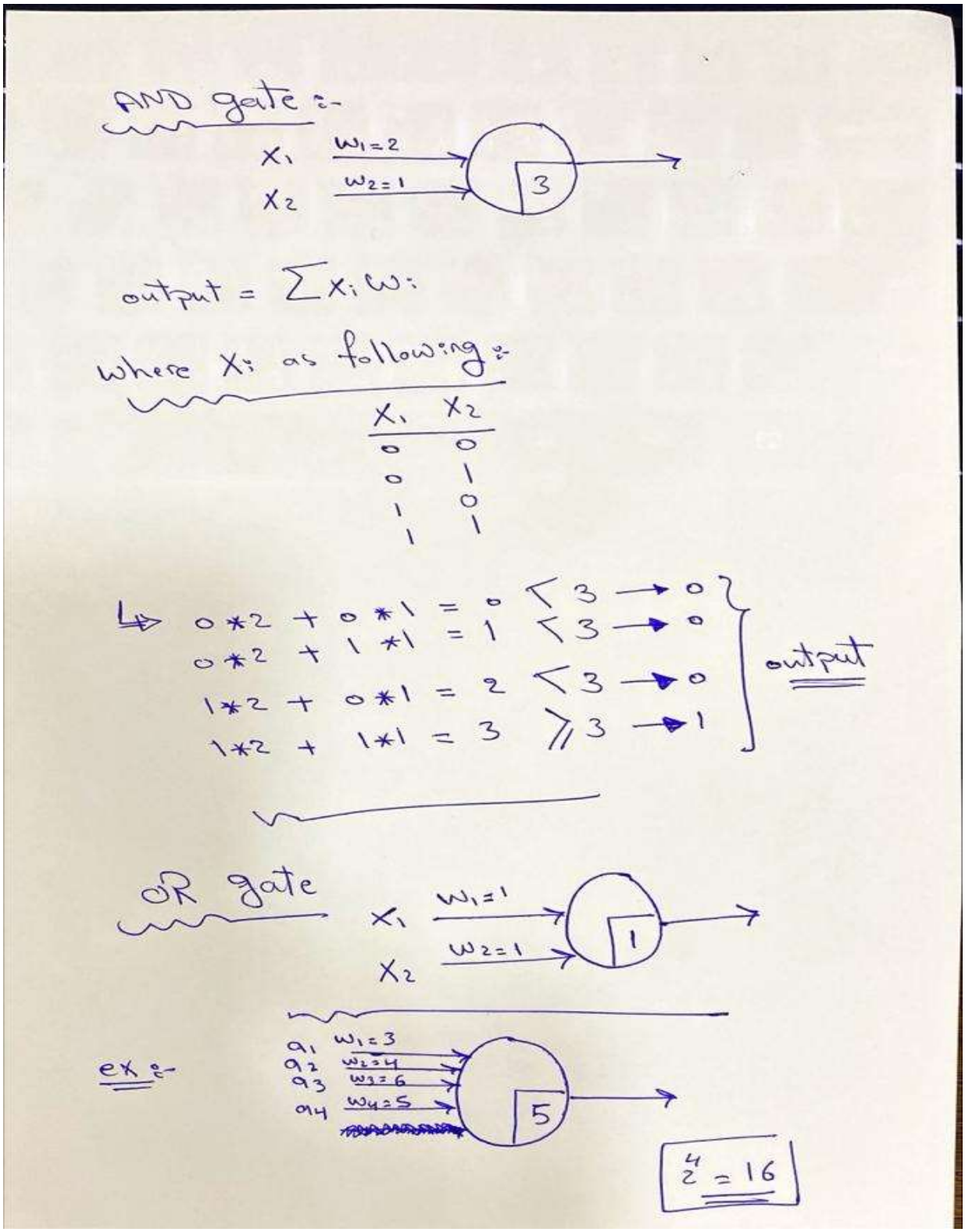
- **Bipolar sigmoidal function** − This activation function performs input editing between -1 and 1. It can be positive or negative in nature. It is always bounded, which means its output cannot be less than -1 and more than 1. It is also strictly increasing in nature like sigmoid function. It can be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

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

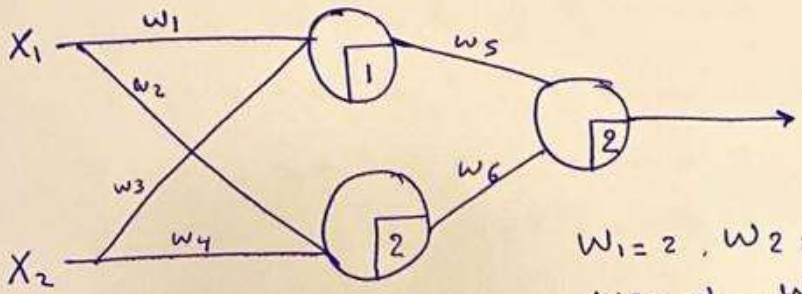


Boolean Functions Using M-P Neuron

AND - OR Gate



Solve for XOR gate

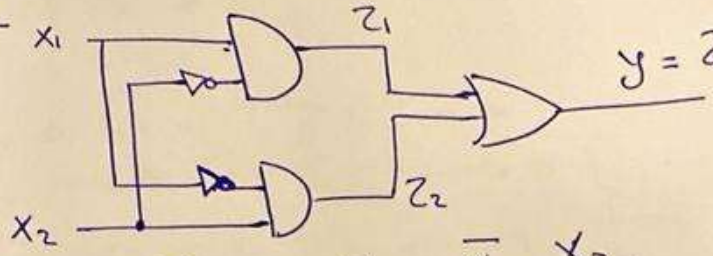


$w_1 = 2$, $w_2 = -1$

$w_3 = -1$, $w_4 = 2$

$w_5 = 2$, $w_6 = 2$.

ans :-



$Z_1 = X_1 \cdot \bar{X}_2$, $Z_2 = \bar{X}_1 \cdot X_2$.

X_1	X_2	$\bar{X}_1$	$\bar{X}_2$
0	0	1	1
0	1	1	0
1	0	0	1
1	1	0	0

$Z_1 = X_1 \cdot \bar{X}_2$ Threshold = 1

$0*2 + 1*-1 = -1 < 1 \rightarrow 0$

$0*2 + 0*-1 = 0 < 1 \rightarrow 0 \Rightarrow Z_1$

$1*2 + 1*-1 = 1 \geq 1 \rightarrow 1$

$1*2 + 0*-1 = 2 > 1 \rightarrow 1$

$Z_2 = \overline{X_1} \cdot X_2$ threshold = 2

$0 * -1 + 1 * 2 = 2 \geq 2 \rightarrow 1$

$1 * -1 + 1 * 2 = 1 < 2 \rightarrow 0 \Rightarrow Z_2$.

$0 * -1 + 0 * 2 = 0 < 2 \rightarrow 0$

$1 * -1 + 0 * 2 = -1 < 2 \rightarrow 0$

$Z_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ $Z_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$ threshold = 2

$1 * 2 + 0 * 2 = 2 \geq 2 \rightarrow 1$

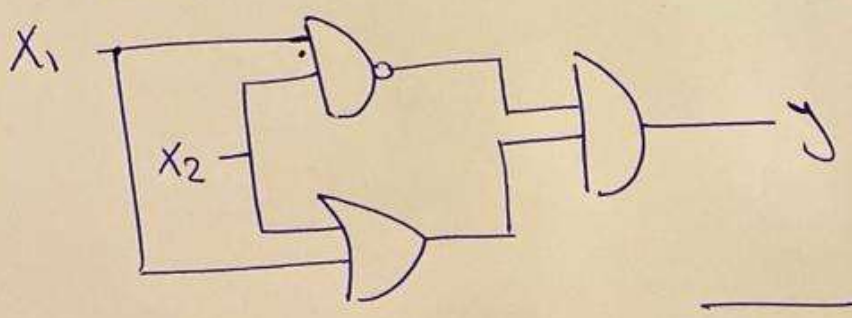
$0 * 2 + 0 * 2 = 0 < 2 \rightarrow 0$

$0 * 2 + 1 * 2 = 2 \geq 2 \rightarrow 1$

$0 * 2 + 1 * 2 = 2 \geq 2 \rightarrow 1$

$\Rightarrow$ final o/p

also we can implement XOR using the below:-



$y = (X_1 + X_2) \cdot \overline{(X_1 \cdot X_2)}$

H.W

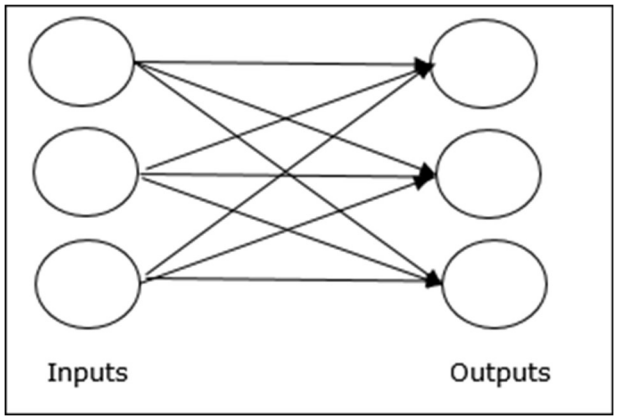
Network Topology

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 −

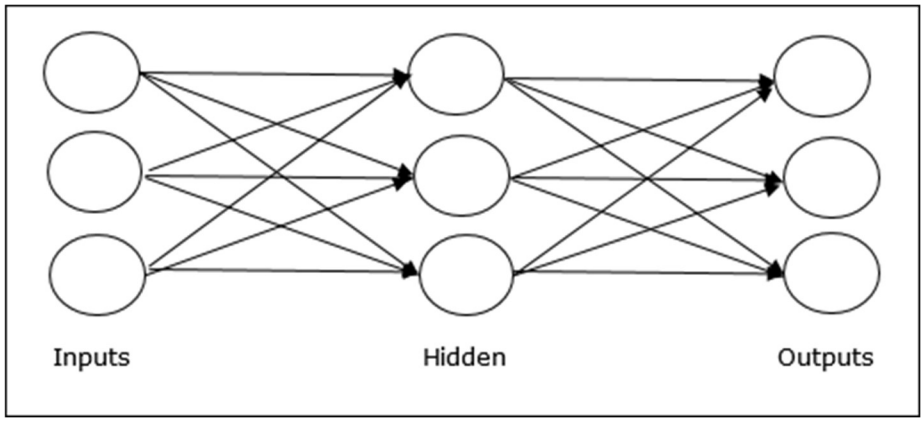
Feedforward Network

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.

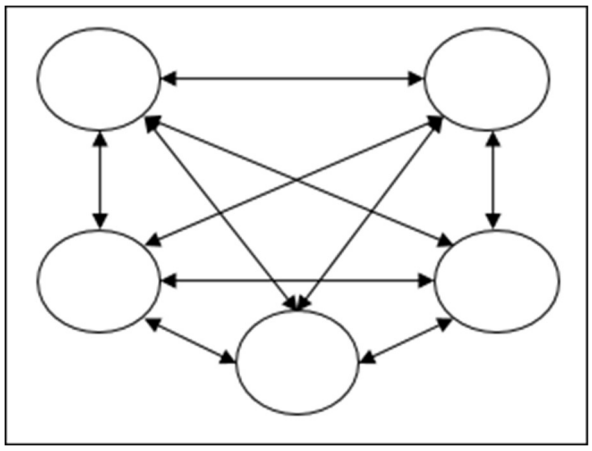


Feedback Network

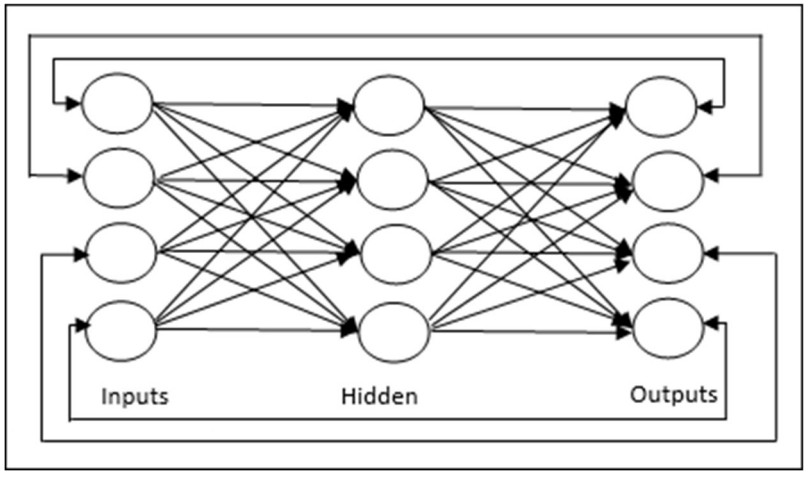
As the name suggests, a feedback network has feedback paths, which means the signal can flow in both directions using loops. This makes it a non-linear dynamic system, which changes continuously until it reaches a state of equilibrium. It may be divided into the following types −

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.



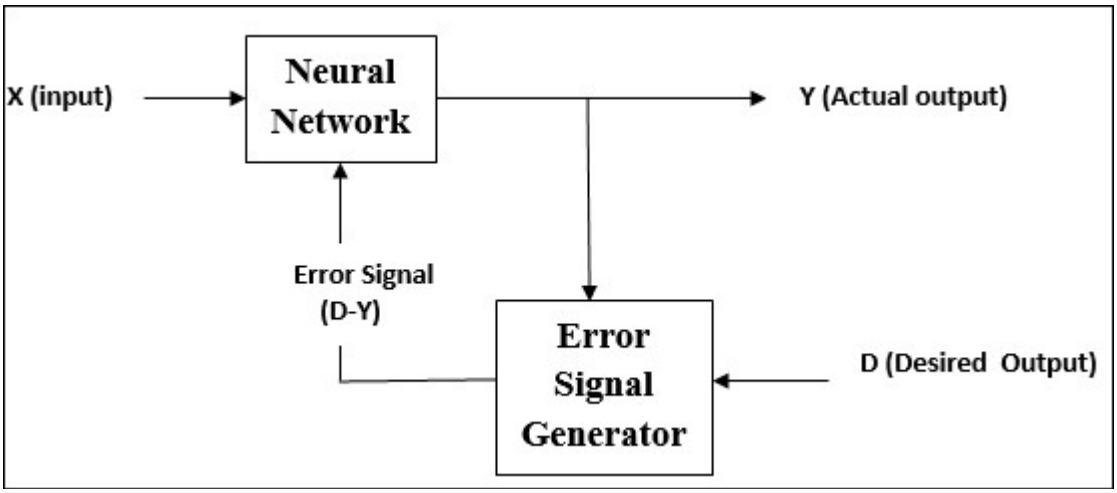
Adjustments of Weights or Learning

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.

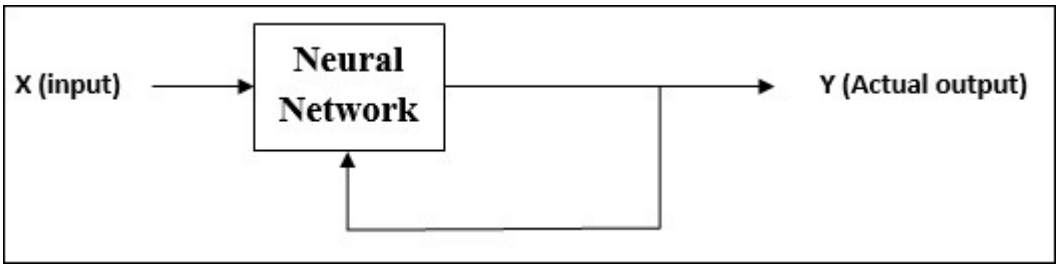


Unsupervised Learning

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.