By - Niraj

mimic the str of human brain

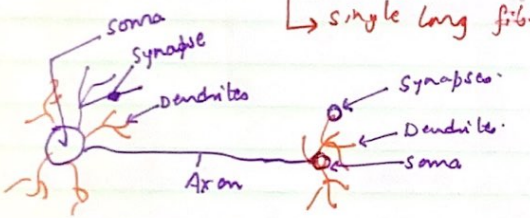
Artificial Neural Network

→ Brain consists of processing units called neurons

→ Human Brain consists of 10 billion neurons and 60 trillion connection kla Synapses bela thevu.

→ By using these multiple neurons, the brain can perform its function much faster than fastest computer in existence.

Neurons :→ consists of → cell body, kla soma
→ no. of fibres kla Dendrite
→ single long fibre kla axon.

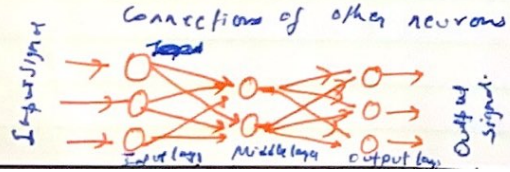


Artificial Neural Network : (ANN)

→ ANN consists of no. of processors called Neurons analogus to biological neurons

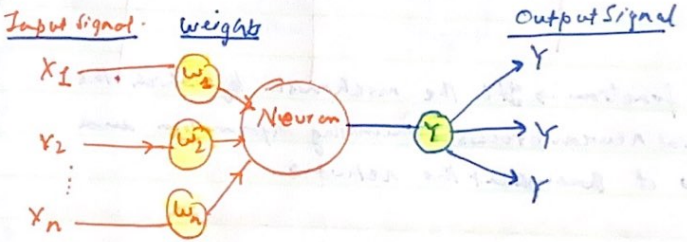
→ Neurons are connected by weighted links passing signals from one neuron to another

→ outgoing signals splits into no. of branches that transmit the same signal. The outgoing branches terminate at the incoming connections of other neurons in the network.



Biological Neural Network.	Artificial Neural Network.
Soma	Neuron
Dendrite	Input
Axon	Output
Synapse	Weight.

Diagram of neuron :→



→ Neurons compute weighted sum of the input signals and compares the result with the threshold value θ

→ If net input is less than threshold → neuron output is -1

→ " " " " equal or > " threshold → " " " is +1

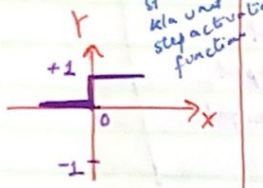
→ Neuron uses the transfer or activation function.

$$X = \sum_{i=1}^{n} x_i w_i \qquad Y = \begin{cases} +1, & \text{if } X \geq \theta \\ -1, & \text{if } X < \theta \end{cases}$$

→ This type of activation function is called a sign function

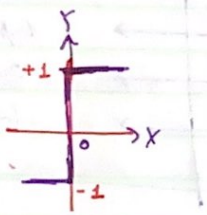
Possible activation function of a neuron

Step function (looks like stair ... kla und step activation function)



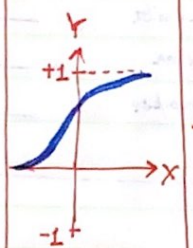
$$Y^{step} = \begin{cases} 1, & \text{if } X \geq 0 \\ 0, & \text{if } X < 0 \end{cases}$$

Sign function



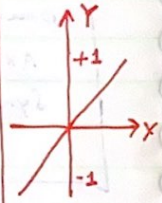
$$Y^{sign} = \begin{cases} +1, & \text{if } X \geq 0 \\ -1, & \text{if } X < 0 \end{cases}$$

Sigmoid function



$$Y^{sigmoid} = \frac{1}{1+e^{-X}}$$

Linear function



$$Y^{linear} = X$$

Activation function → It is the mechanism by which the artificial neuron processes incoming information and passes it throughout the network.

It sums the total input signals
If input signal meets threshold → Neuron passes the signal.

Otherwise it does nothing

Sigmoid activation function :→

→ most commonly used alternative is Sigmoid activation function
→ e is the base of natural algorithm
→ "S" shaped

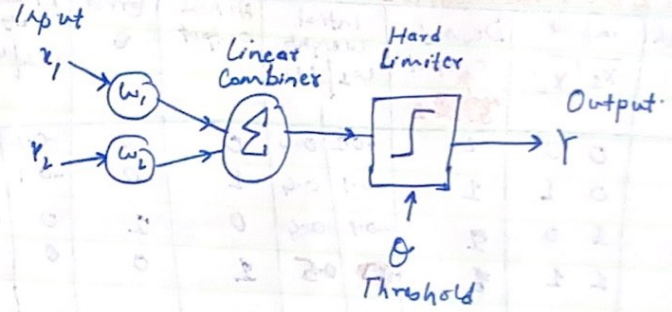
$$Y = \frac{1}{1+e^{-x}}$$

→ The output signal is no longer binary; the output values can fall anywhere in the range 0 to 1.
→ Additionally, the sigmoid is differentiable, which means that it is possible to calculate the derivatives across the entire range of inputs.

Perceptron: →

→ Simplest form of neural network.
→ Consists of a single neuron with adjustable synaptic weight.
→ Uses step function as the activation function, also is/a hard limiter

Single layer two input perceptron



How does the perceptron learn its classification tasks?

→ This is done by making small adjustments in the weights to reduce the difference betn actual and desired output of the perceptions

→ Initial wt are randomly assigned, usually [0.5, 0.5],
→ and then updated to obtain the output consistent with the training examples.

(Q)

Create a ~~regular~~ table for the logical operator OR.

threshold θ is 0.1

learning rate α is 0.1

initial $wt_1 = -0.1$, $w_2 = 0.4$

How many epochs did it take to learn the OR operation

Note: → $X_1 w_1 + X_2 w_2$ (input, weight) if XW value is less than θ → Then 0, other wise 1

→ error = Desired wt - Actual. # If there is error we will increase weight by α i.e $\alpha * X$ ← from wt

In OR operator, if inputs 1, 0 → Desired output is 1

OR operator

Epoch	Input X_1	Input X_2	Desired Output	Initial Weight w_1	Initial Weight w_2	Actual output	Error e	Final weight w_1	Final weight w_2
1	0	0	0	-0.1	0.4	0	0	-0.1	0.4
	0	1	1	-0.1	0.4	1	0	-0.1	0.4
	1	0	1	-0.1	0.4	0	1	0	0.5
	1	1	1	0	0.5	1	0	0	0.5
2	0	0	0	0	0.5	0	0	0	0.5
	0	1	1	0	0.5	1	0	0	0.5
	1	0	1	0	0.5	0	1	0.1	0.6
	1	1	1	0.1	0.6	1	0	0.1	0.6
3	0	0	0	0.1	0.6	0	0	0.1	0.6
	0	1	1	0.1	0.6	1	0	0.1	0.6
	1	0	1	0.1	0.6	1	0	0.1	0.6
	1	1	1	0.1	0.6	1	0	0.1	0.6

It tooks 3 epochs to learn the or operator.

At iteration P,

$e(p) = Y_d(p) - Y(p)$

↑ error, ↑ desired, ↑ actual output

if error is +ve ← increase perception by learning rate α

if error is -ve ← increase " " " " α

The perception learning rule

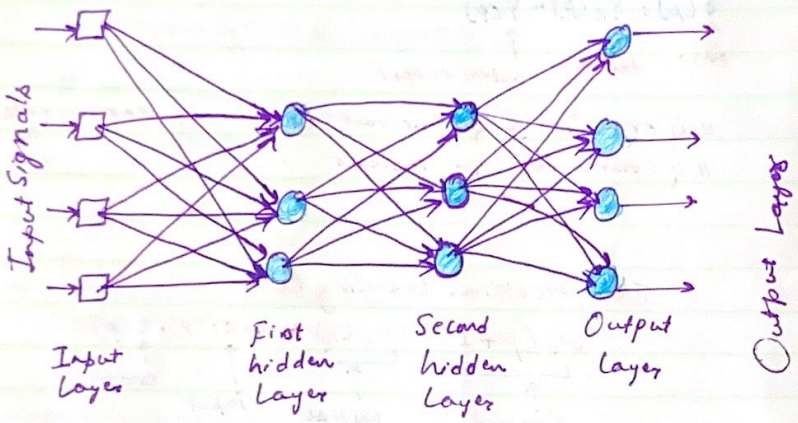
$w_i(p+1) = w_i(p) + \alpha \cdot x_i(p) \cdot e(p)$

↑ increased wt, ↑ initial wt, ↑ input, ↑ error

where, $p = 1, 2, 3 \ldots$

α = learning rate, a positive constant less than unity

Multilayer perception with two hidden Layers



What does the middle layer hide?

- Hidden layer "hides" its desired output
- Neurons in the hidden layer cannot be observed through the input/output behaviour of the network.
- There is no obvious way to know what the desired output of the hidden layer should be.

Blackbox → There is pair of machine learning methods Neural Network and Support Vector machines that they may appear at first glance to be a magic. Though extremely powerful, difficult to understand their inner working. This is referred as blackbox.

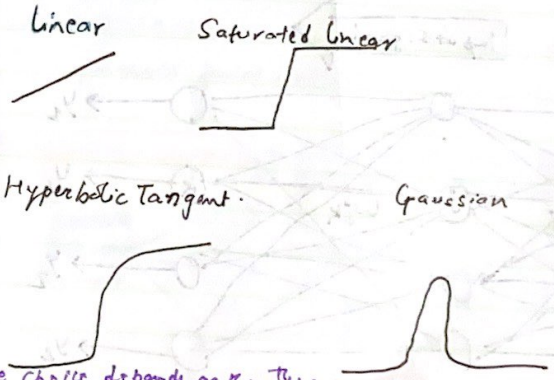
In machine learning → black box is due to the complex mathematics allowing them to functions

Application of ANN

- Speech and handwriting recognition programs
- automation of smart devices like self driving cars, Self ... drones.
- Sophisticated models of weather and climate patterns, tensile strength, fluid dynamics, and many other scientific, social or economic phenomena.

ANN can be applied to nearly any learning tasks
- classification
- Numeric pred^n
- ~~pred~~ unsupervised pattern recognition

Activation function:



The choice depends on the type of data.

A linear activation function results in a neural network similar to linear regression model.

Gaussian activation function results in a model called a Radial Basis function (RBF) network.

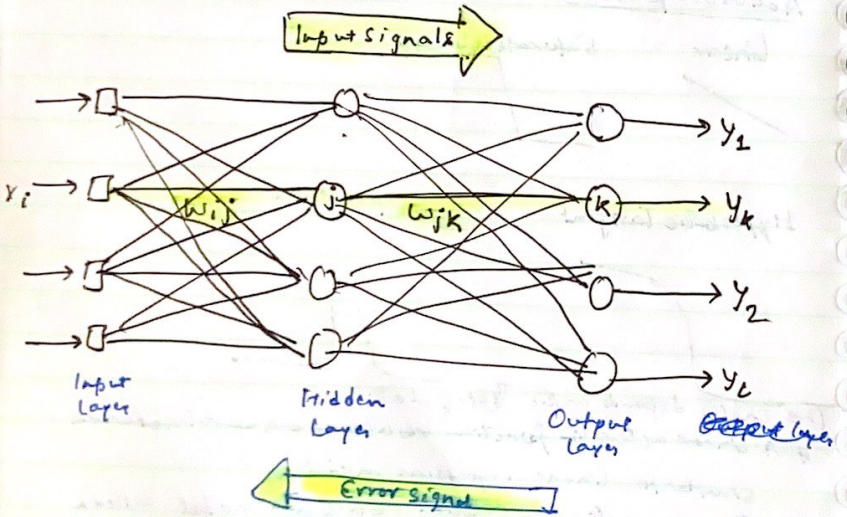
Network topology :→

pattern & structure of interconnected neurons

Network architecture depends upon
- No of layers
- whether information in the network is allowed to travel backward
- The number of nodes within each layer of the network.

Generally larger & more complex networks are capable of identifying more subtle patterns and complex decision boundaries.

Eg **Three layer back propagation neural network**

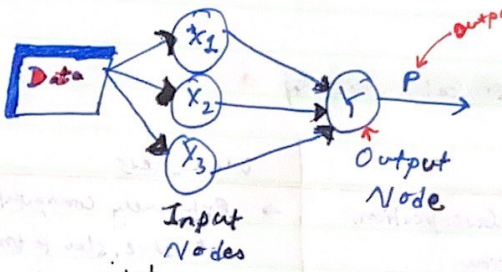


network propagates the input pattern from layer to layer until the output pattern is generated

If output pattern differ from the desired output, an error is calculated and then propagated backwards from output to input layer. The wts are modified as error is propagated

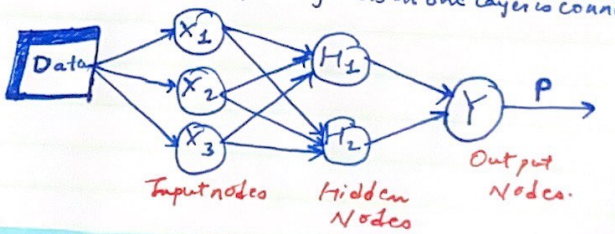
The number of layers

Single-layer network



Each input node process a single features in dataset.
Feature's value will be transformed by the corresponding node's activation function
Signal will be processed from input node to the output node
Output nodes uses it own activation function to generate final output.

Multilayer networks
→ Adds one or more hidden layers.
→ Every nodes in one layer is connected to next layer



Deep Neural Network :→
A neural network with multiple hidden layers is called a Deep Neural Network (DNN)

The practice of training such network is sometimes referred as deep learning