Robotics Club Sastra

Computer Vision

Workshop

2023

Dhiraj
karthik
Manindra
Vaibhav

Models to be discussed :

1. Linear Regression : Its a linear relationship between decision class & one or more independent input variables.

Types : (1) simple (one input variable)
(2) multiple (multiple input variable)

Consider a dataset with n+1 columns, where n columns are input variables and the remaining one is the decision class.

Analysis : By definition of lincar regression

$y = W_0 + W_1 x_1 + W_2 x_2 + \dots + W_n x_n + \epsilon$

y = o/p variable

$x_1, x_2, x_3, \dots x_n$ = i/p variables

$W_1, W_2, \dots W_n$ = weights associated with $x_1, x_2, \dots x_n$ respectively.

ϵ - error

W_0 - bias

The above discussed scenario is a multiple linear regression type

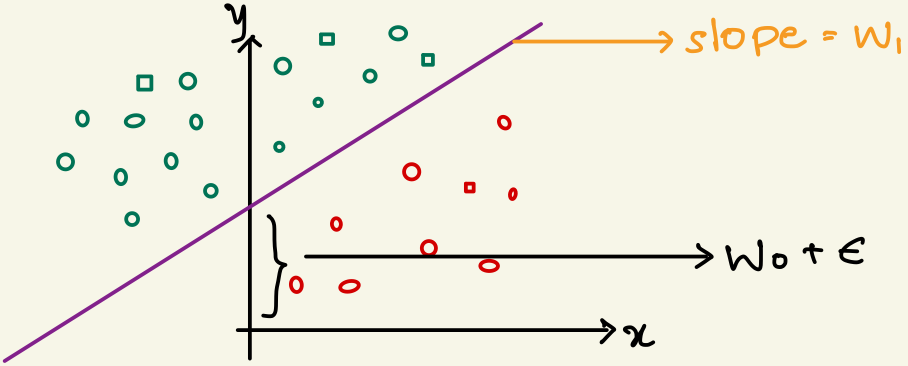
For a simple regression the equation will be of the form :

$$y = W_0 + W_1 x + \epsilon$$

Eg:

○ - data points of class1

○ · data points of class2



compare above equation with $y = mx + c$

$m = W_1$ & $c = W_0 + \epsilon$

2. Logistic Regression

→ a supervised ML technique that Predicts output using the probability calculated on the basis of the input variables

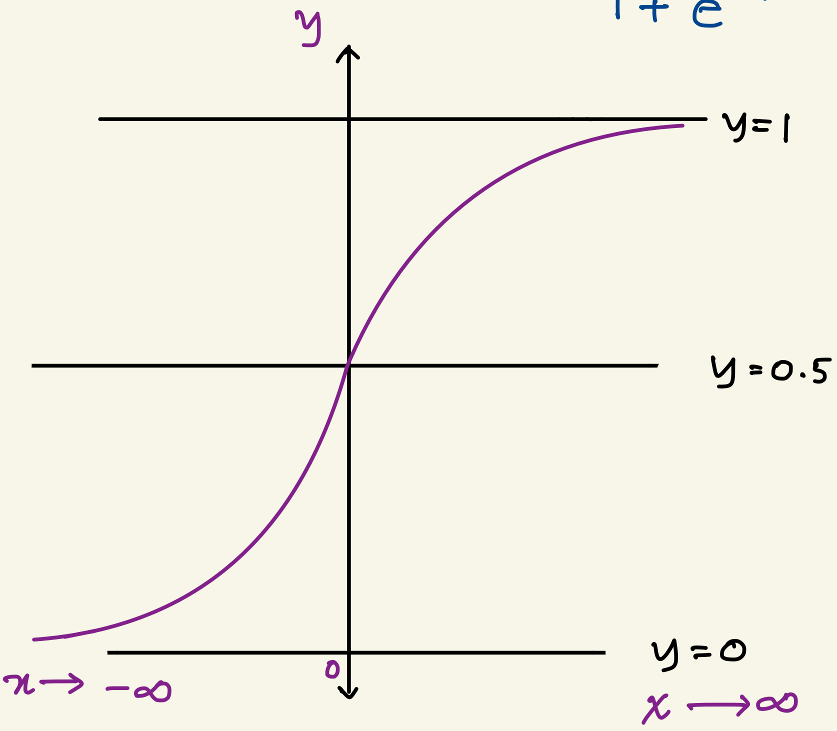
→ output : lies b/w 0 & 1

Derivation :- Its derivation makes use of multiple linear regression as an intermediate :

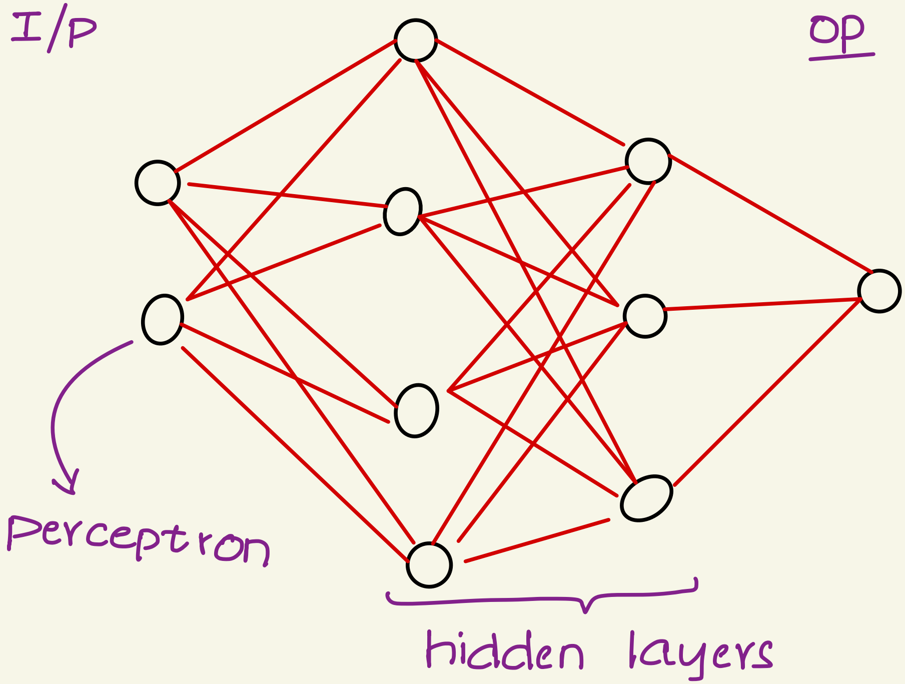
$$y = W_0 + W_1 x_1 + W_2 x_2 + W_3 x_3 + \cdots + W_n x_n$$

Final step is about applying sigmoid function to the above obtained output (y).

Sigmoid function: $y = f(x) = \dfrac{1}{1 + e^{-x}}$



3. **Neural Network :** a network which is a collection of perceptions that are mathematically modelled to perform a prediction with two or more layers



Types :

1. ANN (Artificial Neural Network) :
It is simplest variant of Neural network.
It is a feed-forward only network, initialized with random weights.

(II) CNN(convolutional Neural Networks) -

These types of neural networks are nothing but ANN having convolutional layers (one or more). convolutional layers are the layers of a neural network that breaks a region image into blocks of data & hence non-linear processing is performed.

(III) RNN (recurrent neural network) :

RNN are neural networks with perceptrons having a memory. one of the famous models is LSTM (long short term memory). It takes a time series data as i/p. Hence it is suitable for applications that use weather forecasting etc.

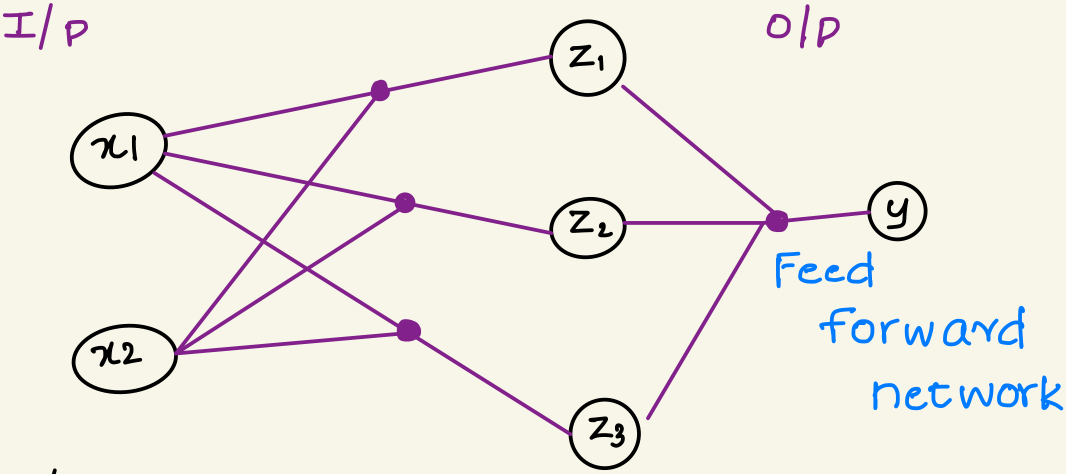
working of neural networks:

given:

no of i/p = 2   hidden layer1 = 3

no of o/p = 1

$$\text{weight matrix} = \begin{bmatrix} w1 & w2 \\ w3 & w4 \\ w5 & w6 \end{bmatrix} \rightarrow \text{randomly initialized}$$



I/P cells : x_1, x_2

Hidden layer cells : Z_1, Z_2, Z_3

O/P cells : y

Step 1 : calculate values at point a,b,c given by :-

$$a = W_1 x_1 + W_2 x_2 + B$$

$$b = W_3 x_1 + W_4 x_2 + B$$

$$c = W_5 x_1 + W_6 x_2 + B$$

Where B = Bias (Assumption)

Step 2 : apply activation function on a, b, c to get Z1, Z2, Z3 respectively. activation function can be of your choice depending ondataset.

Various activation functions are available such as: sigmoid, Relu (rectified linear unit)

so $z_1 = \text{sigmoid}(a)$

$$z_1 = 1/1+e^{-a}$$

simply, $z_2 = 1/1+e^{-b}$ $z_3 = 1/1+e^{-c}$

This step has to be repeated until we reach last hidden layer.

Step3 : The same process as above is carried out to find value of 'y' from hidden layer, since the weight matrix was randomly initialized, there is a need to tune the matrix. mathematically.

First calculate error as

$$\boxed{\text{error} = \text{Actual} - \text{Predicated}}$$

let n denote learning rate generally taken equal to 0.1 then, after one succesful forward traversal, the weight matrix is updated as follows :—

$$\boxed{W_{new} = W_{old} - \eta * \text{error}}$$

(or) in the above case

$$\begin{bmatrix} w_1^{new} & w_2^{new} \\ w_3^{new} & w_4^{new} \\ w_5^{new} & w_6^{new} \end{bmatrix} = \begin{bmatrix} w_1 & w_2 \\ w_3 & w_4 \\ w_5 & w_6 \end{bmatrix} - \eta * error$$

For every record in the dataset the error is calculated during training and an appropriate weight matrix is derived by the end.

Robotics
Club
Sastra