Deep learning

25 April 2025 09:41

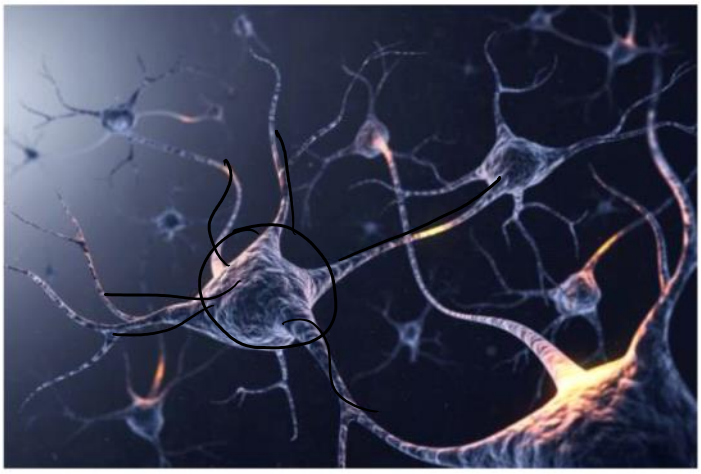
Deep understanding about the data in recognition the patterns in between x and y, is called deep learning

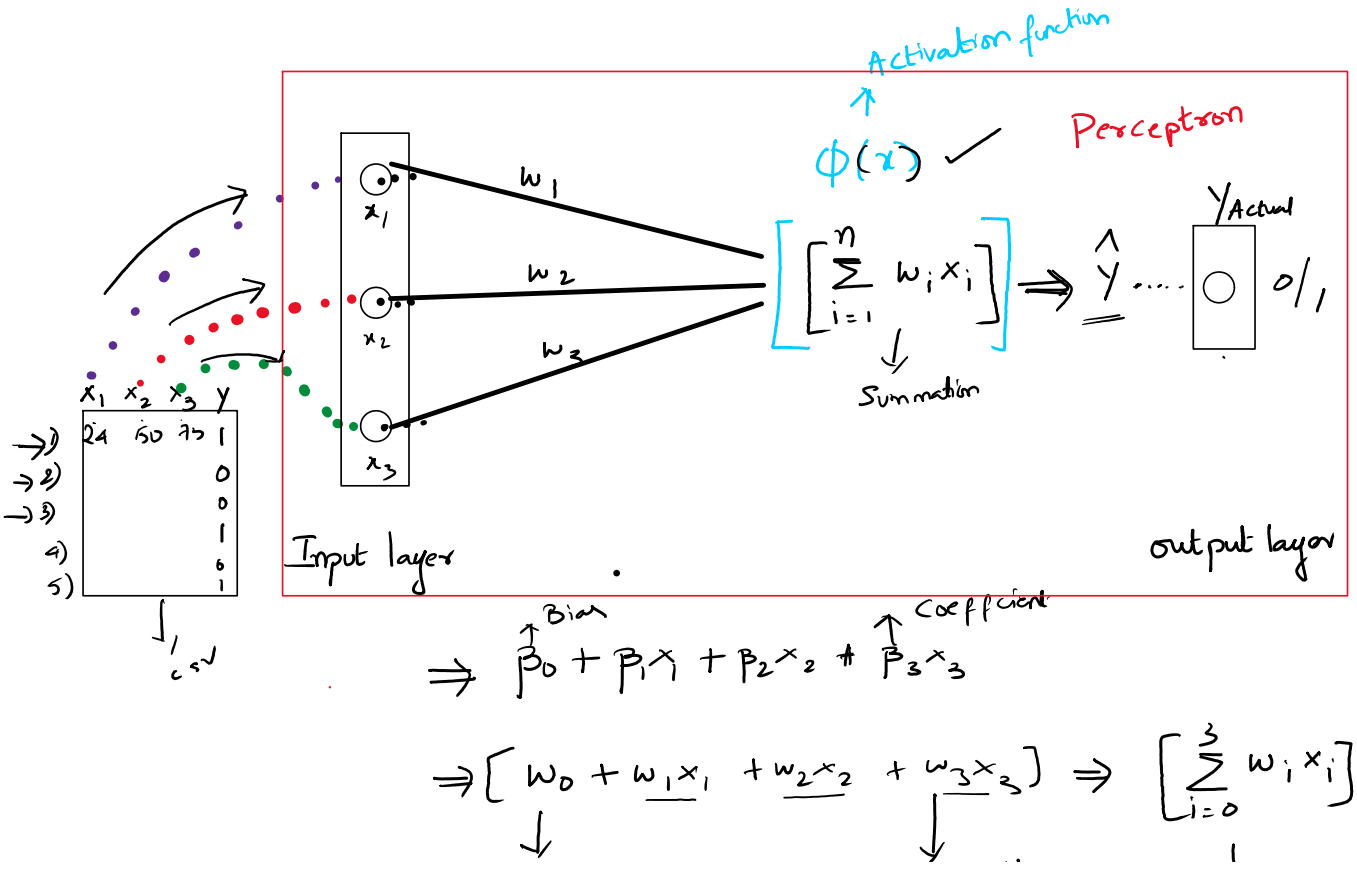
Connect the relationship between X and Y

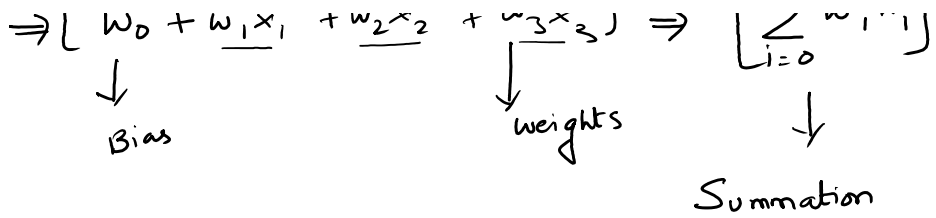
Neural networks:

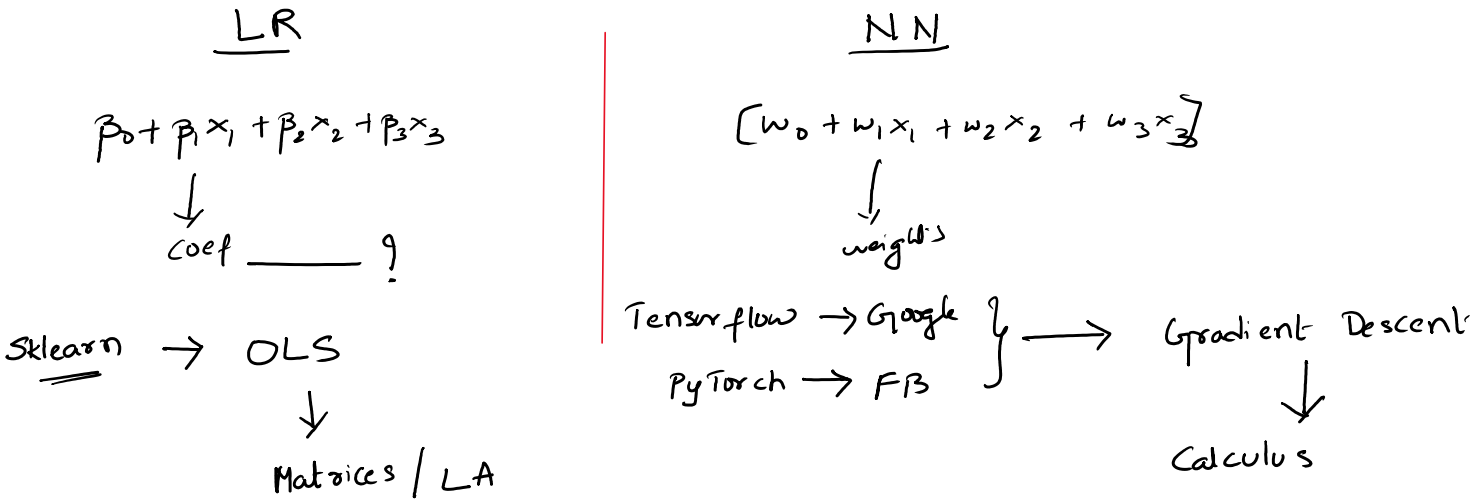
Network?

Neural ---> neurons ---> inspired from the human brains



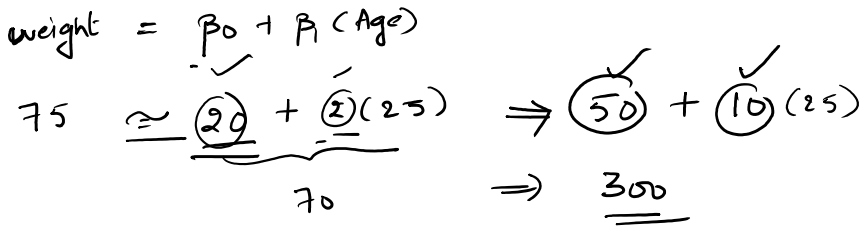






==========

During the perceptron process, the weights are given is not a perfect values at the beginning those are randomly generated.



Our gradient descend method will apply a strategical game such that it will try to identify the best suitable weight values like the first example I have made.

The entire process should be run in a loop will called here as "epochs" and it will identifies the better weight values and replaces over their.

When we are running with the epochs, we are going verify our results.

Regression: R square or root mean square error
Classification: accuracy score or log loss

In between the input layer and output layer:

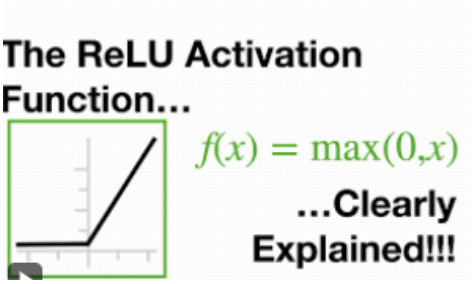
1. Summation
2. Activation function

Activation function: it will manipulate the weighted values associated with X values in such a way that it gives the prediction in favour of Y values.

Depends on the Y variable nature, we are supposed to select activation functions accordingly.

1. Target variable ---> continues ---> regression ---> Relu (Rectified linear unit)
2. Target variable ---> categorical(binary) ---> classification ---> Sigmoid
3. Target variable ---> categorical(multinomial) ---> classification ---> Softmax

What is relu function? How it is works?



weight | sales

→ max (0, -5) → 0

→ max (0, 5) → 5

sigmoid:

$$\sigma(x) = \frac{1}{1+e^{-x}} = \frac{1}{1+\frac{1}{e^x}} = \frac{1}{\frac{e^x+1}{e^x}} = \frac{e^x}{1+e^x} = 0 \text{ to } 1$$

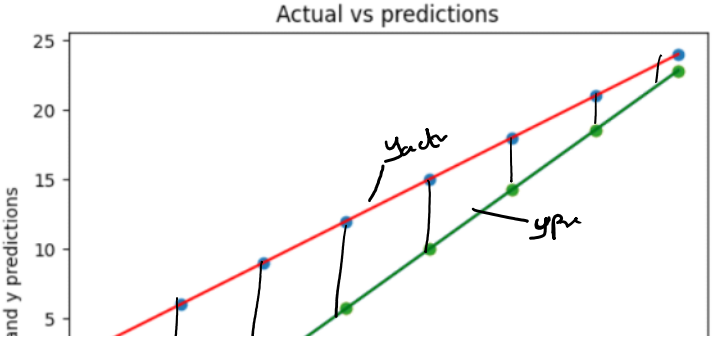
Soft max:

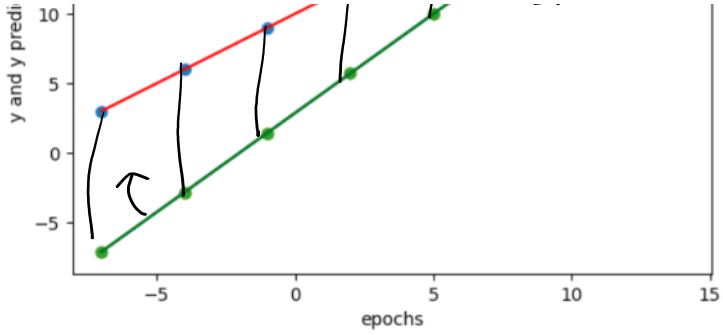
$$softmax(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}}$$

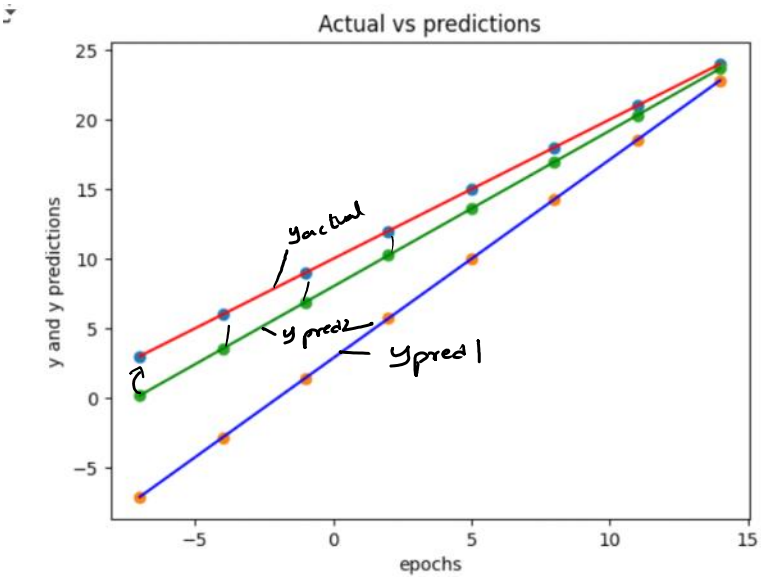
wt		y
20	1)	M
70	2)	H
180	3)	T

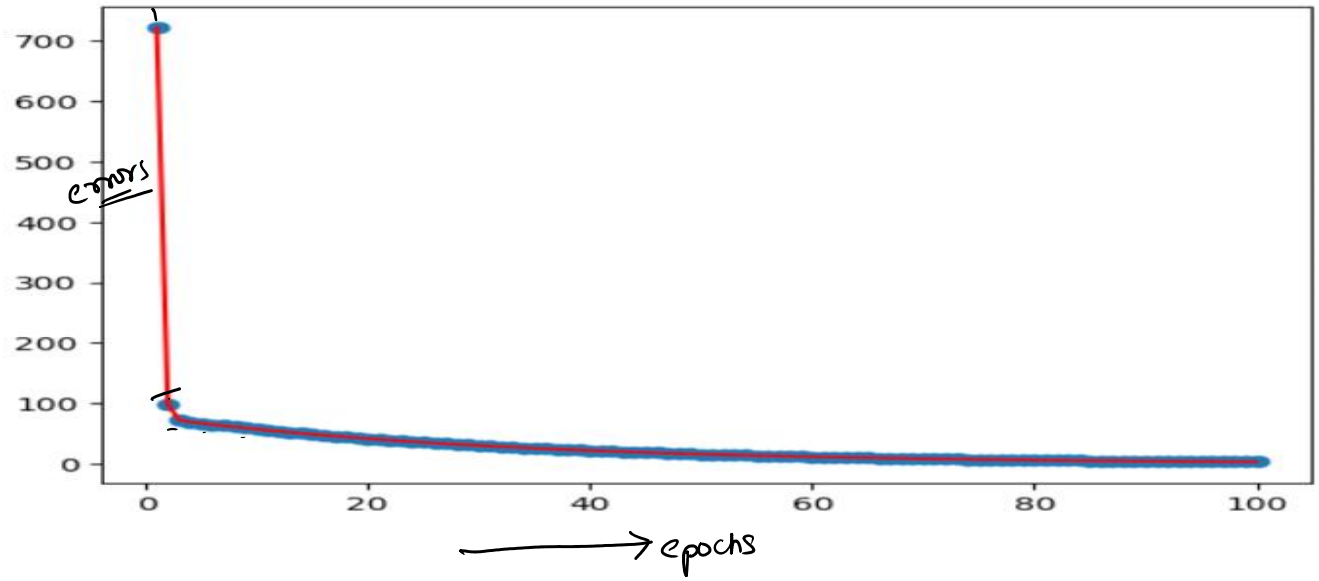
y_M	y_H	y_T	
0.8	0.10	0.10	⇒ Monkey
0.05	0.9	0.05	⇒ Human
0.10	0.15	0.75	⇒ Tiger

==========







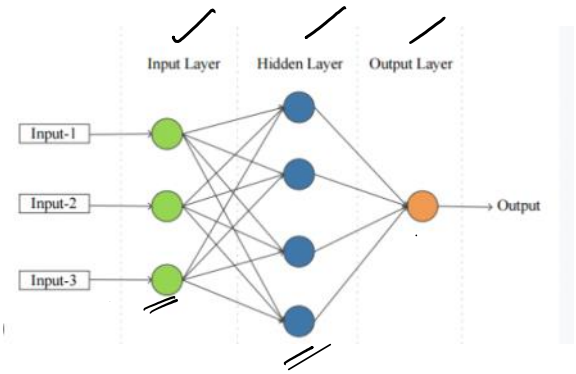


However, our target to reduce the error as much as possible

To reduce the error

1. We can add few more epochs to reduce the error rate

2. We can change activation function and see the results as well.
3. Even though if error is not reduced, we have to insert the hidden layer in between the input layer to output layer



one hidden layer ⟶ more hidden layers