Forward propagation

1 What?

a) forward — move ahead; propagation — related to spreading anything

b) forward propagation means we are moving in only 1 direction, from input to output in a NN.

2 USE → concept used in generally all the NNs.

3 Working → Steps

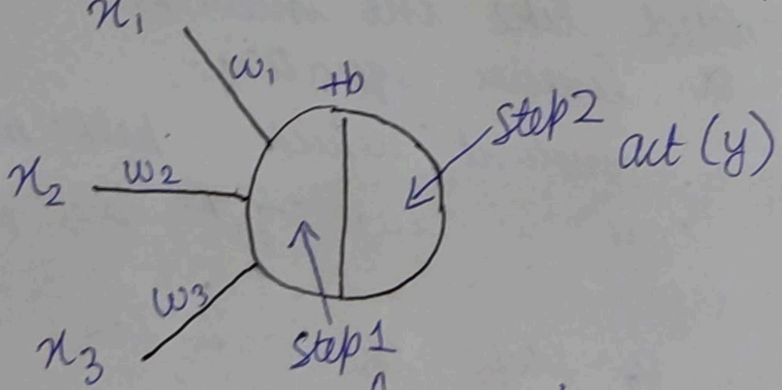
At each hidden neuron or output layer the processing happens in 2 stages

a) Pre-activation

It is weighted Sum of inputs i.e. the linear Transformation of weights w.r.t. inputs available.

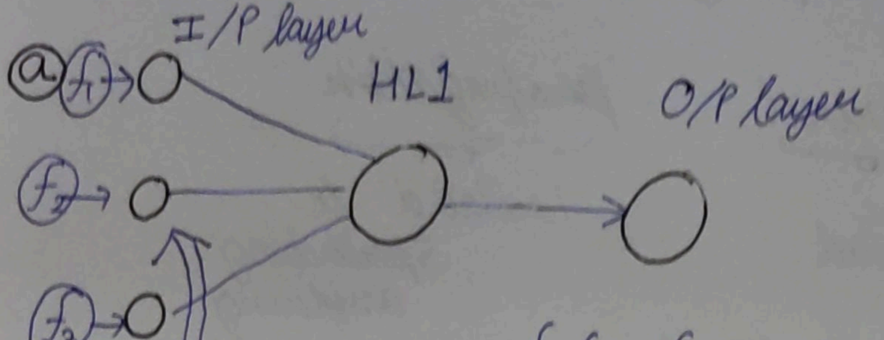
b) Activation

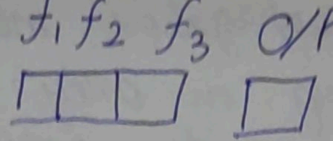
Calculated weighted sum of inputs is passed to the Activation function.

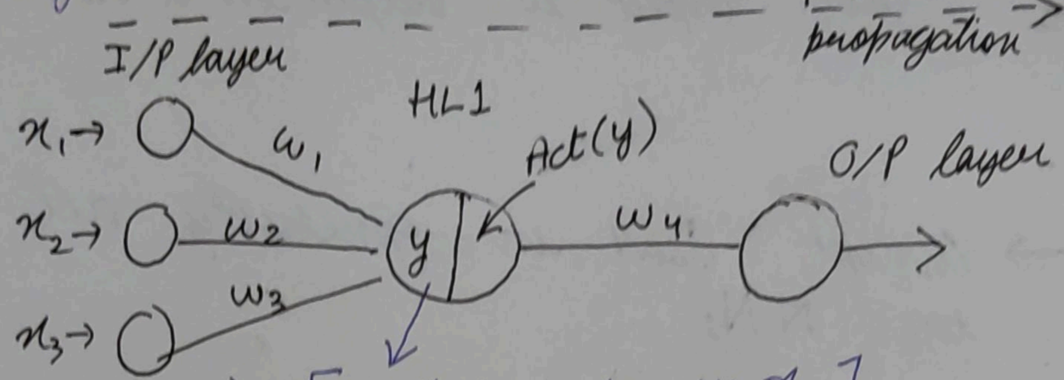


$$y = \sum_{i=1}^{n} x_i w_i + b$$

★ Concept







a) $[w_1x_1 + w_2x_2 + w_3x_3]$

$\Rightarrow \sum_{i=1}^{N} w_ix_i + b_i]$

$\Rightarrow w^T x + b \Rightarrow$ Linear Regression

b) activation function

Q) Why bias added?

Ans $\because$ if weights are 0

$\Rightarrow y = 0$

$\therefore$ Bias is added like the intercept added in a linear equation.

It acts as a constant which helps the model in a way it can fit best for the given data.

Also it allows to shift the activation function to either right or left. (changes in weights simply change gradient / steepness of output)

Q) Dataset :

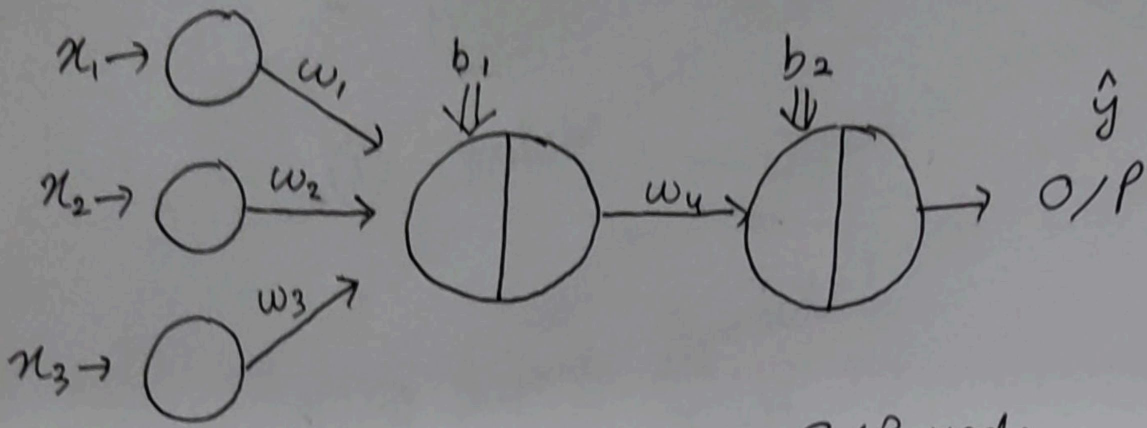
x_1	x_2	x_3	O/P
4	5	6	1

→ Target y^T

Weights :

w_1	w_2	w_3	w_4
0.1	0.2	0.3	0.4

Bias : $b_1 = 0.1$, $b_2 = 0.2$



I/P HL1 O/P Node

Hidden layer 1

Ans: <u>In Hidden layer node</u> : ⓐ $y = \sum_{i=1}^{3} w_i x_i + b_1$

ⓑ $Z = \sigma(y)$

$\Rightarrow$ a) $y = 4 \times 0.1 + 5 \times 0.2 + 6 \times 0.3 + 0.1$
$= 0.4 + 1 + 1.8 + 0.1 = 3.3$

b) Sigmoid func^n : $Z = \sigma(y) = \frac{1}{1+e^{-y}} = 0.96 = Z$

<u>In output layer node</u> ⓐ $Z \times w_4 + b_2 = y'$

ⓑ $\hat{y} = \sigma(Z \times w_4 + b_2)$

$\Rightarrow$ a) $Z \times w_4 + b_2 = y'$

$\Rightarrow y' = 0.96 + 0.4 + 0.2 = 0.584$

b) Sigmoid : $\hat{y} = \sigma(y') = \sigma(Z \times w_4 + b_2 = y')$

$\Rightarrow \hat{y} = 0.64199$

<u>Loss calcula^n</u>

loss func^n $\Rightarrow \frac{1}{2}(y^T - \hat{y})^2$ (MSE) $\Rightarrow \frac{1}{2}(1 - 0.64)^2 = 0.0648$

→ Aim is to reduce this loss

→ No use Back propagaⁿ till $\hat{y} \approx$ Target y $[y^T]$