1/ Q a system with the following logical expression ①

$f(x_1, x_2, x_3, x_4) = \sum m(3,5,6,7,8,9,10,11,12,13,14,15)$

Then the realization of this expression using Threshold logic model yields.

ⓐ

ⓑ

ⓒ

ⓓ

ans

x_1

$x_3x_4 \backslash x_1x_2$	00	01	11	10
00	[0]	[4]	[12] 1	[8] 1
01	[1]	[5] 1	[13] 1	[9] 1
11	[3] 1	[7] 1	[15] 1	[11] 1
10	[2]	[6] 1	[14] 1	[10] 1

x_4 (rows 01, 11) x_3 (rows 11, 10) x_2 (columns 01, 11)

$b_0 = (\sum f(x)=1) - (\sum f(x)=0) = 12 - 4 = 8$

$b_1 = 2(8-4) = 8$

$b_2 = 2(7-5) = 4$

$b_3 = 2(7-5) = 4$

$b_4 = 2(7-5) = 4$

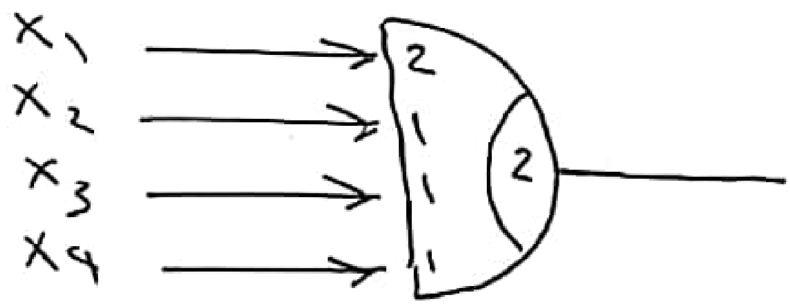
b_0	b_1	b_2	b_3	b_4
8	8	4	4	4
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
2	2	1	1	1
a_0	a_1	a_2	a_3	a_4

$$T = \frac{1}{2}\left[\sum_{i=1}^{n} a_i - a_0 + 1\right]$$

$$= \frac{1}{2}\left[a_1 + a_2 + a_3 + a_4 - a_0 + 1\right]$$

$$= \frac{1}{2}\left[2 + 1 + 1 + 1 - 2 + 1\right]$$

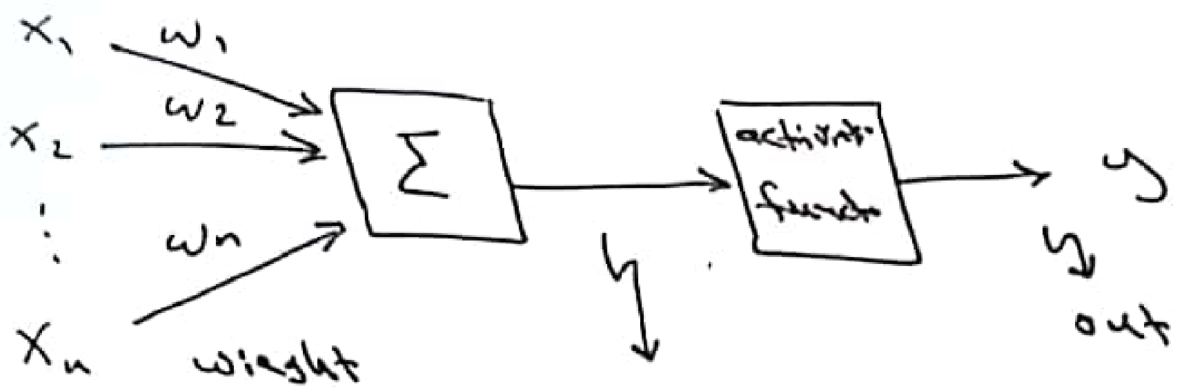
$$= 2$$



$$\langle 2X_1 + X_2 + X_3 + X_4 \rangle 2$$

lecture -4-

* The Mcculloch - pitts neuron



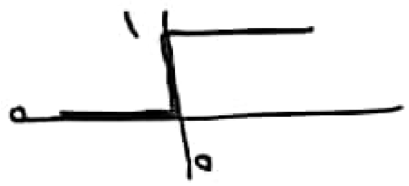
input weight

$$net = \sum w_i x_i = w_1x_1 + w_2x_2 + \dots w_nx_n$$

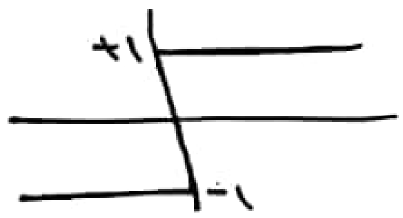
$$y = f(net)$$

Some useful function

$\Rightarrow sgn(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$
unipolar



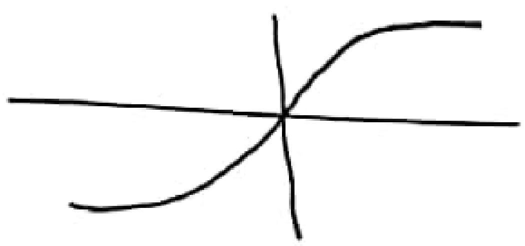
$\Rightarrow sgn(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases}$
bipolar



→ hard limit "binary"

$\Rightarrow Sigmoid = \frac{1}{1+e^{-x}}$
unipolar

$\Rightarrow Sigmoid = \frac{2}{1+e^{-x}} - 1$
bipolar



→ soft limit "contiuous"

hard limit
" binary "
sgn (net)

softlimit
" contineous "

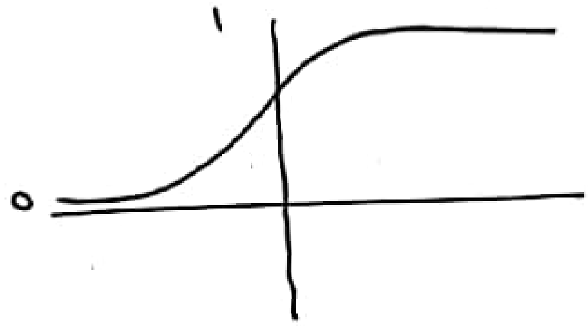
* unipolar

$$f(net) = \begin{cases} 1 & net \geq 0 \\ 0 & net < 0 \end{cases}$$

θ Threshuld

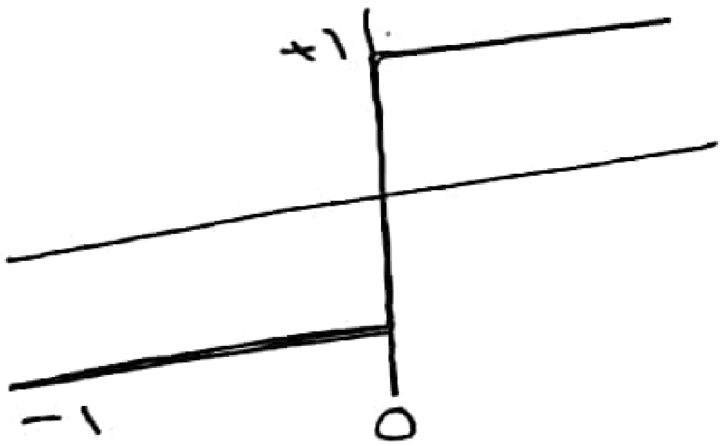
* unipolar

$$f(net) = \frac{1}{1 + e^{-\lambda(net)}}$$



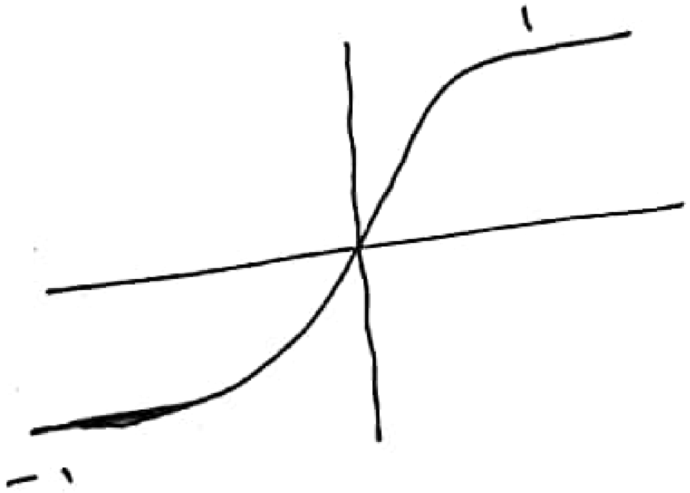
* bipolar

$$f(net) = \begin{cases} 1 & net \geq 0 \\ -1 & net < 0 \end{cases}$$

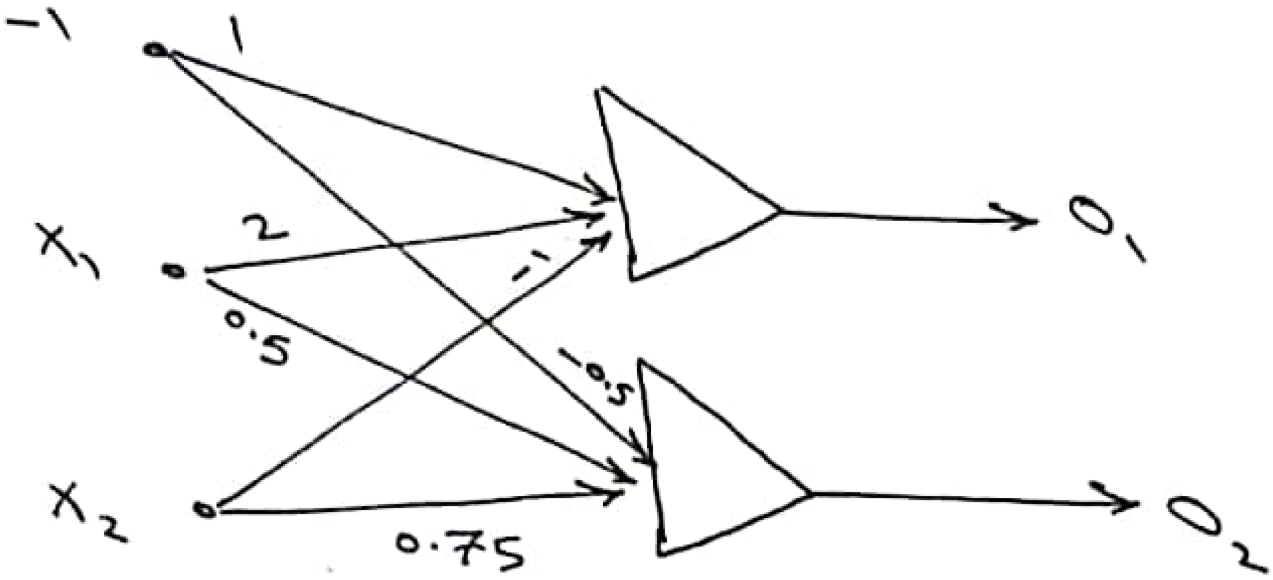


* bipolar

$$f(net) = \frac{2}{1 + e^{-\lambda(net)}} - 1$$



... shown below



use neuron with a bipolar continous activation function with $\lambda = 1$
if The neuron output measure as
$O_1 = 0.2$, $O_2 = -0.8$ find The input
$X = [X_1, X_2]$

ans

$net_1 = -1 \times 1 + 2X_1 + (-X_2)$

$net_1 = 2X_1 - X_2 - 1$

$net_2 = -1 \times -0.5 + 0.5X_1 + 0.75X_2$

$net_2 = 0.5X_1 + 0.75X_2 + 0.5$

$\because$ continouse bipolar

$$O = f(net) = \frac{2}{1+e^{-\lambda net}} - 1 = \frac{2}{1+e^{-net}} - 1$$

$$O_1 = \frac{2}{1+e^{-net_1}} - 1$$

$$\therefore \frac{2}{1+e^{-(net_1)}} - 1 = 0.2$$

$$\frac{2}{1+e^{-net_1}} = 1.2$$

$$\frac{1}{1+e^{-net_1}} = 0.6$$

$$1+e^{-net_1} = 1.667$$

$$e^{-net_1} = 0.667$$

$$-net_1 = \ln 0.667$$

$$\cancel{-}net_1 = \cancel{-}0.405$$

$$net_1 = 0.405$$

$$\therefore 2X_1 - X_2 - 1 = 0.405$$

$$2X_1 - X_2 = 1.405 \quad —①$$

$$O_2 = \frac{2}{1+e^{-net_2}} - 1$$

$$\frac{2}{1+e^{-net_2}} - 1 = -0.8$$

$$\frac{2}{1+e^{-net_2}} = 0.2$$

$$\frac{1}{1+e^{-net_2}} = 0.1$$

$$1+e^{-net_2} = 10$$

$$e^{-net_2} = 9$$

$$-net_2 = \ln 9$$

$$-net_2 = 2.197$$

$$net_2 = -2.197$$

$$0.5X_1 + 0.75X_2 + 0.5 = -2.197$$

$$0.5X_1 + 0.75X_2 = -2.697 \quad *4$$

$$2X_1 + 3X_2 = -10.788 \quad —②$$

$$\therefore 2X_1 - X_2 = 1.405$$

$$- \quad 2X_1 + 3X_2 = -10.788$$

$$-4X_2 = 12.193$$

$$\therefore X_2 = \frac{12.193}{-4}$$

$$\boxed{X_2 = -3.048}$$

$$2X_1 - (-3.048) = 1.405$$

$$2X_1 = 1.405 - 3.048$$

$$\boxed{X_1 = -0.8215}$$

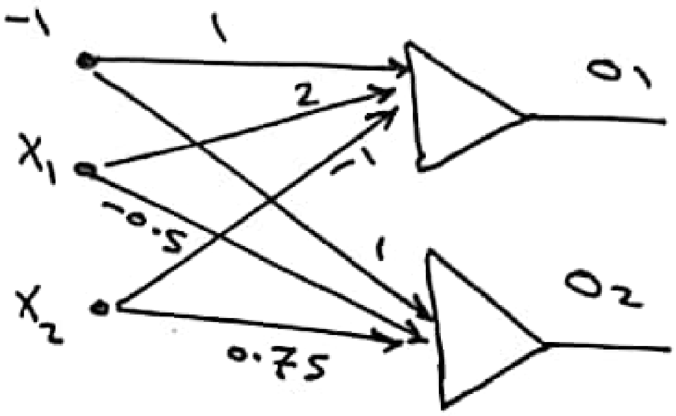
Q The neurons outputs has been measured as $O_1 = 0.2$ and $O_2 = 0.9$ for the network shown in fig 1 which uses bipolar continuous activation function with $\lambda = 0.5$ then the value of input vector $X = [X_1, X_2]$ is

ⓐ $X_1 = 6.3$ $X_2 = 4$

ⓑ $X_1 = 2.1$ $X_2 = 7.9$

ⓒ $X_1 = 5$ $X_2 = 8.6$

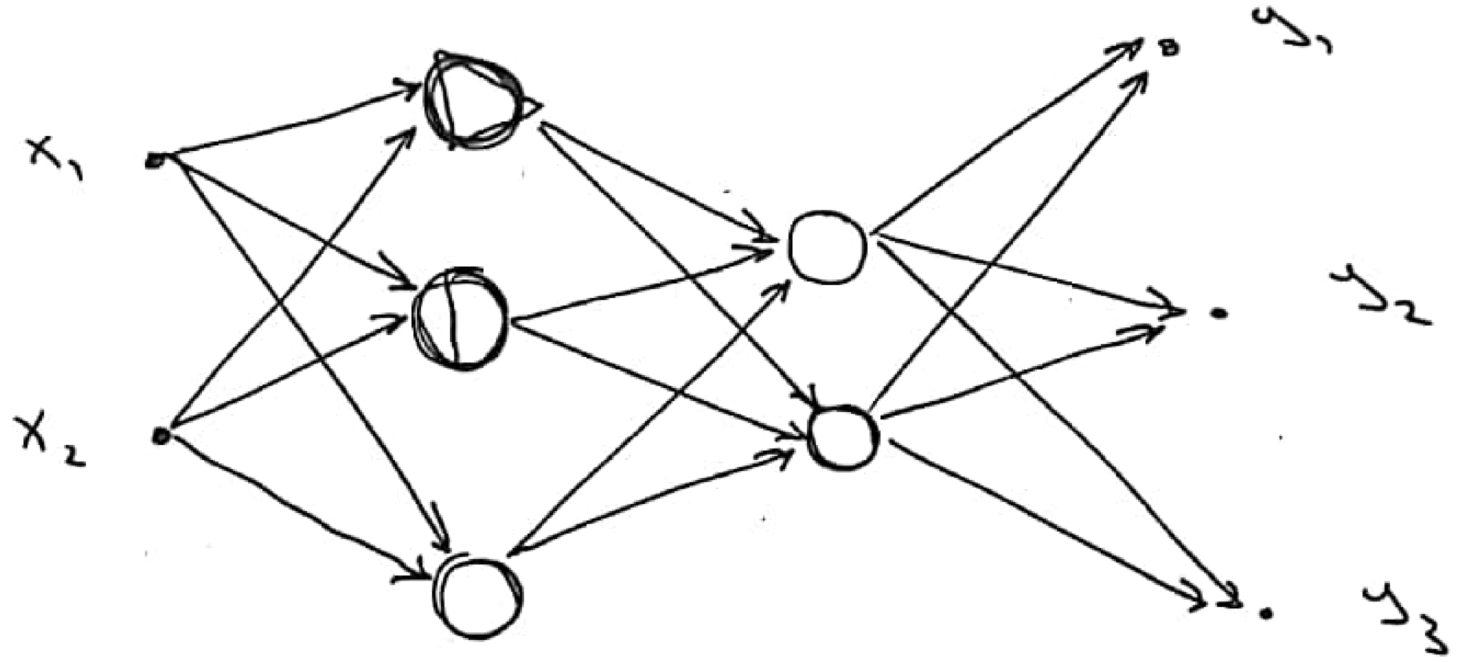
ⓓ $X_1 = 3$ $X_2 = 1.2$



in general "multi layer"

input hidden layer 1 hidden layer 2 .... output

if input is 2
hidden layer 1 3
hidden layer 2 2
output 3



input	hidden layer 1	hidden layer 2	output
2×3=6	3×2=6		2×3=6
6 weight	6 weight		6 weight

$\therefore$ 6 + 6 + 6 = 18 weight

Q: a multi layer feed forward network has 5 input unit, a first hidden layer with 4 unit, a second hidden layer with 3 units, and 2 output unit how many weights does this network have?

ans:

input	hidden layer 1	hidden layer 2	output
5	4	3	2

$5 \times 4 = 20$ $\quad$ $4 \times 3 = 12$ $\quad$ $3 \times 2 = 6$

$\therefore$ no. of Weight $= 20 + 12 + 6 = 38$

(a) 14

(b) 20

(c) 26

(d) 18

(e) 38

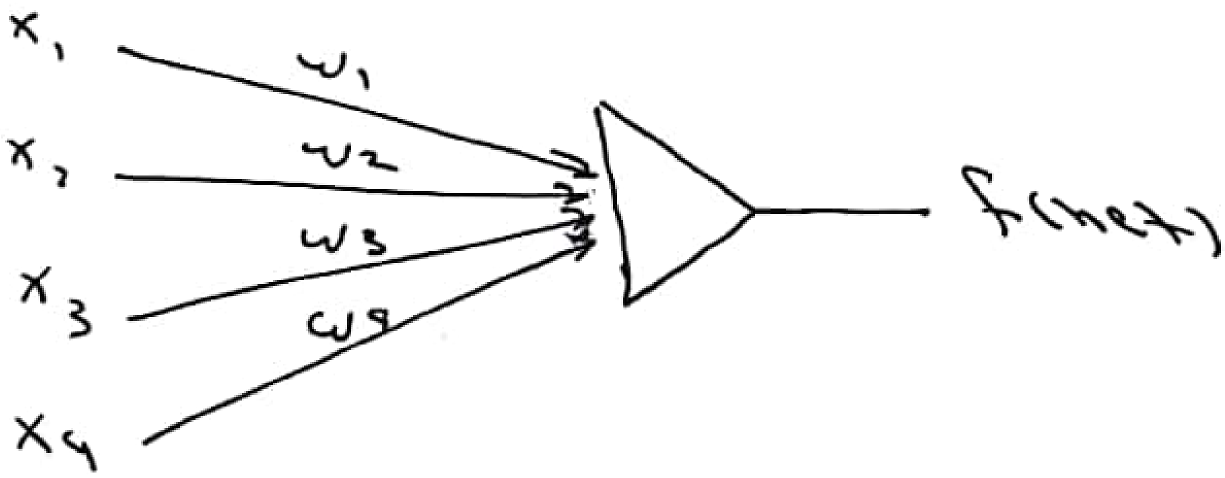
Neural network learning rules

* Hebbian rules

$\left.\begin{matrix}\alpha \\ \eta\end{matrix}\right\}$ learning rate

Ex Use hebbian rules with binary activation functions with initial weight vector $W^1 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix}$ needs to trained using set of three input vector as below

$$X_1 = \begin{bmatrix} 1 \\ -2 \\ 1.5 \\ 0 \end{bmatrix} \qquad X_2 = \begin{bmatrix} 1 \\ -0.5 \\ -2 \\ -1.5 \end{bmatrix} \qquad X_3 = \begin{bmatrix} 0 \\ 1 \\ -1 \\ 1.5 \end{bmatrix}$$



for arbitrary choice of learning constant $\alpha = 1$, and assume a binary bipolar activation function

...s $W^1 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix}$ $X_1 = \begin{bmatrix} 1 \\ -2 \\ 1.5 \\ 0 \end{bmatrix}$

step 1

$$net^1 = W^{1t} X_1 = \begin{bmatrix} 1 & -1 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} 1 \\ -2 \\ 1.5 \\ 0 \end{bmatrix} =$$

$$= 1 \times 1 + (-1)(-2) + (0)(1.5) + (0.5)(0)$$

$$= 3 \geq 0$$

$$\therefore f(net^1) = sgn(net^1) = sgn(3) = 1$$

$$\boxed{W^2 = W^1 + \alpha f(net^1) X_1}$$

وزن محدث — وزن قديم — rate — output قديم — input قديم

$$W^2 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix} + 1(1) \begin{bmatrix} 1 \\ -2 \\ 1.5 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ -3 \\ 1.5 \\ 0.5 \end{bmatrix}$$

step 2 $W^2 = \begin{bmatrix} 2 \\ -3 \\ 1.5 \\ 0.5 \end{bmatrix}$ $X_2 = \begin{bmatrix} 1 \\ -0.5 \\ -2 \\ -1.5 \end{bmatrix}$

$$net^2 = W^{2t} X_2 = \begin{bmatrix} 2 & -3 & 1.5 & 0.5 \end{bmatrix} \begin{bmatrix} 1 \\ -0.5 \\ -2 \\ -1.5 \end{bmatrix} = -0.25 < 0$$

$$f(net^2) = sgn(net^2) = -1$$

$$W^3 = \begin{bmatrix} 2 \\ -3 \\ 1.5 \\ 0.5 \end{bmatrix} + 1(-1) \begin{bmatrix} 1 \\ -0.5 \\ -2 \\ -1.5 \end{bmatrix} = \begin{bmatrix} 2 \\ -3 \\ 1.5 \\ 0.5 \end{bmatrix} - \begin{bmatrix} 1 \\ -0.5 \\ -2 \\ -1.5 \end{bmatrix} = \begin{bmatrix} 1 \\ -2.5 \\ 3.5 \\ 2 \end{bmatrix}$$

step 3 عليكم

.f same as perivouse Example but continous bipolar activation

ans

$$W^1 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix} \quad X_1 = \begin{bmatrix} 1 \\ -2 \\ 1.5 \\ 0 \end{bmatrix} \quad \alpha = 1, \ \text{let } \lambda = 1$$

$$f(net) = \frac{2}{1+e^{-\lambda net}} - 1 = \frac{2}{1+e^{-net}} - 1$$

$$net^1 = W^{1t} X_1 = [1 \ -1 \ 0 \ 0.5] \begin{bmatrix} 1 \\ -2 \\ 1.5 \\ 0 \end{bmatrix} = 3$$

$$f(net^1) = \frac{2}{1+e^{-3}} - 1 = 0.905$$

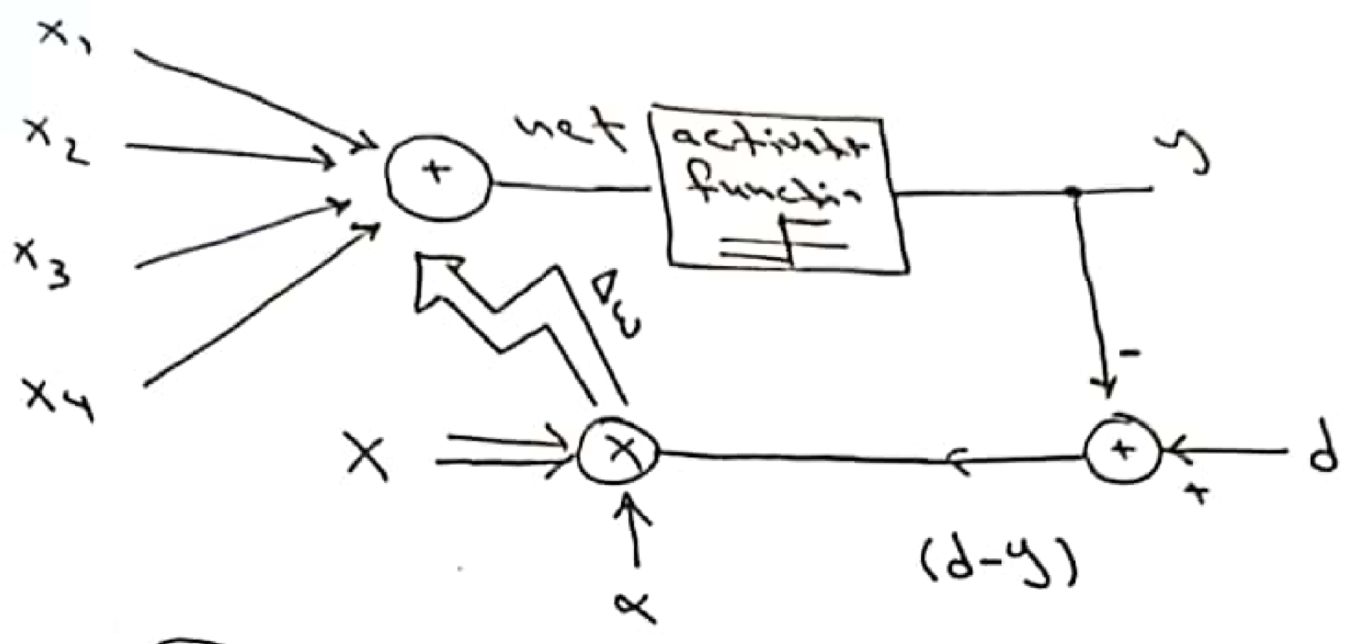
$$W^2 = W^1 + \alpha f(net^1) X_1$$

$$W^2 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix} + (1)(0.905) \begin{bmatrix} 1 \\ -2 \\ 1.5 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix} + \begin{bmatrix} 0.905 \\ -1.81 \\ 1.3575 \\ 0 \end{bmatrix} = \begin{bmatrix} 1.905 \\ -2.81 \\ 1.357 \\ 0.5 \end{bmatrix}$$

and so on.

perceptron learning Rule



$$\underbrace{W^2}_{W_{new}} = \underbrace{W^1}_{w_{old}} + \underbrace{\alpha\,(d^1 - f(net^1))\,X_1}_{\Delta w}$$

W_{new} w_{old} + Δw

Ex perceptron learning rule

$$X_1 = \begin{bmatrix} 1 \\ -2 \\ 0 \\ -1 \end{bmatrix} \quad X_2 = \begin{bmatrix} 0 \\ 1.5 \\ -0.5 \\ -1 \end{bmatrix} \quad X_3 = \begin{bmatrix} -1 \\ 1 \\ 0.5 \\ -1 \end{bmatrix}$$

initial weight $W^t = [1 \; -1 \; 0 \; 0.5]$

$\alpha = 0.1$

$d_1 = -1$, $d_2 = -1$, $d_3 = 1$

use binary bipolar activation function

step 1

$$W^1 = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix} \qquad X_1 = \begin{bmatrix} 1 \\ -2 \\ 0 \\ -1 \end{bmatrix} \qquad d_1 = -1$$

$$net^1 = W^{1t} X_1 = \begin{bmatrix} 1 & -1 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} 1 \\ -2 \\ 0 \\ -1 \end{bmatrix} = 1 + 2 + 0 - 0.5 = +2.5 > 0$$

$$\therefore f(net^1) = sgn(+2.5) = +1$$

$$W^2 = W^1 + \alpha (d_1 - f(net^1)) X_1$$

$$= \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix} + 0.1(-1 - (+1)) \begin{bmatrix} 1 \\ -2 \\ 0 \\ -1 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix} - 0.2 \begin{bmatrix} 1 \\ -2 \\ 0 \\ -1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0.5 \end{bmatrix} - \begin{bmatrix} 0.2 \\ -0.4 \\ 0 \\ -0.2 \end{bmatrix} = \begin{bmatrix} 0.8 \\ -0.6 \\ 0 \\ 0.7 \end{bmatrix}$$

step 2

$$X_2 = \begin{bmatrix} 0 \\ 1.5 \\ -0.5 \\ -1 \end{bmatrix} \qquad d_2 = -1$$

$$net^2 = W^{2t} X_2 = \begin{bmatrix} 0.8 & -0.6 & 0 & 0.7 \end{bmatrix} \begin{bmatrix} 0 \\ 1.5 \\ -0.5 \\ -1 \end{bmatrix} = -1.6 < 0$$

$$f(net^2) = sgn(net^2) = sgn(-1.6) = -1$$

$$W^3 = W^2 + \alpha (d_2 - f(net^2)) X_2$$

$$= \begin{bmatrix} 0.8 \\ -0.6 \\ 0 \\ 0.7 \end{bmatrix} + \underbrace{0.1(-1 - (-1)) \begin{bmatrix} 0 \\ 1.5 \\ -0.5 \\ -1 \end{bmatrix}}_{=0} = \begin{bmatrix} 0.8 \\ -0.6 \\ 0 \\ 0.7 \end{bmatrix}$$

step 3 عليكم

perceptron with a unipolar step function (15) has two inputs with weights $w_1 = 0.5$ $w_2 = -0.2$ and a threshold $\theta = 0.3$ is trained using perceptron learing rule. what are the new values of the weights and threshold after one step of training with the input vector $X = [0 \;\; 1]^T$ and desired output 1, using a learing rate $\eta = 0.5$?

(a) $w_1 = 1.0$; $w_2 = -0.2$; $\theta = -0.2$

(b) $w_2 = 0.5$; $w_2 = 0.3$; $\theta = -0.2$

(c) $w_3 = 0.5$; $w_2 = 0.3$; $\theta = 0.7$

(d) $w_4 = 0.5$; $w_2 = -0.3$; $\theta = 0.2$

ans

$W^{1t} = [0.5 \;\; -0.2]$ $\quad X_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ $\quad \alpha = 0.5$

$d_1 = 1$

$\theta_1 = 0.3$

unipolar step

$$W^2 = W^1 + \alpha(d_1 - f(net^1))X_1$$
$$\theta^2 = \theta^1 + \alpha(d_1 - f(net^1))$$

Preceptron

$net^1 = W^{1t} X_1 = [0.5 \;\; -0.2]\begin{bmatrix} 0 \\ 1 \end{bmatrix} = -0.2 < 0.3$

$\therefore f(net^1) = sgn(net^1) = sgn(-0.2) = 0$

$$\omega^2 = \begin{bmatrix} 0.5 \\ -0.2 \end{bmatrix} + 0.5(1-0)\begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0.5 \\ -0.2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}$$

$$\omega^2 = \begin{bmatrix} 0.5 \\ 0.3 \end{bmatrix} \qquad \therefore \ \omega_1 = 0.5 \qquad \omega_2 = 0.3$$

$$\theta^2 = 0.3 + 0.5(1-0)$$

$$= 0.3 + 0.5$$

$$\theta^2 = 0.8$$

$\therefore$ c

هو الاقرب للحل