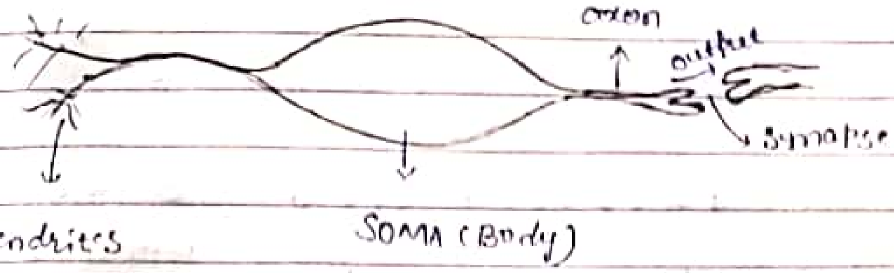
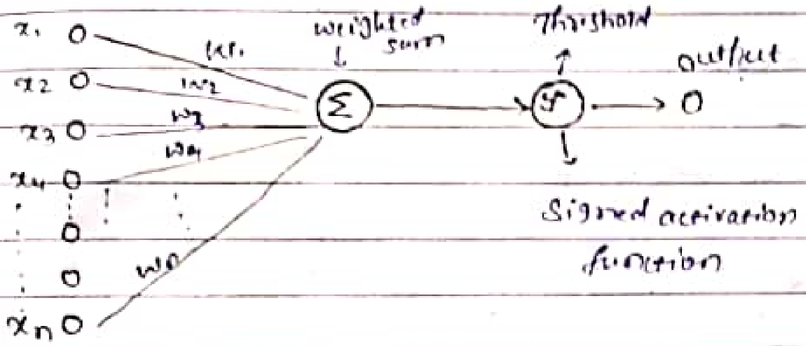
Unit-3

Biological Neuron :





$$\begin{cases} 1 & \text{if } \sum_{i=0}^{n} x_i w_i > 0 \\ -1 & \text{otherwise} \end{cases}$$

1) Step Function
2) Linear Function
3) Sigmoid function

⇒ Perspetron training rule → to adjust the weights.

Perspetron Training Rule:

$$\boxed{W_i = W_i + \alpha \times e \times x_i}$$

(New weight) ; α → Learning rate

e → error rate

$$\boxed{e = (t - o)}$$

t → Target ; o → output

AND gate (Step function)

A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

$W_1 = 1.2$, $W_2 = 0.6$, threshold $= 1$ & (learning rate) $\alpha = 0.5$

i) $A = 0$; $B = 0$ $T = 0$ ($\because$ from Truth Table).

$$\sum_{i=0}^{n} w_i x_i = (W_1 \times x_1 \Rightarrow) 1.2 \times 0 + 0.6 \times 0 = 0 < 1$$

Output $= 0$

(ii) $A = 0$ $B = 1$ $T = 0$

$\sum w_i x_i = 1.2 \times 0 + 0.6 \times 1$

$= 0.6 < 1$

Output $= 0$

(iii) $A = 1$, $B = 0$ $T = 0$

$\sum w_i x_i = 1 \times 1.2 + 0.6 \times 0$

$= 1.2 \not< 1$

Output $= 1$

$W_i = W_i + \alpha \times (t - o) \times x_i$

$= 1.2 + 0.5 \times (0 - 1) \times 1$

$= 1.2 - 0.5$

$\boxed{W_1 = 0.7}$

$W_2 = 0.6 + 0.5\,(0 - 1)\,0 = \boxed{0.6}$

Now, $W_1 = 0.7$, $W_2 = 0.6$ Threshold $= 1$ & $\alpha = 0.5$

(i) $A = 0$, $B = 0$ $T = 0$

$\sum w_i x_i = 0.7 \times 0 + 0 \times 0.6 = 0 < 1$

Output 0

(ii) $A = 0$, $B = 1$ $T = 0$

$\sum w_i x_i = 0 \times 0.7 + 1 \times 0.6 = 0.6 < 1$

Output 0

HLW1 — $w_1 = -0.2$ & $w_3 = 0.4$ $t = 0$ & $a = 0.2$
using OR gate

iii) $A = 1$. $B = 0$ $T = 0$

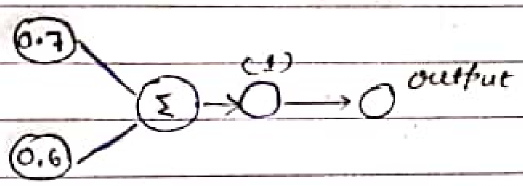
$\Sigma w_i x_i = 1 \times 0.7 + 0 \times 0.6$
$= 0.7 < 1 \rightarrow$ Threshold

Output 0

iv) $A = 1$, $B = 1$ $T = 1$

$\Sigma w_i x_i = 1 \times 0.7 + 1 \times 0.6$
$= 0.7 + 0.6$
$= 1.3 \nless 1$

output 1



OR gate

A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

$w_1 : 0.6$, $w_2 = 0.6$, $t = 1$ & $a = 0.5$

i) $A = 0$, $B = 0$, $T = 0$

$\Sigma w_i x_i = 0.6 \times 0 + 0.6 \times 0 = 0 < 1$
$\therefore$ O/P = 0

ii) $A = 0$ & $B = 1$, $T = 1$

$\Sigma w_i x_i = 0.6 \times 0 + 0.6 \times 1$
$= 0.6 < 1$
$\therefore$ O/P = 0

$W_1 = 0.6 + 0.5(1-0) \times 0$

$= 0.6$

$W_2 = 0.6 + 0.5(1-0) \times 1$

$= 0.6 + 0.5$

$W_2 = 1.1$

Now, $\underline{W_1 = 0.6}$ & $\underline{W_2 = 1.1}$

(i) $A=0$, $B=0$ $T=0$

$\Sigma w_i x_i = 0.6 \times 0 + 1.1 \times 0 = 0$

$\therefore$ O/p = 0

(ii) $A=0$, $B=1$, $T=1$

$\Sigma w_i x_i = 0.6 \times 0 + 1 \times 1.1$

$= 1.1 > 1$

$\therefore$ O/p = $\underline{1}$

(iii) $A=1$, $B=0$, $T=1$

$\Sigma w_i x_i = 0.6 \times 1 + 0 \times 1.1$

$= 0.6 < 1$

$\therefore$ O/p = 0

$W_1 = 0.6 + 0.5 \times (1-0) \times 1$

$= 0.6 + 0.5$

$W_1 = \underline{1.1}$

$\therefore W_2 = 1.1 + 0.5(0-0) \times 0$

$W_2 = \underline{1.1}$

Now, $\underline{W_1 = 1.1}$ & $\underline{W_2 = 1.1}$

(i) $A:0$, $B:0$, $T:0$

$\Sigma W_i X_i = 1.1 \times 0 + 1.1 \times 0 = 0 < 1$

$\therefore$ O/p : $\underline{0}$

Learning rate a range 0 to 1.

(ii) A = 0 , B = 1   T = 1,

$\Sigma W_i X_i = 0 \times 1.1 + 1 \times 1.1$

$= 1.1 > 1$

$\therefore$ O/p = 1,

iii) A = 1 , B = 0   T = 1

$\Sigma W_i X_i = 1 \times 1.1 + 0 \times 1.1$

$= 1.1 > 1$

$\therefore$ O/p = 1

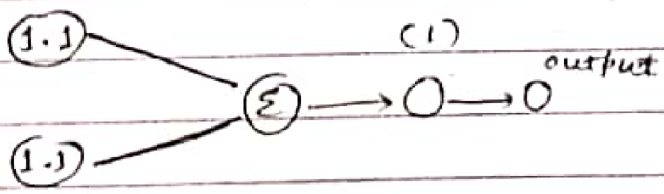
(iv) A = 1 , B = 1 , T = 1

$\Sigma W_i X_i = 1 \times 1.1 + 1 \times 1.1$

$= 1.1 + 1.1$

$= 2.2 > 1$

$\therefore$ O/p = 1



NOT Gate :

A	$\bar{A}$
0	1
1	0

$$X = \sum_{i=1}^{n} w_i x_i$$

$$y = \begin{cases} +1 & \text{for } X > t \text{ (→ Threshold)} \\ 0 & \text{for } X \leq t \end{cases}$$

$$Step(x) = \sum_{i=1}^{n} w_i x_i$$

$$Step(Y) = Step \begin{cases} +1 & \text{for } X > t \\ 0 & \text{for } X \leq t \end{cases}$$

$$y = Step\left(\sum_{i=1}^{n} w_i x_i\right)$$

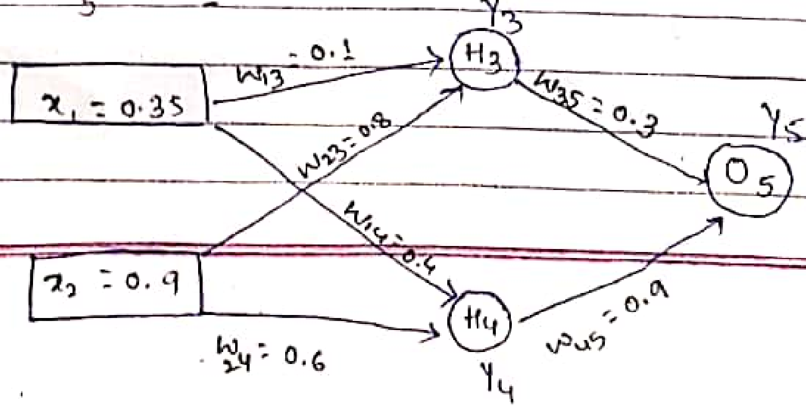
Multi-layer Neural networks:

→ Most real-world problems are not linearly seperable. So all the perception have interesting model something more powerful is needed. Thus we shall learnt about the multi-layer neural networks.

→ Activation f^n → Sigmoid function.

$$\boxed{\sigma(x) = \frac{1}{1 + e^{-x}}}$$

Q) using Sigmoid activation f^n forward pass is being performed & backward pass on the network. Assume that actual o/p of Y is 0.5 and learning rate 1



0.755

ii) $A_j = \sum_j (W_{ij} \cdot x_i)$

$A_1 = 0.1 \times 0.35 + 0.8 \times 0.9$ $\quad (x_1 \times w_{13}) + (x_2 \times w_{23})$

$= 0.035 + 0.72$

$A_1' = 0.755 = X_3$

$Y_3 = \frac{1}{1+e^{-x_3}} = \frac{1}{1+e^{-0.75}} = 0.679$

~~Ans~~

$A_2 = 0.4 \times 0.35 + 0.6 \times 0.9$

$= 0.14 + 0.54$

$A_2 = 0.68 = X_4$

$Y_4 = \frac{1}{1+e^{-x_4}} = \frac{1}{1+e^{-0.68}} = 0.663$

$A_3 = 0.679 \times 0.3 + 0.68 \times 0.9$

$= 0.2037 + 0.612$

$A_3 = 0.8157 = X_5$

$Y_5 = \frac{1}{1+e^{-.75}} = \frac{1}{1+e^{-0.8157}} = 0.69$

Actual o/p - Obtained O/P

$0.5 - 0.69 = -0.19$ (Error rate)

→ Common method to improve performance of backpropagation is include

Momentum. $\Delta w_{ij}t = \alpha . x_i . \delta_i + \beta \Delta w_{ij} (t-1)$

(bias) $\Delta w_{jk}(t) = \alpha . y_j . \delta_j + \beta \Delta w_{jk} (t-1)$

→ An alternate method of speeding up backpropagation is used hyper-bolic tangent function.

$$\boxed{\tanh(x) = \frac{2a}{1+e^{-bx}} - a}$$

→ Last method is adjust the learning rate.

Recurrent Networks:

Field Network - invented by John (1989)

→ Sign activation function

$$Sign(x) = \begin{cases} +1 & \text{for } x \geq 0 \\ -1 & \text{for } x < 1 \end{cases}$$

$$\boxed{W = \sum_{i=1}^{N} X_i X_i^t - NI}$$

where X_i → input

X_i^t → Transpose input

I → Identity matrix

N → No. of inputs, vectors

Q)

$$x_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}_{5\times1} \quad x_2 = \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} \quad x_3 = \begin{bmatrix} 1 \\ -1 \\ 1 \\ 1 \\ -1 \end{bmatrix}$$

soln)

$N = 3.$

$x_1^t = [1 \quad 1 \quad 1 \quad 1 \quad 1]_{1\times5}$

$x_2^t = [-1 \quad -1 \quad -1 \quad -1 \quad -1]$

$x_3^t = [1 \quad -1 \quad 1 \quad 1 \quad -1]$

$$W = \sum_{i=1}^{N} x_i x_i^t - NI$$

$$W = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \cdot [1 \quad 1 \quad 1 \quad 1 \quad 1] + \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} \cdot [-1 \quad -1 \quad -1 \quad -1 \quad -1] + \begin{bmatrix} 1 \\ -1 \\ 1 \\ 1 \\ -1 \end{bmatrix} \cdot [1 \quad -1 \quad 1 \quad 1 \quad -1]$$

$$- 3 \cdot \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$W: \cancel{5} + \cancel{5} + \cancel{5} - \begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{bmatrix}$$

~~W=~~

Scanned with CamScanner

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} + \begin{bmatrix} 1 & -1 & 1 & 1 & -1 \\ -1 & +1 & -1 & -1 & +1 \\ 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 & -1 \\ -1 & +1 & -1 & -1 & +1 \end{bmatrix}$$

$$- \begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{bmatrix}$$

$$W = \begin{bmatrix} 3 & 1 & 3 & 3 & 1 \\ 1 & 3 & 1 & 1 & 3 \\ 3 & 1 & 3 & 3 & 1 \\ 3 & 1 & 3 & 3 & 1 \\ 1 & 3 & 1 & 1 & 3 \end{bmatrix} - \begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{bmatrix}$$

$$W = \begin{bmatrix} 0 & 1 & 3 & 3 & 1 \\ 1 & 0 & 1 & 1 & 3 \\ 3 & 1 & 0 & 3 & 1 \\ 3 & 1 & 3 & 0 & 1 \\ 1 & 3 & 1 & 1 & 0 \end{bmatrix}$$

$y_i = \text{sign}(WX_i - \theta)$ $\qquad$ $\theta \rightarrow$ threshold

$y_i \rightarrow$ Output vector

$$y_1 = \begin{bmatrix} 0 & 1 & 3 & 3 & 1 \\ 1 & 0 & 1 & 1 & 3 \\ 3 & 1 & 0 & 3 & 1 \\ 3 & 1 & 3 & 0 & 1 \\ 1 & 3 & 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$y_1 = \begin{bmatrix} 0+1+3+3+1 \\ 1+0+1+1+3 \\ 3+1+0+3+1 \\ 3+1+3+0+1 \\ 1+3+1+1+0 \end{bmatrix}_{5\times 1}$$

$$y_1 = \text{sign}\begin{bmatrix} 8 \\ 6 \\ 8 \\ 8 \\ 6 \end{bmatrix}$$

$$\therefore y_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = x_1$$

Similarly y_2 & y_3

$$\text{if } x_5 = \begin{bmatrix} -1 \\ 1 \\ -1 \\ 1 \\ 1 \end{bmatrix}$$

$$y_5 = \text{sign}\left[\begin{bmatrix} 0 & 1 & 3 & 3 & 1 \\ 1 & 0 & 1 & 1 & 3 \\ 3 & 1 & 0 & 3 & 1 \\ 3 & 1 & 3 & 0 & 1 \\ 1 & 3 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \\ -1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}\right]$$

$$= \begin{bmatrix} 0+1-3+3+1 \\ -1+0-1+1+3 \\ -3+1-0+3+1 \\ -3+1-3+0+1 \\ -1+3-1+1+0 \end{bmatrix} = \text{sign}\begin{bmatrix} 2 \\ 2 \\ 2 \\ -4 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ 1 \end{bmatrix}$$

$$y = \text{sign}\begin{bmatrix} 0 & 1 & 3 & 3 & 1 \\ 1 & 0 & 1 & 1 & 3 \\ 3 & 1 & 0 & 3 & 1 \\ 3 & 1 & 3 & 0 & 1 \\ 1 & 3 & 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ 1 \end{bmatrix}$$

$$y = \text{sign}\begin{bmatrix} 0+1+3-3+1 \\ 1+0+1-1+3 \\ 3+1+0-3+1 \\ 3+1+3-0+1 \\ 1+3+1-1+0 \end{bmatrix}$$

$$y = \text{sign}\begin{bmatrix} 2 \\ 4 \\ 2 \\ 8 \\ 4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = X_1$$

Q5) On Paper calculate weight matrix for a field network i.e to learn the following two input vector (N)

$$X_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}_{4\times1} \qquad X_2 = \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \end{bmatrix}_{4\times1}$$

Now, calculate the behaviour of Network (matrix) when it presented X_1 as i/p, how does it behave when it present at with following i/p

$$X_3 = \begin{bmatrix} -1 \\ -1 \\ 1 \\ -1 \end{bmatrix}_{4\times1}$$

Scanned with CamScanner

$$y = \text{sign}\begin{bmatrix} 0 & 1 & 3 & 3 & 1 \\ 1 & 0 & 1 & 1 & 3 \\ 3 & 1 & 0 & 3 & 1 \\ 3 & 1 & 3 & 0 & 1 \\ 1 & 3 & 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ 1 \end{bmatrix}$$

$$y = \text{sign}\begin{bmatrix} 0+1+3-3+1 \\ 1+0+1-1+3 \\ 3+1+0-3+1 \\ 3+1+3-0+1 \\ 1+3+1-1+0 \end{bmatrix}$$

$$y = \text{sign}\begin{bmatrix} 2 \\ 4 \\ 2 \\ 8 \\ 4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = X_1$$

(Q) On paper calculate weight matrix for a field network i.e to learn the following two input vector (N)

$$X_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}_{4\times1} \qquad X_2 = \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \end{bmatrix}_{4\times1}$$

Now, calculate the behaviour of network (matrix) when it presented X_1 as i/p, how does it behave when it present at with following i/p

$$X_3 = \begin{bmatrix} -1 \\ -1 \\ 1 \\ -1 \end{bmatrix}_{4\times1}$$

Soln)

$N = 2$

$x_1^t = [\,1 \quad 1 \quad 1 \quad 1\,]_{1\times 4}$

$x_2^t = [\,-1 \quad -1 \quad -1 \quad -1\,]_{1\times 4}$

$$W = \sum_{i=1}^{N} x_i x_i^t - NI$$

$$W = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \cdot [1 \ 1 \ 1 \ 1] + \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} \cdot [-1 \ -1 \ -1 \ -1] - 2 \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$W = \begin{bmatrix} 1+1+1+1 \\ 1+1+1+1 \\ 1+1+1+1 \\ 1+1+1+1 \end{bmatrix} + \begin{bmatrix} 1+1+1+1 \\ 1+1+1+1 \\ 1+1+1+1 \\ 1+1+1+1 \end{bmatrix} - \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} - \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

$$W = \begin{bmatrix} 0 & 2 & 2 & 2 \\ 2 & 0 & 2 & 2 \\ 2 & 2 & 0 & 2 \\ 2 & 2 & 2 & 0 \end{bmatrix}$$

$y_i = \text{sign}(Wx_i - \theta)$

$$y_1 = \text{sign}\begin{bmatrix} 0 & 2 & 2 & 2 \\ 2 & 0 & 2 & 2 \\ 2 & 2 & 0 & 2 \\ 2 & 2 & 2 & 0 \end{bmatrix}_{4\times4} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}_{4\times1} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$y_1 = \text{sign}\begin{bmatrix} 0+2+2+2 \\ 2+0+2+2 \\ 2+2+0+2 \\ 2+2+2+0 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$y_1 = \text{sign}\begin{bmatrix} 6 \\ 6 \\ 6 \\ 6 \end{bmatrix}$$

$$y_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = x_1$$

Given:

$$x_3 = \begin{bmatrix} -1 \\ -1 \\ 1 \\ -1 \end{bmatrix}$$

$$y_3 = \text{sign}\begin{bmatrix} 0 & 2 & 2 & 2 \\ 2 & 0 & 2 & 2 \\ 2 & 2 & 0 & 2 \\ 2 & 2 & 2 & 0 \end{bmatrix} \begin{bmatrix} -1 \\ -1 \\ 1 \\ -1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$y_3 = \text{sign}\begin{bmatrix} 0-2+2-2 \\ -2+0+2-2 \\ -2-2+0-2 \\ -2-2+2-0 \end{bmatrix} = \text{sign}\begin{bmatrix} 2 \\ -2 \\ -6 \\ -2 \end{bmatrix} = \begin{bmatrix} 1 \\ -1 \\ -1 \\ -1 \end{bmatrix}$$

$$y_4 = \text{sign}\begin{bmatrix} 0 & 2 & 2 & 2 \\ 2 & 0 & 2 & 2 \\ 2 & 2 & 0 & 2 \\ 2 & 2 & 2 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \\ -1 \\ -1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$= \text{sign}\begin{bmatrix} 0-2-2-2 \\ 2+0-2-2 \\ 2-2+0-2 \\ 2-2-2-0 \end{bmatrix}$$

$$y_4 = \text{sign}\begin{bmatrix} -6 \\ -2 \\ -2 \\ -2 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} = X_2$$

Bi-directional Association Memory [BAMP]

→ It is a neural network (bart-kosko 1980) i.e similar to network of hopfield, that can be used to associate items from on set to another set of items.

$$\boxed{W = \sum_{i}^{n} X_i Y_i^t} \qquad \boxed{Y_i = \text{sign}(W^t X_i)}$$

eg) $X_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ $X_2 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$

$Y_1 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$ $Y_2 = \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}$

Soln) $W = X_1 Y_1^t + X_2 Y_2^t$

$$= \begin{bmatrix} 1 \\ 1 \end{bmatrix}_{2\times1} \cdot \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}_{1\times3} + \begin{bmatrix} -1 \\ -1 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix}_{2\times3} + \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}_{2\times3}$$

$$W = \begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix}_{2\times 3}$$

$$Y = \text{sign}\begin{bmatrix} 2 & 2 \\ 2 & 2 \\ 2 & 2 \end{bmatrix}_{3\times 2} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix}_{2\times 1}$$

$$= \text{sign}\begin{bmatrix} 2+2 \\ 2+2 \\ 2+2 \end{bmatrix}$$

$$= \text{sign}\begin{bmatrix} 4 \\ 4 \\ 4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = Y_1$$

$$Y = \text{sign}\begin{bmatrix} 2 & 2 \\ 2 & 2 \\ 2 & 2 \end{bmatrix} \cdot \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$

$$= \text{sign}\begin{bmatrix} -2-2 \\ -2-2 \\ -2-2 \end{bmatrix}$$

$$= \text{sign}\begin{bmatrix} -4 \\ -4 \\ -4 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix} = Y_2$$

To find 'x' $\boxed{X_i = \text{sign}(W Y_i)}$

$$X_1 = \text{sign}\begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$= \text{sign}\begin{bmatrix} 6 \\ 6 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} = X_1$$

Pattern classification

Q) using Heb rule find weight require to perform the following classification of given i/p Pattern '+' symbol that represent value 1 & empty square minus 1. consider 'I' belongs to members of class, so that target value will be 1 & 'O' does not belong to members of class i.e [-1]

soln)

+	+	+
	+	
+	+	+

I

$Y = 1$

+	+	+
+		+
+	+	+

O

$Y = -1$

Pattern		x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	Test o/p Y
I	=	1	1	1	-1	1	-1	1	1	1	1
O	=	1	1	1	1	-1	1	1	1	1	-1

$$\boxed{\begin{aligned} W_{new} &= W_{old} + x_i y \\ b_{new} &= b_{old} + y \end{aligned}}$$

$x_i \rightarrow$ i/p vector

$Y \rightarrow$ test o/p

$\boxed{\text{Change in } w \quad \Delta w = x_i y}$

initial $W = 0$ (all) Initial $b = 0$

Now,

$W_1 = 0 + 1 \times 1 = 1$ $W_4 = 0 + (-1)(1) = -1$ $W_7 = 0 + 1 \times 1 = 1$

$W_2 = 0 + 1 \times 1 = 1$ $W_5 = 0 + (1)(1) = 1$ $W_8 = 1$

$W_3 = 0 + 1 \times 1 = 1$ $W_6 = 0 + (-1)(1) = -1$ $W_9 = 1$

$W_{new} = [1 \ 1 \ 1 \ -1 \ 1 \ -1 \ 1 \ 1 \ 1]$

New weight i.e W_{new} becomes W_{old} for 'O'

Scanned with CamScanner

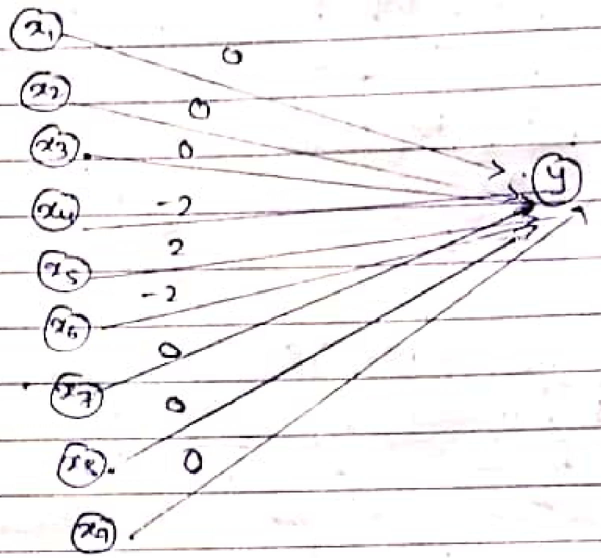
$w_1 = 1 + 1(-1) = 0$ $w_4 = -1 + (1)(-1) = -2$ $w_7 = 0$

$w_2 = 1 + 1(-1) = 0$ $w_5 = 1 + (1)(-1) = 2$ $w_8 = 0$

$w_3 = 1 + 1(-1) = 0$ $w_6 = -1 + (1)(-1) = -2$ $w_9 = 0$

for '0':

$W_{new} = [0 \; 0 \; 0 \; -2 \; 2 \; -2 \; 0 \; 0 \; 0]$



General hypothesis {? ? ? ? ? ?}

Specific " { ϕ ϕ ϕ ϕ }

Candidate Elimination Algorithm

Example	Sky	Air Temp	Humidity	wind	water	Forecast	Enjoysport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	"	"	"	Yes
3	Rainy	~~Warm~~ cold	High	"	"	change	No
4	Sunny	~~Warm~~ warm	High	"	cool	"	yes

soln) S_0 : < ϕ ϕ ϕ ϕ ϕ ϕ >

S_1 : <Sunny, Warm, Normal, Strong, Warm, Same>

S_2 : <Sunny, Warm, ?, Strong, Warm, Same>

S_3 : <Sunny, Warm, ?, Strong, Warm, Same>

S_4 : <Sunny, Warm, ?, Strong, ?, ?>

G_0 : < ? ? ? ? ? ? >

G_1 : < ? ? ? ? ? ? >

G_2 : < ? ? ? ? ? ? >

G_3 : <Sunny, ?, ?, ?, ?, ?> <?, warm, ?, ?, ?, ?> <?, ?, Normal, ?, ?, ?>
<?, ?, ?, ?, cool, ?> <?, ?, ?, ?, ?, same>

G_4 : <sunny, ?, ?, ?, ?, ?> <?, warm, ?, ?, ?, ?>

eliminate normal, cool, same

To write few more hypothesis towards the learn origin space by candidate elimination, we consider one from general & one from specific hypothesis

S_4 : <sunny, warm, ?, strong, ?, ?>

G_4 : <Sunny, ?, ?, ?, ?, ?> <?, warm, ?, ?, ?, ?>

<sunny, warm, ?, ~~strong~~ ?, ?, ?> <Sunny, ?, ?, Strong, ?, ?>
<?, warm, ?, Strong, ?, ?>

8)

	Size	Color	Shape	Class/label
1	Big	Red	Circle	No
2	Small	Red	Δle	No
3	Small	Red	Circle	Yes
4	Big	Blue	Circle	No
5	Small	Blue	Circle	Yes

Soln) S_0 = < ϕ ϕ ϕ ϕ ϕ >

S_1 : ~~Big Red circle~~ < ϕ ϕ ϕ >

S_2 : < ϕ ϕ ϕ >

S_3 : <Small, Red, circle>

S_4 : <Small, Red, Circle>

S_5 : <Small, ?, circle> {if negative retained previous Hyp}

$G_0 = <?\ ?\ ?>$

G_1 : <Small, ?, ?> <?, Blue, ?> <?, ?, Δle>

G_2 : <Small, Blue, ?> <Small, ?, Circle>

<?, Blue, ?> <?, Blue, Triangle> <Big, ?, Triangle>

G_3 : <Small, ?, Circle>

G_4 : <Small, ?, Circle>

G_5 : <Small, ?, Circle>

final hypothesis

S_5 : <Small, ?, Circle>

G_5 : <Small, ?, Circle>

Decision Tree:

→ Entropy - Eliminate the impurity from dataset.

→ Information Gain

$$\boxed{\text{Entropy}(S) = -P_1 \log_2 P_1 - P_0 \log_2 P_0}$$

P_1 → Probability of +ve value

P_0 → Probability of -ve

Information gain

$$\boxed{\text{Gain}(S,A) = \text{Entropy}(S) - \sum_{V \in \text{values}(A)} \frac{|S_V|}{|S|} \text{Entropy}(S_V)}$$

if equal no. of positive values & -ve values then $\boxed{\text{Entropy}(S) = 1}$

only +ve values then $\boxed{\text{Entropy}(S) = 0}$

only -ve values then $\boxed{E(S) = 0}$

Note:

If Gain is more make it as root node

1)

Film	Country of origin	Big star	Genre	Success
1	U.S.A	Yes	Sci-fic	True
2	U.S.A	NO	comedy	False
3	U.S.A	Yes	"	True
4	Europe	NO	"	True
5	Europe	Yes	Sci-fic	False
6	Europe	Yes	Romance	False
7	rest of the world	Yes	comedy	False
8	"	NO	Sci-fic	False
9	Europe	Yes	comedy	True
10	U.S.A	Yes	comedy.	True

Sol)

$$H(U.S.A) = -\underbrace{\frac{3}{4}\log_2\frac{3}{4}}_{True} - \underbrace{\frac{1}{4}\log_2\frac{1}{4}}_{False}$$

$$= 0.811$$

$H(Europe) = 1$ $(\because$ no. True = no. False$)$

$H(\text{rest-of-the-world}) = 0$ $(\because$ all negative i.e False$)$

$$\underset{\text{entire attribute}}{Gain} = H(CO) - \left[\left(\frac{4}{10}\right)\cdot(0.811) + \left(\frac{4}{10}\right)(1) + \left(\frac{2}{10}\right)\cdot(0)\right]$$

$$= 1 - \left[\frac{2}{5}\cdot 0.811 + \frac{2}{5} + 0\right]$$

$\{H(CO) = 1 \because$ equal no. of True & False$\}$

$$Gain = \underline{0.2756}$$

For Big star

$$H(Yes) = -\frac{4}{7}\log_2\left(\frac{4}{7}\right) - \frac{3}{7}\log_2\left(3/7\right)$$

$$= \underline{0.985}$$

$$H(No) = -\frac{1}{3}\log_2\left(\frac{1}{3}\right) - \frac{2}{3}\log_2\left(2/3\right) = \underline{0.918}$$

$H(\text{Big star}) = \frac{\cancel{0}}{7} \quad \frac{1}{-}$

Now,

$$\text{Gain} = 1 - \left[\frac{7}{10}(0.985) + \frac{3}{10}(0.918) \right]$$

$$\text{Gain} = 0.035$$

For Genre:

$$H(\text{Sci-fi}) = -\frac{1}{3}\log_2\left(\frac{1}{3}\right) - \frac{2}{3}\log_2\left(\frac{2}{3}\right)$$

$$= 0.918$$

$$H(\text{Comedy}) = -\frac{4}{6}\log_2\left(\frac{4}{6}\right) - \frac{2}{6}\log_2\left(\frac{2}{6}\right)$$

$$= 0.918$$

$$H(\text{Romance}) = 0$$

Now,

$$\text{Gain} = 1 - \left[\frac{3}{10}(0.918) + \frac{6}{10}(0.918) + 0 \right]$$

$$\text{Gain} = 0.1738$$

Since Gain (Country origin) is more compare to other

Country of origin is root note.

For U.S.A.

Film	Big Star	Genre	Success
1	Yes	Sci-fic	True
2	No	Comedy	False
3	Yes	Comedy	True
10	Yes	Comedy	True

$H(S) = -\frac{3}{4}\log_2\left(\frac{1}{4}\right) - \frac{1}{4}\log_2\left(\frac{1}{4}\right) = 0.811$

Big star

$H(\text{Yes}) = 0$

$H(\text{No}) = 0$

Gain = 0.811

For genre

$H(\text{sci-fic}) = 0$

$H(\text{comedy}) = -\frac{2}{3}\log_2\left(\frac{2}{3}\right) - \frac{1}{3}\log_2\left(\frac{1}{3}\right)$

$= 0.918$

$H(\text{Genre}) = -\frac{1}{4}\log_2\left(\frac{1}{4}\right) - \frac{3}{4}\log_2(3/4) = 0.811$

Gain $= 0.811 - [1/4 \times 0 + 3/4 \times 0.918]$

Gain $= 0.1225$

Big star is root node

For Europe:

Film	Bigstar	Genre	Success
4	No	Comedy	True
5	Yes	Sci-fic	False
6	Yes	Romance	False
9	yes	comedy	True

Bigstar

$H(Yes) = -\frac{1}{3}\log_2(1/3) - \frac{2}{3}\log_2(2/3)$

$= 0.918$

$H(NO) = 0$

$H(Bigstar) = -\frac{2}{4}. 1$

$Gain = 1 - \left(\frac{3}{4} \times 0.918 + 0\right) = \underline{0.3115}$

Genre

$H(comedy) = 0$

$H(Sci\text{-}fic) = 0$

$H(Romance) = 0$

$H(Genre) = 1$ (∵ no. of True = no. of False)

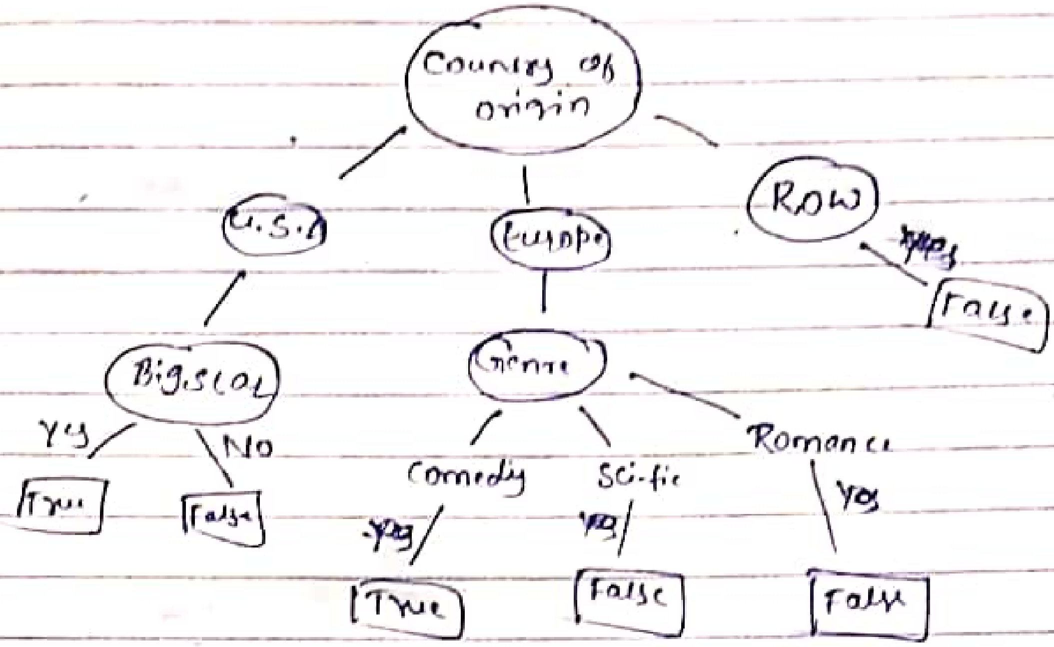
$Gain = 1 - (0 + 0 + 0)$

$= \underline{1}$

$Gain_{(Genre)} > Gain_{(Bigstar)}$

Genre root node

Final decision tree



Scanned with CamScanner