Neural networks

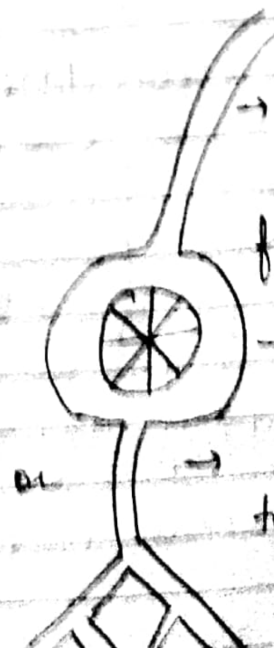
Artificial neural networks (ANN)

1 ANN will mimic the activities of Biological neural network present in human brain

2 ANN → parallel distributed system / adaptive system, since it composes of series of interconnected processing elements that operate in parallel

Biological neural network

Diagram:



→ dendrites (accept inputs) from surrounding neurons

→ soma (processing unit)

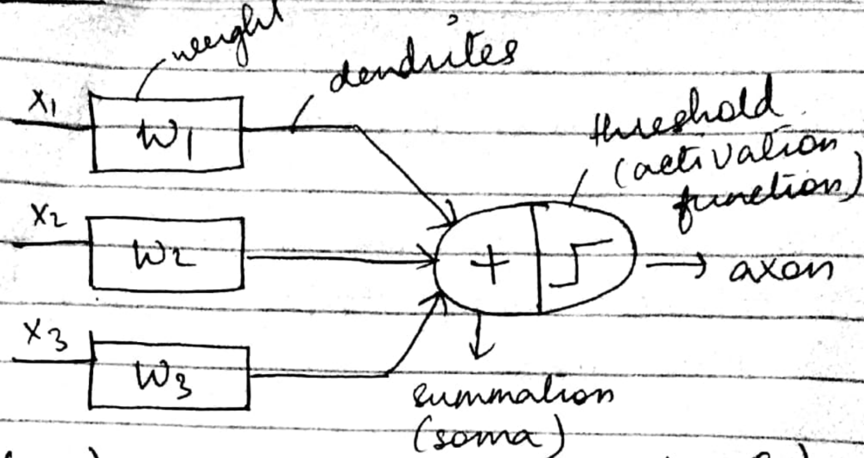
→ axon (channel) transfer the signal to o/p unit

→ synapses → when a particular amt of i/p is ~~reached~~ received, then cell will transfer it.

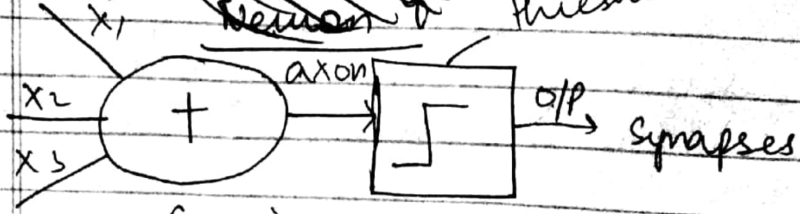
after set value or set point is reached, it is given to the neighbouring neurons

We'll convert this biological model into a mathematical one

ANN

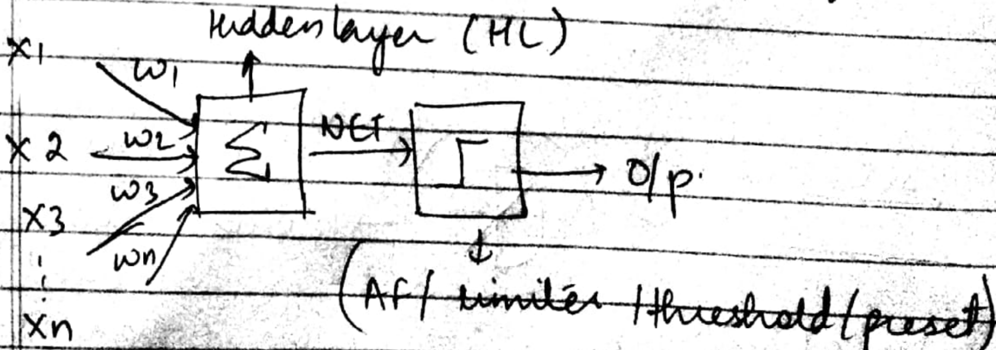


(ANN) ~~Structure of Neuron~~



(given to synapses after threshold is reached), else no output.

<u>Structure of Neuron (ANN) (simplified version</u>



NET → intermediate answer

$$NET = X_1W_1 + X_2W_2 + \cdots X_nW_n$$

$$\boxed{NET = XW}$$

X → summation of inputs

W → summation of outputs (weights)

O/p = out = K(NET)

OUT = 1

if NET ≥ T

otherwise, T → threshold

OUT = 0

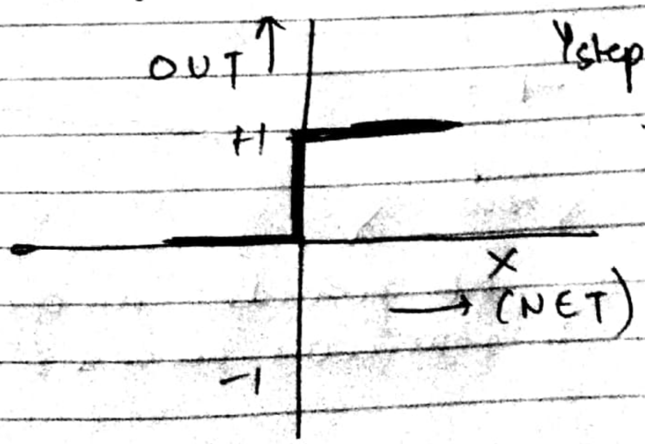
Term comparison

BNN	ANN
Soma	Neuron
Dendrite	i/p
Axon	o/p
Synapse	weight / T

Types of Activation functions (AF)

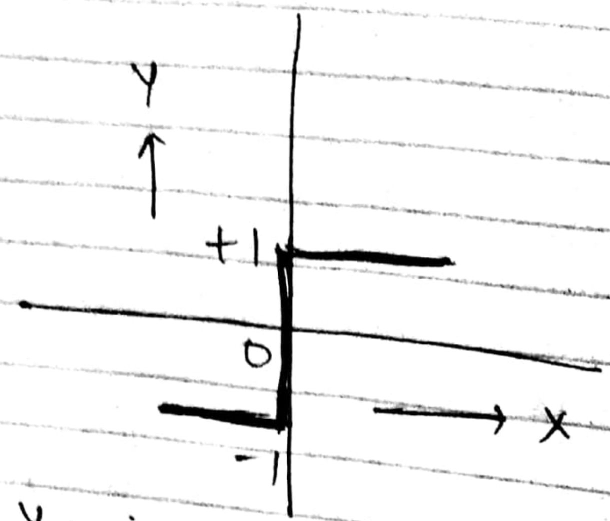
1. step
2. Sign
3. Linear
4. Sigmoid

1. Step function



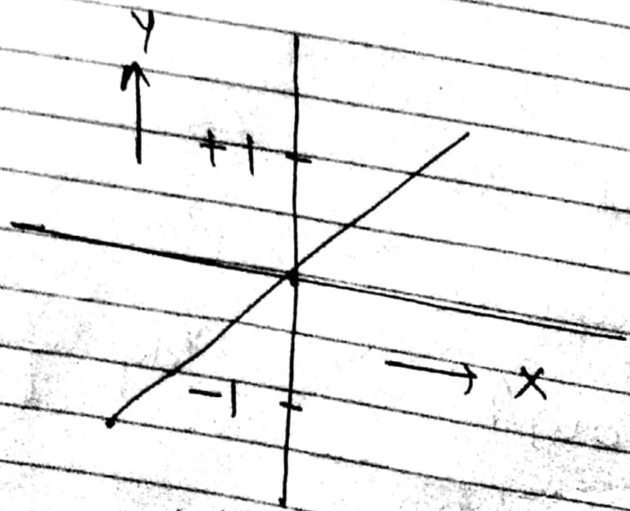
$$Y_{step} = \begin{cases} 1, & \text{if } X \geq 0 \\ 0, & \text{if } X < 0 \end{cases}$$

2. Sign function



$$Y_{sign} = \begin{cases} +1, & \text{if } X \geq 0 \\ -1, & \text{if } X < 0 \end{cases}$$

3. Linear function



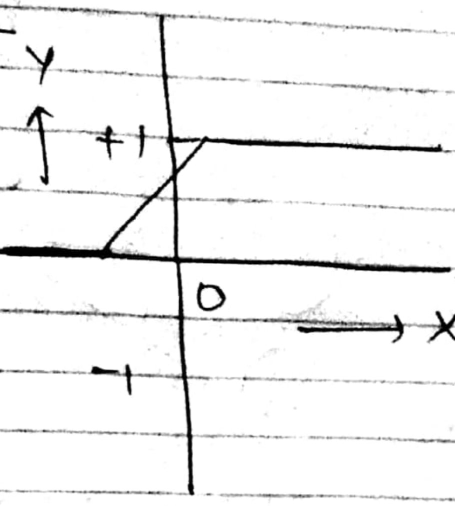
$y^{new} = X$

The linear AF provides an o/p signal equal to the neuron weighted i/p.

Application → Linear Approximation

4. <u>Sigmoid function</u>

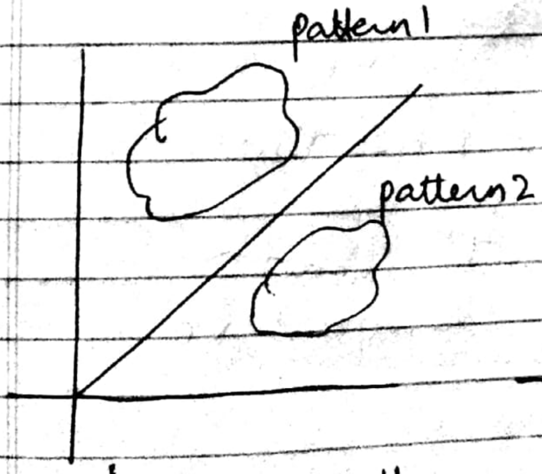
$$\left[\bar{Y}^{\text{sign out}} = \frac{1}{1+e^{-x}}\right]$$



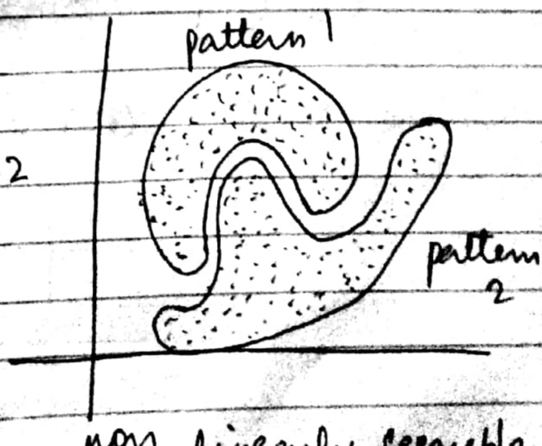
Application → used in back propagation networks

I/P will vary from -1 to +1 but o/p will vary from 0 to 1.

<u>linearly separable tasks</u> [2 marks Ques]

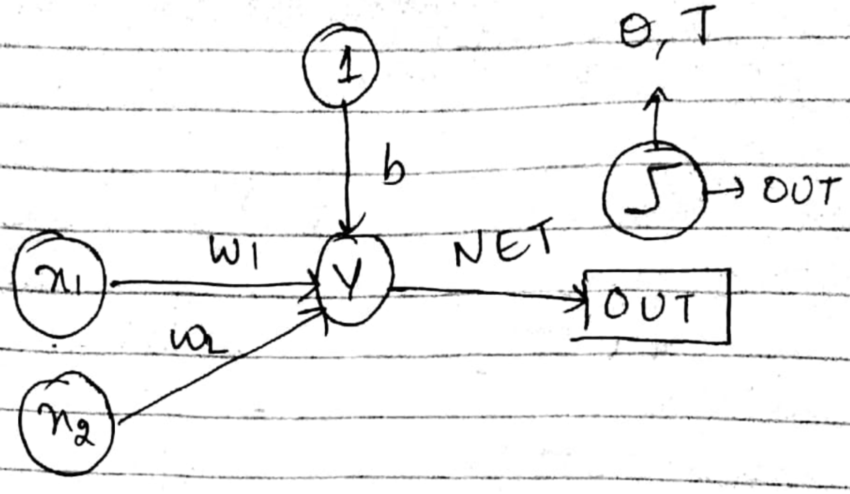


linearly separable patterns



non linearly separable patterns

BIAS



ANN

1. Bias will improve overall performance of the ANN.

2. It is equal to weight w.

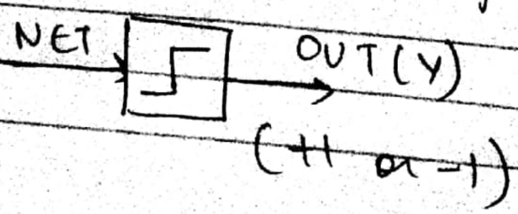
3. Normally 'b' value will be fixed at 1 but can be varied between 0.1 to 1

$$NET = OUT = b + \sum x_i w_i$$

$$A.F. = \begin{cases} +1, & \text{if net} \geq 0 \\ -1 & \text{if net} < 0 \end{cases}$$

Threshold (θ, T)

The threshold value will be fixed by the user



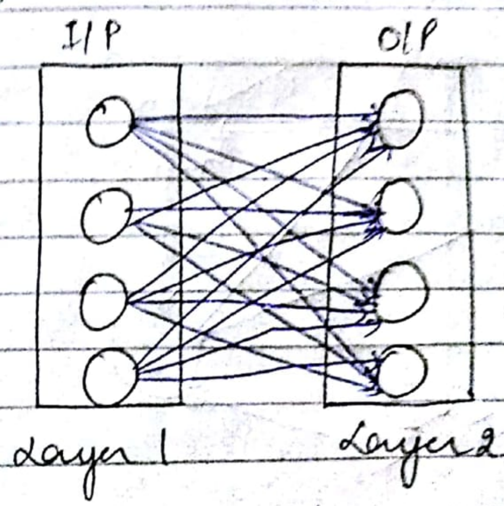
$$Y = f(Net) = \begin{cases} + , & \text{if } net > \theta \\ - , & \text{if } net < \theta \end{cases}$$

net → out
θ → user defined value

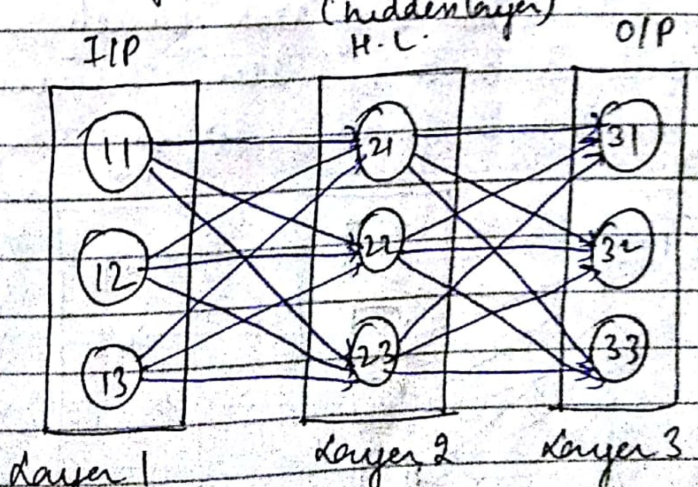
Basic building blocks of ANN

1) Network architecture
2) Setting of weights
3) Activation function

I) Single layer feed forward network



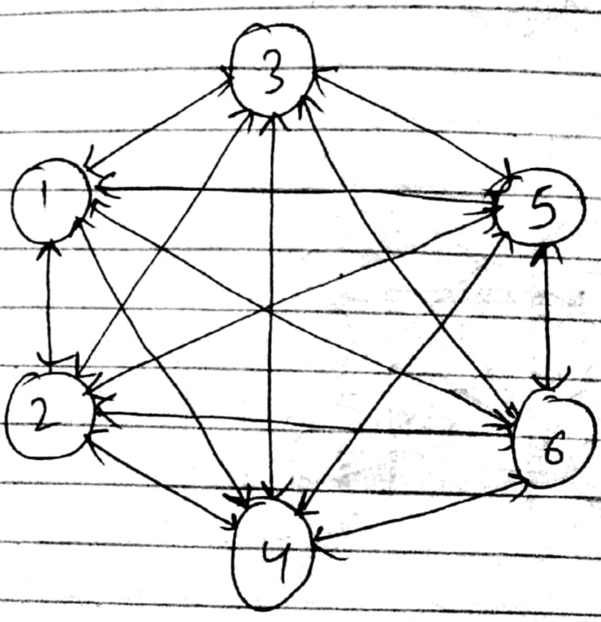
II) Multi layer feed forward network



In general, two layer ANN can be represented as (m, n, c)
where,

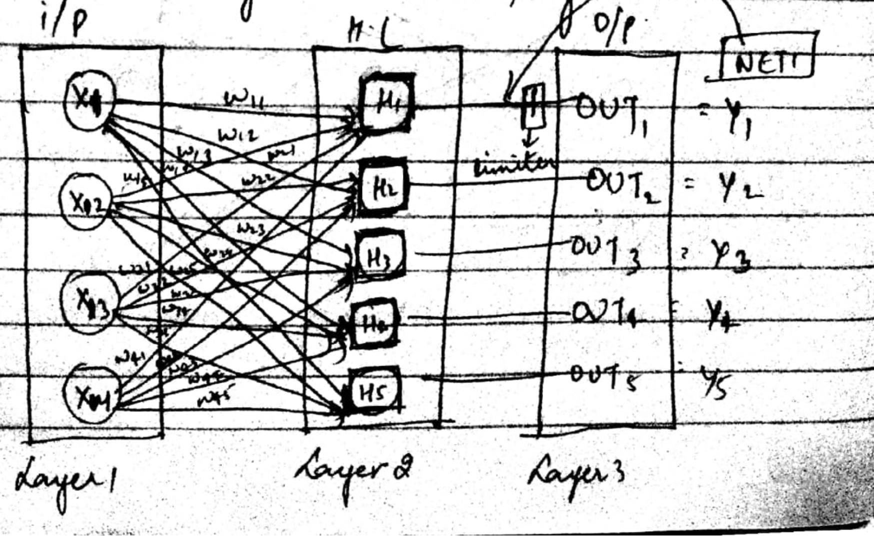
m → no of i/p layers
n → no of hidden layers
c → no of o/p layers

Fully Recurrent Network



Ques. Draw a structure of ANN which contains 4 inputs and 5 number of neurons in a single hidden layer

Soln.



writing the weights for all i/p lines.

Equations for o/p.

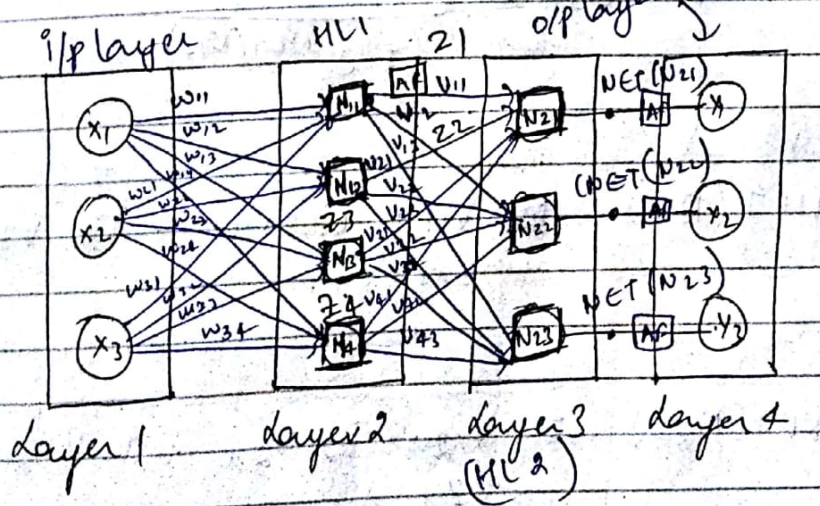
$NET1 = X_1 W_{11} + X_2 W_{21} + X_3 W_{31} + X_4 W_{41}$

$$Y_1 = \frac{1}{1+e^{-NET1}} \qquad Y_3 = \frac{1}{1+e^{-NET3}}$$

$$Y_2 = \frac{1}{1+e^{-NET2}} \qquad Y_4 = \frac{1}{1+e^{-NET4}}$$

$$Y_5 = \frac{1}{1+e^{-NET5}}.$$

Q. Draw an ANN structure for 3 inputs, 2 hidden layers (first H.L → 4 neurons and second H.L → 3 neurons) and 3 neurons in the o/p layer.



NET

$NET(N_{11}) = X_1 W_{11} + X_2 W_{21} + X_3 W_{31}$

$NET(N_{12}) = X_1 W_{12} + X_2 W_{22} + X_3 W_{32}$

$NET(N_{13}) = x_1 w_{13} + x_2 w_{23} + x_3 w_{33}$

$NET(N_{14}) = x_1 w_{14} + x_2 w_{24} + x_3 w_{34}$

Considering sigmoidal activation function, and applying A.f. after the intermediate o/p,

A.f.

o/p of first hidden layer

$$Z_1 = \frac{1}{1+e^{-NET(N_{11})}}$$

$$Z_2 = \frac{1}{1+e^{NET(N_{12})}}$$

$$Z_3 = \frac{1}{1+e^{NET(N_{13})}}$$

$$Z_4 = \frac{1}{1+e^{-NET(N_{14})}}$$

$NET(N_{21}) = Z_1 V_{11} + Z_2 V_{21} + Z_3 V_{31} + Z_4 V_{41}$

$NET(N_{22}) = Z_1 V_{12} + Z_2 V_{22} + Z_3 V_{32} + Z_4 V_{42}$

$NET(N_{23}) = Z_1 V_{13} + Z_2 V_{23} + Z_3 V_{33} + Z_4 V_{43}$

Step

Applying activation function (sigmoidal)

$$Y_1 = \frac{1}{1+e^{-NET(N_{21})}}$$

$$Y_2 = \frac{1}{1+e^{-NET(N_{22})}}$$

$$Y_3 = \frac{1}{1+e^{-NET(N_{23})}}$$

Q Draw an ANN structure for the given details: single i/p layer with 4 neurons, 1st hidden layer with 4 neurons, 2nd hidden layer with 5 neurons, 3rd hidden layer with 3 neurons, single o/p layer with 3 neurons.

Plotting of response region for the AND function

Binary form.

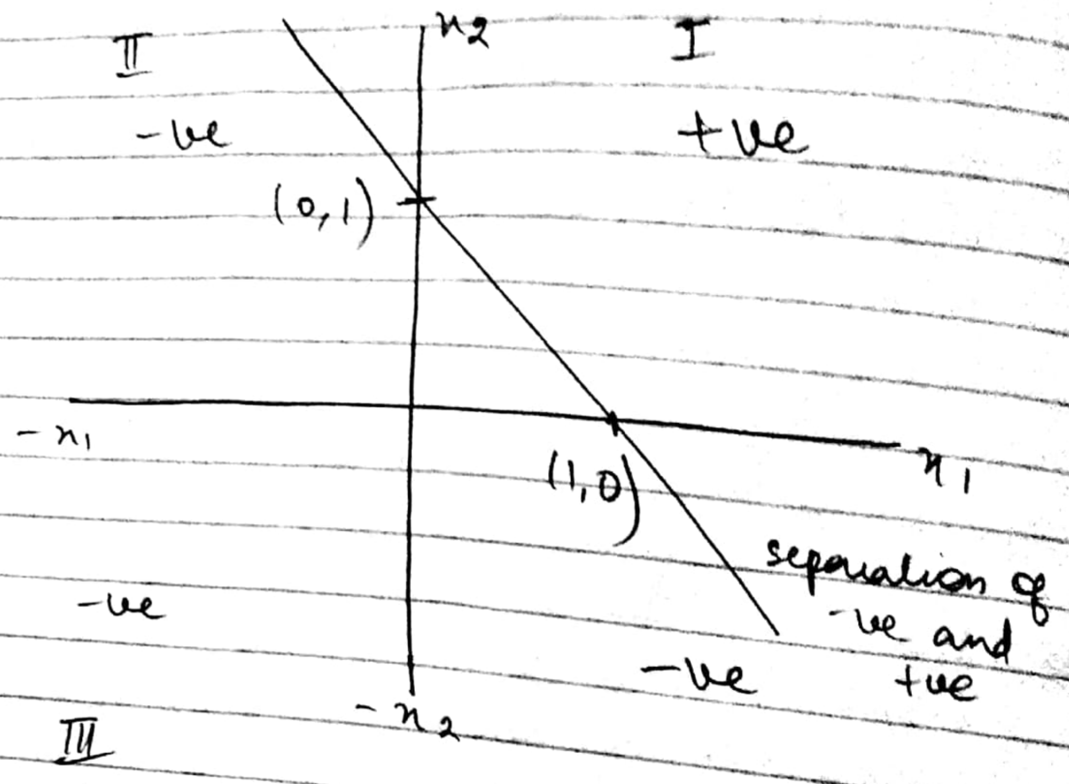
Truth table for AND function

X_1	X_2	Y
0	0	0
0	1	0
1	0	0
1	1	1

Step 1: Convert binary to bipolar form
(0 → -1)

X_1	X_2	Y	
-1	-1	-1	negative o/p
-1	1	-1	
1	-1	-1	
1	1	1	

Step 2: Plotting response graph (using bipolar form)



Separating line eqn.

$$x_2 = -\frac{w_1}{w_2} x_1 - \frac{b}{w_2}$$

Assume, $b = -1$

$w_1 = w_2 = 1$

$$x_2 = \frac{-1}{1} x_1 - \frac{(-1)}{1}$$

$$x_2 = -x_1 + 1$$

~~for (1,1) Combination~~

~~or~~ $\boxed{x_1 + x_2 = 1}$

Scanned by CamScanner

Assume,
$x_1 = 0$

If $x_1 = 0$, $x_2 = 1$ $(0, 1)$
If $x_2 = 0$, $x_1 = 1$ $(1, 0)$

Plotting of response for OR gate

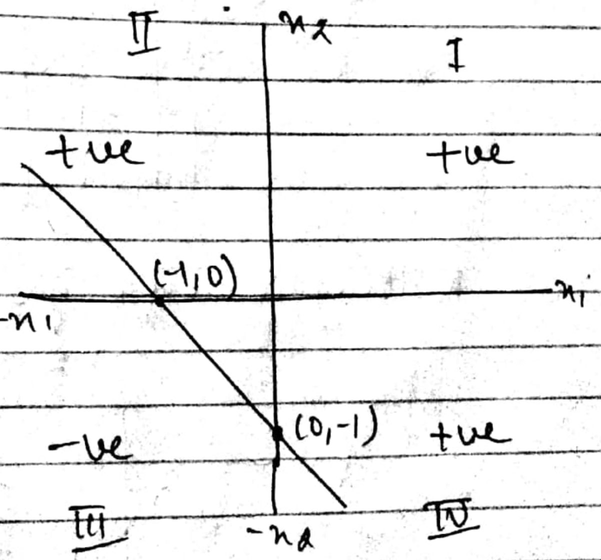
Truth table for OR gate

X_1	X_2	Y
0	0	0
0	1	1
1	0	1
1	1	1

Converting to bipolar form

X_1	X_2	Y
-1	-1	-1
-1	1	1
1	-1	1
1	1	1

Plotting for bipolar form



Separating line eq^n $\Rightarrow$ $x_2 = \frac{-w_1}{w_2} x_1 - \frac{b}{w_2}$

Assume, $b = 1$
$w_1 = w_2 = 1$

$x_2 = -x_1 - 1$
$x_1 + x_2 = -1$

$x_1 + x_2 = -1$

If $x_1 = 0$, $x_2 = -1$ $(0, -1)$
$x_2 = 0$, $x_1 = -1$ $(-1, 0)$

Plotting of response region for XOR gate

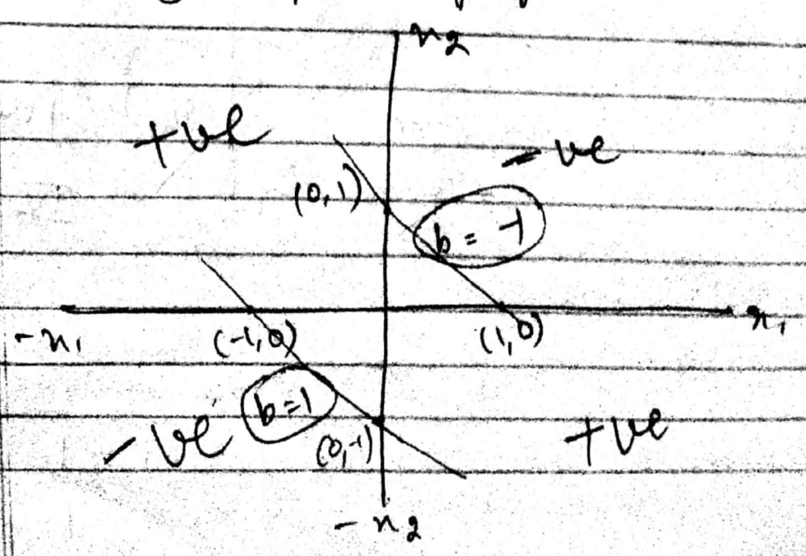
Truth table in binary form

X_1	X_2	Y
0	0	0
0	1	1
1	0	1
1	1	0

Step 1: Convert binary to bipolar form $(0 \to -1)$

X_1	X_2	Y
-1	-1	-1
-1	1	1
1	-1	1
1	1	-1

Step 2: Plotting response graph.



Case 1: $b = 1$, $w_1 = w_2 = 1$

Separating line eq^n: $x_2 = -\frac{w_1}{w_2} x_1 - \frac{b}{w_2}$

$$x_2 = -\left(\frac{1}{1}\right)x_1 - \left(\frac{1}{1}\right)$$

$$x_2 = -x_1 - 1$$
$$x_1 + x_2 = -1$$

If $x_1 = 0$, $x_2 = -1$ $(0, -1)$
If $x_2 = 0$, $x_1 = -1$ $(-1, 0)$

Case 2:

$b = -1$, $w_1 = w_2 = 1$

Separating line eq^n: $x_2 = -\frac{w_1}{w_2} x_1 - \frac{b}{w_2}$

$$x_2 = -\left(\frac{1}{1}\right)x_1 - \left(\frac{-1}{1}\right)$$

$$x_2 = -x_1 + 1$$
$$x_1 + x_2 = 1$$

If $x_1 = 0$, $x_2 = 1$ $(0, 1)$
If $x_2 = 0$, $x_1 = 1$ $(1, 0)$

Plotting these points in the graph.

→ Using this mathematical technique, the o/p can't be segregated.

TRAINING ALGORITHM FOR ANN

Step 1: Initialize weights and bias (initially it can be zero)

$$(w_1 = w_2 = b = 0)$$

Set learning rate (α) from (0 to 1)

Step 2: If stopping condition fails, do steps 3 to 7.

Step 3: For each training pair (s : t), do steps 4 to 6.

i/p combination (x_1, x_2) → s ; t → target

for example in AND

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	0	1

S ← i/p comb.

t ← target

Step 4: Set activations of i/p units

$$x_i = s_j \quad \text{for } i = 1 \text{ to } n$$

Step 5: Compute the o/p unit response

$$y_{in} = \underbrace{b}_{\text{bias}} + \sum_{i=1}^{n} x_i w_i$$

n → no. of inputs (2 for above example)

Activation function (AF)

$$Y = f(Y_{in}) = \begin{cases} 1, & \text{if } Y_{in} > \theta \text{ (Preset value)} \\ 0, & \text{if } -\theta \leq Y_{in} \leq \theta \\ -1, & \text{if } Y_{in} < -\theta \end{cases}$$

Step 6: The weights and bias are updated if the target is not equal to the o/p response (calculated).

If $t \neq Y$ and the value of X_i(i/p) $\neq 0$ then,

$$\Rightarrow \; w_i(\text{new}) = w_i(\text{old}) + \underbrace{\boxed{\alpha \cdot t \cdot x_i}}_{\Delta w \rightarrow \text{change in weight}}$$

and,

$$\Rightarrow b(\text{new}) = b(\text{old}) + \alpha \cdot t$$

else, $t = Y$

$$w_i(\text{new}) = w_i(\text{old})$$
$$b(\text{new}) = b(\text{old})$$

Step 7: Check for stopping condition.

Question:

→ Develop a perceptron [ANN] for the AND function with bipolar inputs and outputs

Sol^n: The training ~~pattern~~ patterns for AND function can be

Scanned by CamScanner

AND Truth table

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Conversion to bipolar

4 comb. = patterns

x_1	x_2	y
-1	-1	-1
-1	1	-1
1	-1	-1
1	1	1

Step 1: $w_1 = w_2 = 0$

$b = 0$

Learning value/factor/ $\alpha = 1$

Threshold value/preset value / $\theta = 0$

Step 2: Begin computation / iterations

Step 3: For i/p pair (pattern no 4) $\overset{x_1}{(1}, \overset{x_2}{1)} : \overset{y}{1}$ do steps 4 to 6.

Step 4: Set activations of i/p unit

$x_i = (1, 1)$

Step 5: Calculate the net i/p

$y_{in} = b + \sum x_i w_i$

$y_{in} = 0 + \cancel{x_1 w_1 + x_2 w_2}$ $\quad (\because w_1 = w_2 = 0)$

$y_{in} = 0.$

Applying activation function (AF)

θ (preset value)

$$Y = f(Y_{in}) = \begin{cases} 1 & , \text{if } Y_{in} > \theta \\ 0 & , \text{if } -\theta \le Y_{in} \le \theta \\ -1 & , \text{if } Y_{in} < -\theta \end{cases}$$

$\boxed{Y = 0}$ $\neq$ target value i.e. 1

Step 6: $t \neq Y$

$$w_{i(new)} = w_i(old) + \alpha \cdot t \cdot x_i$$

$$b(new) = b(old) + \alpha \cdot t$$

preset, target, i/p 1

$$\Rightarrow w_1(new) = w_1(old) + \alpha \cdot t \cdot x_1$$
$$= 0 + 1 \cdot 1 \cdot 1$$
$$= 1$$

$$\Rightarrow w_2(new) = w_2(old) + \alpha \cdot t \cdot x_2$$
$$= 0 + 1 \cdot 1 \cdot 1$$
$$= 1$$

$$\Rightarrow b_1(new) = b_1(old) + \alpha \cdot t$$
$$= 0 + 1 \cdot 1$$
$$= 1$$

~~$\Rightarrow b_2$ (new)~~

At the end of this step, we get $w_1 = w_2 = b = 1$ (next iteration initial values)

After this, solve for other 3 i/p - o/p patterns also. (In the exam, follow the alternative tabular method)

1: Design of perceptron for the AND function using bipolar i/p's and output (target)

Iteration 1

(Tabular column method)

$y_{in} = b + \sum_{i}^{n} x_i w_i$ → AF

$\Delta b = \alpha \cdot t$, $\Delta w_1 = \alpha t x_1$, $\Delta w_2 = \alpha t x_2$

(0) → initial values

I/P X_1	I/P X_2	B	NET (y_{in})	O/P (Y)	Target (t)	Change in weights Δw_1	Δw_2	Δb	Weights w_1	w_2	b
									(0)	(0)	(0)
1	1	1	0	0 $Y \neq t$	1	1	1	1	1	1	1
-1	1	1	1	1 $Y \neq t$	-1	1	-1	-1	2	0	0
1	-1	1	2	1 $Y \neq t$	-1	-1	1	-1	1	1	-1
-1	-1	1	-3	-1 $Y = t$	-1	0	0	0 (if $Y = t$)	1	1	-1

$y_{in} = B + X_1 w_1 + X_2 w_1$

$0 + 0 + 0 = 0$

Truth table in bipolar form

X_1	X_2	Y
-1	-1	-1
-1	1	-1
1	-1	-1
1	1	1

=)

X_1	X_2	Y
1	1	1
-1	1	-1
1	-1	-1
-1	-1	-1

for o/p

$$Y = f(Y_{in}) = \begin{cases} +1, & \text{if } Y_{in} > 0 \\ 0, & \text{if } -0 \le Y_{in} \le 0 \\ -1, & \text{if } Y_{in} < -0 \end{cases}$$

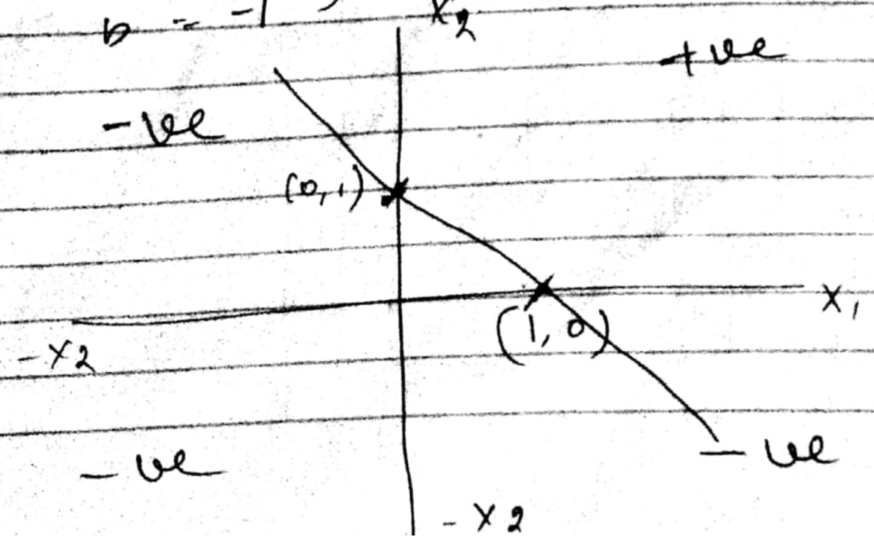
$W_1(new) = W_1(old) + \Delta W_1$
$= 1 + 0 = 1$

B(bias) = expected value

Iteration 2.

	Input x_1	x_2	B	NET (Y_{in})	O/P (Y)	Target (t)	Change in weights ΔW_1	ΔW_2	Δb	Weights w_1 (1	w_2 1	b -1)
P-1	1	1	1	1	1 $Y=t$	1	0	0	0	1	1	-1
P-2	-1	1	1	-1	-1 $Y=t$	-1	0	0	0	1	1	-1
P-3	1	-1	1	-1	-1 $Y=t$	-1	0	0	0	1	1	-1
P-4	-1	-1	1	3	-1 $Y=t$	-1	0	0	0	1	1	-1

So, $w_1 = 1$, $w_2 = 1$, $b = -1$



$$x_2 = -x_1 \frac{w_1}{w_2} - \frac{b}{w_2}$$

$w_1 = w_2 = 1$, $b = -1$

$x_2 = -x_1 + 1$

$\boxed{x_1 + x_2 = 1}$ ⇒ eq^n of st. line

if $x_1 = 0$, $x_2 = 1$ (0,1)
if $x_2 = 0$, $x_1 = 1$ (1,0) } 2 pts.

Que. Design a perception for the OR function with binary inputs and bipolar targets ($\alpha = 1$, $\theta = 0.2$)

Sol^n. Truth Table for OR function (binary i/p's bipolar o/p.)

x_1	x_2	y
0	0	-1
0	1	1
1	0	1
1	1	1

$$\text{A.F.} = y = \begin{cases} 1, & \text{if } y_{in} > 0.2 \\ 0, & \text{if } -0.2 \le y_{in} \le 0.2 \\ -1, & \text{if } y_{in} < -0.2 \end{cases}$$

I/P x_1	I/P x_2	B	NET (y_{in})	O/P (y)	t (target)	Δw_1 ($\alpha t x_1$)	Δw_2 ($\alpha t x_2$)	Δb (αt)	w_1 (0)	w_2 (0)	b (0)
0	0	1	0	0	-1	0	0	-1	0	0	-1
0	1	1			1						
1	0	1			1						
1	1	1			1						

Ans
→ 5th iteration

Input seq x_1	x_2	After	Updated values w_1	w_2	b
1	1	1st iteration →	1	1	0
1	0	2nd →	1	1	−1
0	1	3rd →	2	1	−1
0	0	4th →	2	2	−1
		5th →	2	2	−1

Perception for the AND function (with bipolar inputs and targets)

I/P patterns	I/P x_1	x_2	O/P y
A	1	1	1
B	−1	1	−1
C	1	−1	−1
D	−1	−1	−1

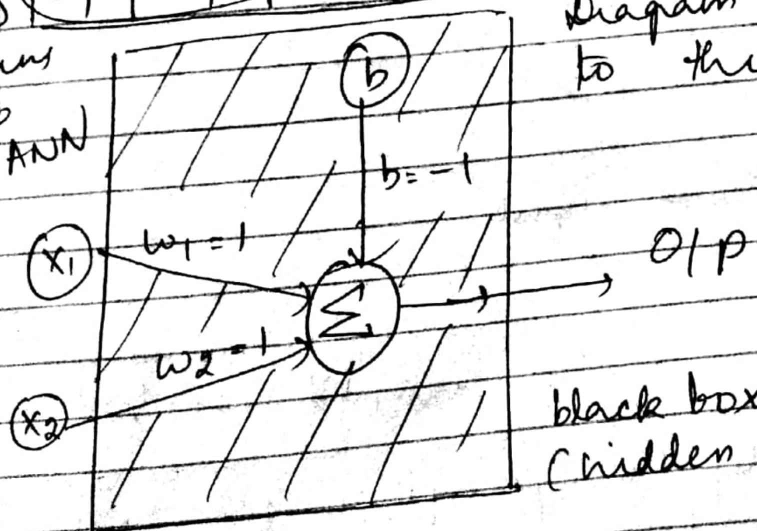
patterns used to train ANN

updated

$\boxed{w_1 = w_2 = 1, \; b = -1}$

↓

Diagram corresponding to this



black box (hidden ANN)

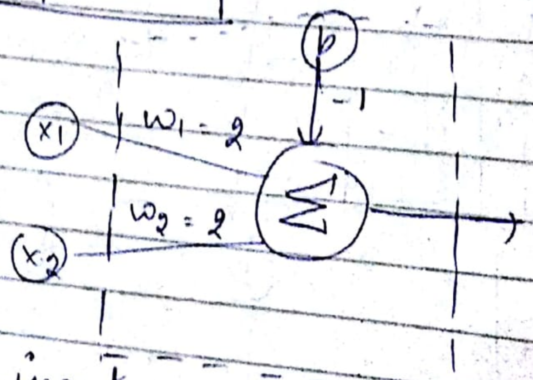
$$y_{in} = b + \sum_i x_i w_i$$
$$= b + x_1 w_1 + x_2 w_2$$
$$= 1$$
$$f(y_{in}) = \text{Applying A.F.}$$
$$= 1$$

(0, 1) } untrained
(1, 0) } input : Generation of unexpected output.

Perceptron for the OR function (binary i/p and bipolar o/p)

Input X_1	X_2	O/P Y
1	1	1
1	0	1
0	1	+1
0	0	−1

① $w_1 = w_2 = 1$
$b = -1$



Untrained input

O/P ⇒ −1 ⇐ (−1, −1)
−1 ⇐ (−1, 0)
1 ⇐ (2, 1)

} random outputs.

Q. Develop a perceptron for AND function with binary i/p's and bipolar targets without bias. Execute 2 complete iterations.

Case I : Considering (0, 0) pattern
Case II : Without (0, 0) pattern

(Assume $\alpha = 1$) and $\theta = 0$.

Sol^n: Truth table for AND function

x_1	x_2	Y
0	0	
0	0	
0	1	
0	1	

x_1	x_2	Y
1	1	1
0	1	-1
1	0	-1
0	0	-1

Bias = 0

$\alpha = 1$

$\theta = 0$

Iteration 1

I/P x_1	I/P x_2	NET (Y_{in})	O/P (Y)	Target (t)	Change in ω $\Delta\omega_1$	$\Delta\omega_2$	Weights ω_1 (0)	ω_2 (0)
1	1	0	0 $Y \neq t$	1	1	1	1	1
0	1	1	1 $Y \neq t$	-1	0	-1	1	0
1	0	0	0 $Y \neq t$	-1	-1	0	0	0
0	0	0	0 $Y \neq t$	-1	0	0	0	0

For iteration 2.

$\omega_1 = \omega_2 = 0.$

Scanned by CamScanner

Q. Using the perception learning rule, find the weights required to perform the following classifications:

Vectors (1, 1 1 1) and (-1 1 -1 -1) are members of class (having target value as 1);

Vector (1 1 1 -1) and (1 -1 -1 1) are not members of class (having target value as -1)

Use learning rate = 1, starting weights as 0.
Test the response of ANN.
Iterations = 2

Sol.

I/P				t
1	1	1	1	1
1	1	1	-1	-1
-1	1	-1	-1	1
1	-1	-1	1	-1

Take $\theta = 0$, $\alpha = 1$

$w_1 = w_2 = w_3 = w_4 = b = 0$.

At the end of 2nd iteration updated values:

$w_1 = -2$

$w_2 = 2$

$w_3 = 0$

$w_4 = 2$

$b = 0$.

Iteration 1

x_1	x_2	x_3	x_4	B	y_m	Y	t	Δw_1	Δw_2	Δw_3	Δw_4	Δb	w_1 (0)	w_2 (0)	w_3 (0)	w_4 (0)	w_b (0)
1	1	1	1	1	0	0 $Y \neq t$	1	1	1	1	1	1	1	1	1	1	1
-1	1	-1	-1	1	-1	-1 $Y \neq t$	1	-1	1	-1	-1	1	0	2	0	0	2
1	1	1	-1	1	4	1 $Y \neq t$	-1	-1	-1	-1	1	-1	-1	1	-1	1	1
1	-1	-1	1	1	1	1 $Y \neq t$	-1	-1	1	1	-1	-1	-2	2	0	0	0

Iteration 2

x_1	x_2	x_3	x_4	B	y_m	Y	t	Δw_1	Δw_2	Δw_3	Δw_4	Δb	w_1	w_2	w_3	w_4	w_b
1	1	1	1	1	0	0 $Y \neq t$	1	1	1	1	1	1	-1	3	1	1	1
-1	1	-1	-1	1	3	1 $Y = t$	1	0	0	0	0	0	-1	3	1	1	1
1	1	1	-1	1	3	1 $Y \neq t$	-1	-1	-1	-1	-1	-1	-2	2	0	2	0
1	-1	-1	1	1	-2	-1 $Y = t$	-1	0	0	0	0	0	-2	2	0	2	0

Check

Ist

x_1	x_2	x_3	x_4
1	1	1	1

$w_1 = 2$
$w_2 = 2$
$w_3 = 0$
$w_4 = 2$
$b = 0$

$Y_{in} = b + \sum x_i w_i$
$= 2$ $\boxed{t=1}$

A.F.

$Y = f(Y_{in}) = 1$

Preset value = 0

$$A.F = \begin{cases} 1, & \text{if } Y_{in} > 0 \\ 0, & -0 \le Y_{in} \le 0 \\ -1, & \text{if } Y_{in} < 0 \end{cases}$$

$Y = f(Y_{in}) = 1$

II $\boxed{-1 \quad 1 \quad -1 \quad -1}$ $t = -1$

$Y_{in} = -2$
$Y = t = -1$

III $\boxed{1 \quad 1 \quad 1 \quad -1}$ $t = -1$

$Y_{in} = -2$
$Y = -1$

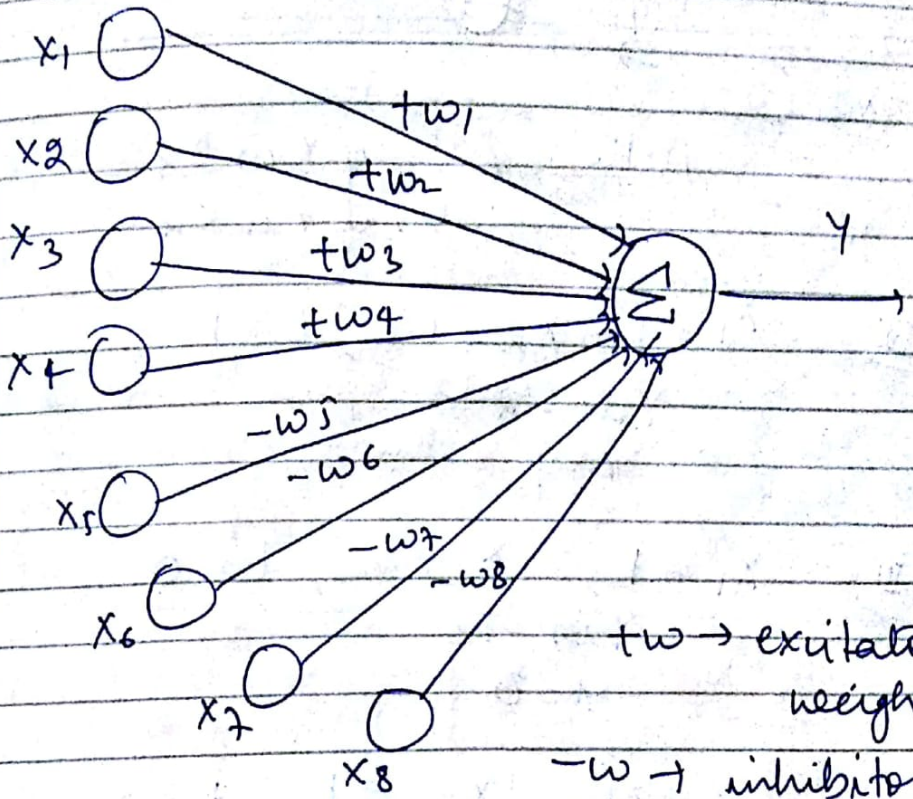
IV $(1 \quad -1 \quad -1 \quad 1)$ $t = -1$

$Y_{in} = -2$
$Y = -1$

Hence, the calculated test o/p values matches with the target o/p values for the corresponding i/p vectors

Mc Culloch - Pitts Neural Model

Architecture of the model



$+w \rightarrow$ excitatory weight

$-w \rightarrow$ inhibitory weight

1. Allows only binary states (0 or 1 only)

2. $f(Y_{in}) = \begin{cases} 1, & \text{if } Y_{in} \geq \theta \\ 0, & \text{if } Y_{in} < \theta \end{cases}$

Ques Generate o/p of logic AND function by Mc-Culloch Pitts model

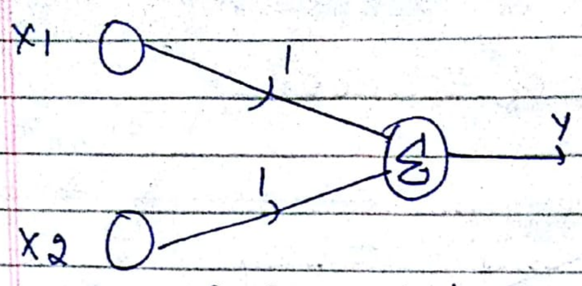
Sol.n AND T.T (in binary)

x_1	x_2	Y
0	0	0
0	1	0
1	0	0
1	1	1

Assume, $w_1 = w_2 = 1$

$\theta = 2$

(B is not considered here)



$$A.F. = \begin{cases} 1, & \text{if } Y_{in} \geq 2 \\ 0, & \text{if } Y_{in} < 2 \end{cases}$$

P-IV $\quad x_1 = 1, \; x_2 = 1, \; t = 1$

$Y_{in} = 2$

$Y = 1$

P-III $\quad x_1 = 1, \; x_2 = 0, \; t = 0$

$Y_{in} = 1$

$Y = 0$

P-II $\quad x_1 = 0, \; x_2 = 1, \; t = 0$

$Y_{in} = 1$

$Y = 0$

P-I $\quad x_1 = 0, \; x_2 = 0, \; t = 0$

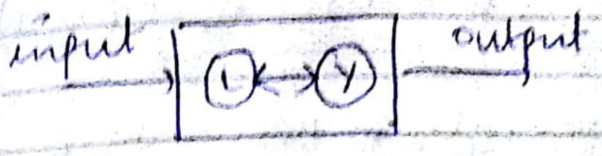
$Y_{in} = 0$

$Y = 0$

Ques 2 Generate OR function using Mc-Culloch Pitts model.

Ques 3 Realise NOT function using Mc-Culloch Pitts model

Hetero associative memory:



Procedure:

1. Weights are initialized using Hebb Rule
2. For each i/p vector do steps 3 to 5:
3. Set the activations for i/p layer unit equal to the current vector X_i
4. Compute NET i/p to the o/p units

$$Y_{inj} = \sum_{i=1}^{n} X_i W_{ij}$$

5. Determine the activation of the o/p units

Examples:

① i/p $[1\ 1\ 0\ 0]$, t $[1, -1]$ or $[-1, 0]$ $\rightarrow f(Y_{inj})$

$$= \begin{cases} 1, & \text{if } Y_{inj} \geq 0 \\ 0, & \text{if } Y_{inj} = 0 \\ -1, & \text{if } Y_{inj} < 0 \end{cases}$$

② $[1\ 1\ 0\ 0][1\ 0] \rightarrow f(Y_{inj}) = \begin{cases} 1, & \text{if } Y_{inj} > 0 \\ 0, & \text{if } Y_{inj} \leq 0 \end{cases}$

Scanned by CamScanner

Problem - I

$S_1 = (1\ 1\ 0\ 0)$, $t_1 = (1, 0)$
$S_2 = (0\ 1\ 0\ 0)$, $t_2 = (1, 0)$
$S_3 = (0\ 0\ 1\ 1)$, $t_3 = (0, 1)$
$S_4 = (0\ 0\ 1\ 0)$, $t_4 = (0, 1)$

A heteroassociative network is given. Find the weight matrix and test the network with training input.

Solⁿ Step I : Initialize the weight using Hebb ~~rule~~ outer product rule method

① $S_1 = (1\ 1\ 0\ 0)$, $t_1 = (1, 0)$

$\boxed{W_1 = S_1^T t_1}$ ⇒ Hebb outer product rule

$$W_1 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}_{4\times1} \begin{bmatrix} 1 & 0 \end{bmatrix}_{1\times2}$$

$$= \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}_{4\times2}$$

② $S_2 = (0\ 1\ 0\ 0)$

$W_2 = S_2^T t_2$

$$w_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} [1, 0]$$

$$w_2 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}_{4 \times 2}$$

(3) $s_3 = (0\ 0\ 1\ 1)$

$$w_3 = s_3^T t_3$$

$$w_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} [0\ 1]$$

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

(4) $s_4 = (0\ 0\ 1\ 0)$ $\quad w_4 = s_4^T t_4$

$$w_4 = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} [0, 1]$$

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

Net weight (final value for all i/ps) $= w_1 + w_2 + w_3 + w_4 = \begin{bmatrix} 1 & 0 \\ 2 & 0 \\ 0 & 2 \\ 0 & 1 \end{bmatrix}$

Using pattern 1 for training, repeating steps 3 to 5.

1st

$$X_1 = \begin{array}{cccc} x_1 & x_2 & x_3 & x_4 \\ \end{array}$$
$$X_1 = [\,1\ \ 1\ \ 0\ \ 0\,] \quad t = [\,1\ \ 0\,]$$

$$Y_{inj} = \sum_{i=1}^{n} x_i w_{ij}$$

$$Y_{in1} = x_1 w_{11} + x_2 w_{21} + x_3 w_{31} + x_4 w_{41}$$

w_{11}, w_{21} → elements of net weight matrix

$$Y_{inj} = 1 + 2 + 0 + 0 = 3$$

$$Y_{in2} = x_1 w_{12} + x_2 w_{22} + x_3 w_{32} + x_4 w_{42} = 0.$$

A.F.

$$Y_1 = f(Y_{in1}) = f(3) = 1$$

$$f(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{if } x \le 0 \end{cases}$$

$$Y_2 = f(Y_{in2}) = f(0) = 0$$

2nd

$$X_2 = [\,0\ \ 1\ \ 0\ \ 0\,] \quad t = [1, 0]$$

$$Y_{in1} = 0 + 2 + 0 + 0 = 2$$
$$Y_{in2} = 0 + 0 + 0 + 0 = 0$$

$Y_1 = (f(Y_{in1})) = f(2) = 1$
$Y_2 = f(Y_{in2}) = f(0) = 0$

3rd $X_3 = [\overset{x_1}{0}\ \overset{x_2}{0}\ \overset{x_3\,x_4}{1\ 1}]$ $t = [\overset{t_1}{0}\ \overset{t_2}{1}]$

$Y_{in1} = 0 + 0 + 0 + 0 = 0$
$Y_{in2} = 0 + 0 + 2 + 0 = 2$

$Y_1 = f(Y_{in1}) = f(0) = 0$
$Y_2 = f(Y_{in2}) = f(2) = 1$

4th $X4 = [0\ 0\ 1\ 0]$ $t = [\overset{t_1}{0},\ \overset{t_2}{1}]$

$Y_{in1} = 0 + 0 + 0 + (1 \times 0) = 0$
$Y_{in2} = 0 + 0 + 0 + 2 = 2$

$Y_1 = f(Y_{in1}) = f(0) = 0$
$Y_2 = f(Y_{in2}) = f(2) = 1$

Problem -2 Check the above problems with [1 1 1 0]

$Y_{in1} = 1 + 2 + 0 + 0 = 3$
$Y_{in2} = 0 + 0 + 2 + 0 = 2$

$Y_1 = f(Y_{in1}) = f(3) = 1$
$Y_2 = f(Y_{in2}) = f(2) = 1$

Problem 3 Do problem 1 with bipolar inputs:

$S_1 = (1\ 1\ -1\ -1)$ $t_1 = (1\ -1)$
$S_2 = (-1\ 1\ -1\ -1)$ $t_2 = (1\ -1)$
$S_3 = (-1\ -1\ 1\ 1)$ $t_3 = (-1\ 1)$
$S_4 = (-1\ -1\ 1\ -1)$ $t_4 = (-1\ 1)$

For testing use – binary
W calculation – bipolar.

$s_1 = (1\ 1\ -1\ -1)$ $\quad t_1 = (1, -1)$

$w_1 = s_1^T t_1$

$$w_1 = \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix} \begin{bmatrix} 1 & -1 \end{bmatrix}$$

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

$$w_2 = \begin{bmatrix} -1 \\ 1 \\ -1 \\ -1 \end{bmatrix} \begin{bmatrix} 1 & -1 \end{bmatrix}$$

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

$$w_3 = \begin{bmatrix} -1 \\ -1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & 1 \end{bmatrix}$$

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

$$w_4 = \begin{bmatrix} -1 \\ -1 \\ 1 \\ -1 \end{bmatrix} \begin{bmatrix} -1 & 1 \end{bmatrix}$$

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

$W = W_1 + W_2 + W_3 + W_4$

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

$$W = \begin{bmatrix} 2 & -2 \\ 4 & -4 \\ -4 & 4 \\ -2 & 2 \end{bmatrix}$$

Testing the 4 patterns now.

1st $\quad X_1 = [1\ 1\ 0\ 0] \quad t = [1\ 0]$

$Y_{in1} = x_1 w_{11} + x_2 w_{21} + x_3 w_{21} + x_4 w_{41}$

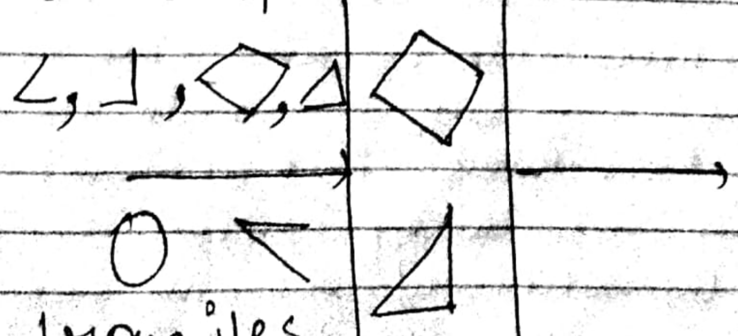
$= 2 + 4 + 0 + 0 = 6$

$Y_{in2} = x_1 w_{12} + x_2 w_{22} + x_3 w_{32} + x_4 w_{42}$

$= -2 - 4 + 0 + 0 = -6$

Auto Associative Memory (image refinement)

correct i/ps



wrong i/ps

Training algorithm

Step 1: Initialize all weights, $i = 1 \dots n$
$j = 1 \dots n$

Step 2: For each vector to be stored follow steps 3 to 4

Step 3: Set activation for each i/p unit $i = 1 \dots n$

$$x_i = s_i$$

Step 4: Adjust the weight for $i = 1$ to n and $j = 1$ to n

$$w_{ij}(new) = w_{ij}(old) + x_i y_j$$

or

Hebb's rule,

$$W = \sum_{p=1}^{n} s^T \cdot w$$

A.F. (normally)

$$y_j = f(y_{inj}) = \begin{cases} 1, & \text{if } y_{inj} > 0 \\ 0, & \text{if } y_{inj} \leq 0 \end{cases}$$

Q. Use Hebb rule to store the vector (1 1 −1 −1) in an auto associative neural network. Find the weight matrix. Test the i/p vector X = (i/p vector). Test the NET with one mistake in the i/p vector. Test the NET with 1 missing data in the i/p vector. Test the NET with 2 missing components in the i/p vector. Test the NET with 2 mistakes in i/p vector

Sol^n:

Step 1: To initialize the weight using Hebb rule

(a)

$$W = \underbrace{S^T(P)}_{i/p}\ \underbrace{t(p)}_{o/p \text{ or target}}$$

$\bar{s}(P) = [1 \ 1 \ -1 \ -1]$

$t(P) = [1 \ 1 \ -1 \ -1]$

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

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

(b)

$X = [1 \ 1 \ -1 \ -1]$

$Y_{in} = \sum x_i w_{ij}$

$$\begin{bmatrix} 1 & 1 & -1 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & 1 \end{bmatrix}$$

$[4 \ 4 \ -4 \ -4] / 4$

$= [1 \ 1 \ -1 \ -1]$

The response vector is same as the i/p vector

Q.2 Consider an auto associative network with bipolar step function as the ~~output~~ activation function and weights set by the Hebb rule ~~above~~ when the main diagonal of the weight matrix is set to be zero.

(1) find the weight matrix to store the vector [1 1 1 1 -1 -1]

(2) Test the response of the network with the same i/p.

Then response vector Y_{in} is same as the i/p vector

(c) Given vector X = [1 1 -1 -1] (with one mistake)

$$\begin{bmatrix} -1 & 1 & -1 & -1 \end{bmatrix}$$

1. $\begin{bmatrix} -1 & 1 & -1 & -1 \end{bmatrix}$
2. $\begin{bmatrix} 1 & -1 & -1 & -1 \end{bmatrix}$
3. $\begin{bmatrix} 1 & 1 & 1 & -1 \end{bmatrix}$
4. $\begin{bmatrix} 1 & 1 & -1 & 1 \end{bmatrix}$

$$Y_{in} = \sum X_i w_{ij}$$

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

$$= \begin{bmatrix} 2 & 2 & -2 & -2 \end{bmatrix} / 2$$

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

(original i/p as answer)

Scanned by CamScanner

CAT 1 Portions upto today
5 Q's × 10 marks

$$Y_{in} = [1 \ \ 1 \ \ -1 \ \ -1]_{1\times 4} \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & 1 \end{bmatrix}_{4\times 4}$$

$$[2 \ \ -2 \ \ -2 \ \ -2] / 2$$

$$= [1 \ \ -1 \ \ -1 \ \ -1]$$

Concl: A network will recognise the mistakes but not the missing components

(d) One missing component in the given i/p vector

1. $[0 \ \ 1 \ \ -1 \ \ -1]$
2. $[1 \ \ 0 \ \ -1 \ \ -1]$
3. $[1 \ \ 1 \ \ 0 \ \ -1]$
4. $[1 \ \ 1 \ \ -1 \ \ 0]$

$\left.\begin{array}{l} +1 \text{ to } -1 \\ -1 \text{ to } +1 \end{array}\right\}$ mistake

$\left.\begin{array}{l} +1 \text{ to } 0 \\ -1 \text{ to } 0 \end{array}\right\}$ missing

Conclusion:

(e) Two mistakes

$$[-1 \ \ -1 \ \ -1 \ \ -1]$$

$$Y_{in} = [0 \ \ 0 \ \ 0 \ \ 0]$$

(f) Two missing

1. $[0 \ \ 0 \ \ -1 \ \ -1]$
2. $[0 \ \ 1 \ \ 0 \ \ -1]$
3. $[0 \ \ 1 \ \ -1 \ \ 0]$
4. $[1 \ \ 0 \ \ 0 \ \ -1]$

11/1/18

Q2. Given vector i/p : $X = [1\ 1\ 1\ 1\ -1\ -1]$

$$W = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}_{6\times1} \begin{bmatrix} 1 & 1 & 1 & 1 & -1 & -1 \end{bmatrix}_{1\times6}$$

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

Now, introducing small error in weight matrix.

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

Even with errors, the system gives the i/p as response.

11/1/18

Bidirectional Associative Memory. (BAM)

1. It is a two - layer network
2. Information flows from i/p to o/p $\rightarrow W$
   and o/p to i/p $\rightarrow W'$

Q. Consider a BAM network (with bipolar vectors) to map two simple letters (size of the letter $\rightarrow 5 \times 3$)